

# **New Cryptanalysis and Modelling for Wireless Networking**

**Mohamed Alzaabi**

*School of Engineering and Technology*

*UNIVERSITY OF HERTFORDSHIRE*

Thesis submitted to the University of Hertfordshire

in partial fulfilment of the requirements of the

Degree of Doctor of Philosophy

Nov 2015

## Abstract

High data rates and interoperability of venter devices have made WiMAX a prime desire for use worldwide.

WiMAX is based on the IEEE 802.16 standard. IEEE 802.16a, b, c & d versions were updated within three years of the first launch of WiMAX. However, during those early years reports were published that highlighted the security weaknesses of the standard. These weaknesses prompted the IEEE to issue a new version, 802.16e to tackle the security issues. Despite this security enhancement, WiMAX remains vulnerable.

This research project looks at the vulnerability of WiMAX 802.16e Subscriber Station/Mobile Station authentication at the initial entry and proposes approaches to the prevention of Denial of Service (DoS) attacks at this point in order to secure the Media Access Control (MAC) layer from such threats.

A new protocol has been designed and developed to provide confidentiality, authentication and integrity to WiMAX users. This new protocol is integrated with Z algorithm (an algorithm described later in this paper) to provide:

- Confidentiality of management messages
- Message Authentication code
- ID to provide for message integrity and user authentication.

A simulation package was also required, to prove that a linear load of DoS attack would disable or exhaust the capacity of the base station of a WiMAX network, as well as providing other simulation functions. The freely available simulation tool NIST (NIST IPsec (Internet Protocol Security) and IKE (Internet Key Exchange) Simulation) is oriented towards fixed network communications (NIST, 2003). There are no other relevant simulation tools; hence the purpose of this research project is to develop a new tool to simulate WiMAX security vulnerabilities and test the new protocol.

## Declaration Statement

I certify that the work submitted is my own and that any material derived or quoted from the published or unpublished work of other persons has been duly acknowledged.

Student Full Name: Mohamed Abdulla Alzaabi

Student Registration Number: 12229593

Signed:  .....

Date: Oct 2015

## Acknowledgement

First of all, thanks to God for making me reach this level of my education. Without his blessing I would never have reached this far.

It was not an easy journey but with the full support and encouragement of my family I can say, today, it was an enjoyable one and I would never have been able to complete it without them always being there for me.

I would like to thank my advisor, Professor T Alukaidey, for his continuous support during my PhD study. Having a demanding advisor like him incited me to widen my research from different angles. He really lightened my way when it was so dark on my research path.

My sincere thanks also go to Mr. David Brewer for proofreading my thesis. His effort and time is really appreciated.

Last but not least, I can say today, it is a dream comes true.

## Table of Contents

Abstract.....	2
Declaration Statement.....	3
Acknowledgement.....	4
Abbreviations .....	10
Terminology .....	16
List of Figures .....	17
Chapter 1: Introduction and Problem Definition .....	19
1.1 Overview .....	19
1.2 WiMAX Architecture .....	20
1.2.1 WiMAX main three sections .....	21
1.2.1.1 User Terminals .....	21
1.2.1.2 Access Service Network (ASN).....	21
1.2.1.3 Connectivity Service Network (CSN) .....	22
1.2.2 Vulnerable parts of WiMAX to breach security .....	22
1.3 Problem Definition .....	23
1.4 The Aim of the Thesis.....	23
1.5 Research Methodology.....	23
1.5.1 Simulation Model.....	27
1.6 Report Guidance .....	28
1.7 Summary.....	29
Chapter 2: Wireless Data Networks .....	30
2.1 Network Types.....	30
2.2 Wireless Network Technology Brief History .....	31
2.2.1 WiBro .....	32
2.2.2 WiMAX.....	32
2.2.2.1 WiMAX Protocols.....	33
2.2.2.2 Overview .....	33
2.2.2.3 Security Sub-Layer .....	34

2.2.2.4 WiMAX MAC – Protocol Data Unit (PDU) .....	35
2.2.2.5 WiMAX IEEE 802.16e Authentication/Authorisation Protocol .....	37
2.2.2.5.1 Privacy Key Management (PKM) Protocol.....	37
2.2.2.5.2 PKM v1 .....	38
2.2.2.5.3 PKM RSA Authentication .....	39
2.2.2.5.4 Authorisation via RSA Authentication Protocol .....	39
2.2.3 Global System for Mobile Communication (GSM) .....	40
2.2.3.1 GSM Architecture .....	41
2.2.3.2 GSM Security Algorithms.....	42
2.2.4 Long Term Evolution (LTE) .....	42
2.2.4.1 LTE and LTE Radio Interface architectures .....	43
2.2.4.2 LTE Security .....	46
2.2.4.3 Security and Cryptography Algorithms.....	46
2.2.4.3.1 Security Features and Functions.....	46
2.2.4.4 Authentication and Cipher Key Generator Algorithms.....	47
2.2.4.4.1 A3 Authentication Algorithm .....	47
2.2.4.4.2 A8 Cipher Key Generator Algorithm .....	48
2.2.4.4.3 A5 Ciphering/Deciphering Algorithm.....	49
2.2.5 Vulnerability of GSM and LTE networks .....	51
2.2.5.1 GSM .....	51
2.2.5.1.1 GSM networks are vulnerable to IMSI detach attacks .....	51
2.2.5.1.2 SS7 .....	51
2.2.5.2 LTE.....	52
2.2.6 The current Status of LTE Security .....	53
2.2.7 WiMAX-LTE Comparison .....	54
2.2.8 Merging.....	54
2.3 Summary .....	55
Chapter 3: The Proposed Algorithm: Zaabi Security Algorithm (Z Algorithm or ZA).....	56
3.1 Overview .....	56

3.2 Overall ZA .....	56
3.2.1 Exponential Back Off Counter (EBOC).....	57
3.2.2 Enhanced Hash-Based Method Authentication Code: (EHMAC).....	58
3.2.3 Simple Authentication Protocol.....	61
3.2.4 Authentication using Proxy Base Station.....	62
3.2.5 Zaabi RSA (ZRSA).....	63
3.2.5.1 Cryptography.....	63
3.2.5.1.1 Encryption Techniques.....	63
3.2.5.2 ZRSA.....	69
3.2.5.3 Working Principle of One Part of ZRSA .....	70
3.2.5.3.1 Key Generation .....	70
3.2.5.3.2 Encryption.....	72
3.2.5.3.3 Decryption.....	72
3.2.5.3.4 Working Example for ZRSA Algorithm.....	73
3.2.5.4 Brute Force Attack on ZRSA.....	75
3.2.5.5 Working Principle of Modified ZRSA .....	78
3.2.5.5.1 Key Generation .....	78
3.2.5.5.2 Encryption and Decryption .....	80
3.2.5.5.3 Example.....	81
3.2.5.6 Comparison Between Brute Force Attacks on RSA and Modified ZRSA.....	83
3.2.6 Firewall at Client's Premises .....	89
3.3 Zaabi Security Algorithm based on Message Authentication Code for WiMAX (Z Algorithm).....	90
3.4 Summary.....	91
Chapter 4: The Certificate Authority Tools of the Proposed Algorithm: Zaabi Algorithm .....	93
4.1 Overview .....	93
4.2 Certificate Authority .....	93
4.2.1 Certificate Authority Control .....	93
4.2.2 X.509 .....	93

4.2.3 Universal Certificate Exchange in WiMAX .....	95
4.2.4 Z Algorithm Version A .....	98
4.2.5 Z Algorithm Version B .....	98
4.3 Summary .....	99
Chapter 5: Network Security Simulator (NetSecSim) Design.....	100
5.1 Simulator Overview .....	100
5.1.1 Simulator Concept and Design.....	100
5.1.2 Simulator Flowchart .....	101
5.2 Initialisation Process.....	103
5.2.1 Scanning and Synchronisation .....	104
5.2.2 Achieving Downlink Channel.....	105
5.2.3 Achieving Uplink Channel .....	105
5.2.4 Initial Ranging .....	107
5.2.5 Basic Capabilities.....	108
5.2.6 SS Authorisation and Key Exchange.....	110
5.2.7 Registration.....	111
5.2.8 Initiate IP Connectivity.....	113
5.2.8.1 Initiate Time of Day .....	114
5.2.10 Transfer Operational Parameters .....	115
5.3 MAC Management Messages in WiMAX .....	116
5.3.1 MAC PDU Formats .....	116
5.3.2 MAC Header Formats .....	117
5.4 NetSecSim Simulator Security .....	120
5.4.1 Encryption and Decryption Algorithms.....	120
5.4.2 NetSecSim Simulator with DoS Attack .....	122
5.5 Summary .....	125
Chapter 6: Conclusion and Future Work .....	126
6.1 Conclusion.....	126
6.2 Future Work.....	127

References .....	128
Appendix A .....	131
Appendix B .....	133
Appendix C .....	134

## Abbreviations

<b>Acronym</b>	<b>Description</b>
<b>3GPP</b>	3rd Generation Partnership Project
<b>AAA</b>	Authentication, Authorization and, Accounting
<b>ACK</b>	Acknowledge
<b>ADS</b>	User Domain Security
<b>AES</b>	Advanced Encryption Standard
<b>AK</b>	Authorization Key
<b>AMPS</b>	Advanced Mobile Phone System
<b>ANSI</b>	American National Standards Institute
<b>ARQ</b>	Automatic Repeat reQuest
<b>ASN</b>	Access Service Network
<b>AUC</b>	Authentication Centre
<b>AV</b>	Authentication Vectors
<b>BFA</b>	Brute Force Attack
<b>BR</b>	Bandwidth Request
<b>BS</b>	Base Station
<b>BSC</b>	Base Station Controller
<b>BTS</b>	Base Transmitter Station
<b>BW</b>	BandWidth
<b>CA</b>	Certification Authority
<b>CDMA</b>	Code Division Multiple Access
<b>CID</b>	Connection IDentifier
<b>CK</b>	Cipher Key
<b>CPE</b>	Customer Premise Equipment
<b>CPS</b>	Common Part Sub-layer
<b>CRC</b>	Cyclic Redundancy Check
<b>CS</b>	Convergence Sub-layer
<b>CSMA/CA</b>	Carrier Sense Multiple Access/Collision Avoidance
<b>CSN</b>	Connectivity Service Network
<b>CTS</b>	Clear to Send

<b>DCD</b>	Downlink Channel Descriptor
<b>DES</b>	Data Encryption Standard
<b>DHCP</b>	Dynamic Host Configuration Protocol
<b>DL-MAP</b>	DownLink map
<b>DoS</b>	Denial of Service
<b>DSL</b>	Digital Subscriber Line
<b>EA</b>	Encryption Algorithm.
<b>EAP</b>	Extensible Authentication Protocol
<b>EBOC</b>	Exponential Back Off Counter
<b>EDGE</b>	Enhanced Data rates for GSM Evolution
<b>EHMAC</b>	Enhanced Hash-Based Method Authentication Code
<b>eNB</b>	evolved Node B
<b>EP</b>	Encryption Protocol
<b>EPC</b>	Evolved Packet Core
<b>EPS</b>	Evolved Packet System
<b>ETSI</b>	European Telecommunications Standards Institute
<b>E-UTRAN</b>	Evolved Universal Terrestrial Access Network
<b>FDD</b>	Frequency Division Duplex
<b>FF</b>	Flip Flop
<b>GCD</b>	Greatest Common Divisor
<b>GPRS</b>	General Packet Radio Services
<b>GPS</b>	Global Positioning System
<b>GSM</b>	Global System for Mobile Communications
<b>GW</b>	Gateway
<b>HE</b>	Home Environment
<b>HLR</b>	Home Location Register
<b>HMAC</b>	Hash-based Message Authentication Code
<b>HSPA</b>	High Speed Packet Access
<b>IEEE</b>	Institute of Electrical and Electronics Engineers
<b>IETF</b>	Internet Engineering Task Force
<b>IK</b>	Interchange Key
<b>IKE</b>	Internet Key Exchange

<b>IMS</b>	Information Management System
<b>IMSI</b>	International Mobile Subscriber Identity
<b>IDU</b>	InDoor Unit
<b>IP</b>	Internet Protocol
<b>IPSec</b>	Internet Protocol Security
<b>ISDN</b>	Integrated Services Digital Network
<b>ITU-T</b>	International Telecommunication Union- Telecommunication
<b>Kc</b>	Ciphering Key
<b>KEK</b>	Key Encryption Key
<b>Ki</b>	Individual Subscriber Authentication Key
<b>LLC</b>	Logical Link Control
<b>LOS</b>	Line Of Site
<b>LSB</b>	Least Significant Bit
<b>LTE</b>	Long Term Evolution
<b>MAC</b>	Media Access Control
<b>MACA</b>	Multiple Access with Collision Avoidance
<b>MACAW</b>	Multiple Accesses with Collision Avoidance for Wireless
<b>ME</b>	Mobile Equipment
<b>MIMO</b>	Multiple Input, Multiple Output
<b>MIP HA</b>	Mobile IP Home Agent
<b>MIPS</b>	Million Instructions Per Second
<b>MITM</b>	Man In The Middle
<b>MM</b>	Management Messages
<b>MMMP</b>	MAC Management Message Payload
<b>MS</b>	Mobile Station
<b>MSB</b>	Most Significant Bit
<b>MSC</b>	Mobile Switching Centre
<b>NAS</b>	Network Access Security
<b>NDS</b>	Network Domain Security
<b>NetSecSim</b>	Network Security Simulator
<b>NIST</b>	National Institute of Standards and Technology
<b>NLOS</b>	None Line Of Site

<b>NMT</b>	Nordic Mobile Telephone
<b>NSA</b>	National Security Agency
<b>NSP</b>	Network Service Providers
<b>ODU</b>	Out Door Unite
<b>OFDMA</b>	Orthogonal Frequency Division Multiple Access
<b>OMC</b>	Operations and Maintenance Centre
<b>OSA</b>	Open System Authentication
<b>OSI</b>	Open Systems Interconnection
<b>PAK</b>	Product Authorization Key
<b>PDCP</b>	Packet Data Convergence Protocol
<b>PDU</b>	Protocol Data Unit
<b>PEK</b>	Product Encryption Key
<b>PHY</b>	Physical Layer
<b>PKCS</b>	Public Key Cryptography Standards
<b>PKM</b>	Privacy Key Management
<b>PS</b>	Proxy Server
<b>PSK</b>	Pre Shared Key
<b>PSTN</b>	Public Switched Telephone Network
<b>PtP</b>	Point to Point
<b>QC</b>	Quantum Computer
<b>QoS</b>	Quality of Service
<b>RAND</b>	RANdOm Number
<b>RFC</b>	Request for Comments
<b>RLC</b>	Radio Link Control
<b>RNC</b>	Radio Network Controllers
<b>RNG-REQ</b>	Ranging Request
<b>RNG-RSP</b>	Ranging Response
<b>RSSI</b>	Received Signal Strength indicator
<b>RTS</b>	Request to Send
<b>SA</b>	Security Association
<b>SAE</b>	System Architecture Evolution
<b>SAID</b>	Security Association IDentities

<b>SAP</b>	Service Access Point
<b>SBC</b>	Station Basic Capability
<b>SC-FDMA</b>	Single Carrier Frequency Division Multiple Access
<b>SDU</b>	Service Data Unit
<b>SGSN</b>	Serving GPRS Support Node
<b>SI</b>	Signal Initialisation
<b>SIADDR</b>	Server Ip ADDRESS
<b>SIM</b>	Subscriber Identity Module
<b>SKA</b>	Shared Key Authentication
<b>SMS</b>	Short Message Service
<b>SN</b>	Service Node
<b>SNMP</b>	Simple Network Management Protocol
<b>SOFDMA</b>	Scalable Orthogonal Frequency Division Multiplexing Access
<b>SRES</b>	Signed RESponse
<b>SS</b>	Subscriber Station
<b>SS7</b>	Signalling System 7
<b>SSBC</b>	Subscriber Station Basic Capability
<b>TA</b>	Terminal Adapter
<b>TACS</b>	Total Access Communication Systems
<b>TDD</b>	Time Division Duplex
<b>TE</b>	Terminal Equipment
<b>TEK</b>	Traffic Encryption Keys
<b>TFTP</b>	Trivial File Transfer Protocol
<b>TMSI</b>	Temporary Mobile Subscriber Identity
<b>TSSI</b>	Temporal Subscriber Station Identity
<b>UCD</b>	Uplink Channel Descriptor
<b>UE</b>	User Equipment
<b>UDP</b>	User Datagram Protocol
<b>UDS</b>	User Domain Security
<b>UL-MAP</b>	Up Link for Multiple Access Protocol
<b>UMTS</b>	Universal Mobile Telecommunications System
<b>USIM</b>	User Services Identity Module

<b>VLR</b>	Visitor Location Register
<b>WAN</b>	Wide Area Network
<b>WiFi</b>	Wireless Fidelity
<b>WLAN</b>	Wireless Local Area Network
<b>WMAN</b>	Wireless Metropolitan Area Network
<b>WPAN</b>	Wireless Personal Area Network
<b>WiBro</b>	Wireless Broadband
<b>WiMAX</b>	Worldwide Interoperability for Microwave Access
<b>ZA</b>	Zaabi Algorithm

## Terminology

- **n** is the **modulus**
- **E** is the **public exponent**
- **d** is the **private exponent**
- The **public key** consists of the pair  $(E, n)$  generated by ZRSA
- The **private key** consists of the pair  $(d, n)$  generated by ZRSA
- The **public keys** consist of the pairs  $(E_p, n_p)$ ,  $(E_q, n_q)$  &  $(E_z, n_z)$  generated by Modified ZRSA via primes  $p$ ,  $q$  &  $r$
- The **private keys** consist of the pairs  $(d_p, n_p)$ ,  $(d_q, n_q)$  &  $(d_z, n_z)$  generated by Modified ZRSA via primes  $p$ ,  $q$  &  $z$

## List of Figures

FIGURE 1.1.1: IEEE 802.16 – 2004 STANDARDS SCOPE PROTOCOL LAYERS IN OSI MODEL .....	20
FIGURE 1.2.1: WIMAX NETWORK IP-BASED ARCHITECTURE .....	21
FIGURE 1.5.1.1: SIMULATION MODEL FOR RSA WITHOUT DOS ATTACK .....	27
FIGURE 1.5.1.2: SIMULATION MODEL FOR ZA WITH DOS ATTACK .....	28
FIGURE 2.1.1: NETWORK TYPES.....	30
FIGURE 2.2.1: EVOLUTION OF WIRELESS STANDARDS .....	31
FIGURE 2.2.2.2.1: WIMAX NETWORK.....	33
FIGURE 2.2.2.3.1: MAC & PHYSICAL LAYER OF 802.16 SHOWING MAC PRIVACY SUB-LAYER (ULVAN, ET AL., 2009).....	34
FIGURE 2.2.2.3.2: SECURITY SUB-LAYER .....	35
FIGURE 2.2.2.4.1: GENERIC MAC PDU .....	36
FIGURE 2.2.2.4.2: GENERIC MAC PDU .....	37
FIGURE 2.2.2.5.1.1: SECURITY KEY.....	38
FIGURE 2.2.3.1.1: GSM SYSTEM ARCHITECTURE.....	41
FIGURE 2.2.4.1: NETWORK SOLUTIONS FROM GSM TO LTE.....	43
FIGURE 2.2.4.1.1: OVERALL LTE ARCHITECTURE .....	44
FIGURE 2.2.4.1.2: LTE RADIO INTERFACE ARCHITECTURE .....	45
TABLE 2.2.4.1.1: LTE RADIO INTERFACE LAYERS .....	45
FIGURE 2.2.4.4.1.1: PROCESS OF AUTHENTICATION BETWEEN BS AND MS.....	48
FIGURE 2.2.4.4.3.1: INPUTS AND OUTPUT OF A5 ALGORITHM .....	50
FIGURE 2.2.4.4.3.2: BASIC DESIGN OF LINEAR FEEDBACK SHIFT REGISTER WITH 3 FLIP-FLOP CLOCKS.....	50
FIGURE 2.2.4.4.3.3: CIPHER/DECIPHER EXAMPLE .....	51
FIGURE 2.2.5.2.1: AUTHENTICATION AND KEY AGREEMENT PROTOCOL .....	53
TABLE 2.2.7.1: TECHNICAL DIFFERENCES BETWEEN WIMAX (IEEE 802.16M) AND LTE-ADVANCED TO DELIVER 4G .....	54
FIGURE 3.2.1: Z ALGORITHM .....	57
FIGURE 3.2.2: OVERALL Z ALGORITHM OVER WIMAX ARCHITECTURE .....	57
FIGURE 3.2.2.1: MESSAGE AUTHENTICATION WITH MAC .....	59
FIGURE 3.2.2.2: MESSAGE BASED MESSAGE AUTHENTICATION AND CONFIDENTIALITY.....	60
FIGURE 3.2.3.1: SIMPLE AUTHENTICATION PROTOCOL.....	61
FIGURE 3.2.4.1: AUTHENTICATION USING PROXY BASE STATION.....	62
FIGURE 3.2.5.1.1.2.1.1.1: RSA BASED MAC LAYER IN IEEE 802.16.....	65
FIGURE 3.2.5.1.1.2.4.1: ENCRYPTION USING PUBLIC KEY.....	66
FIGURE 3.2.5.1.1.2.5.1: DECRYPTION USING PUBLIC KEY.....	67
IF $E = R_{i-1}$ AND $\Phi = R_{i-2}$ , USE: .....	71
FIGURE 3.2.5.3.2.1: ENCRYPTION USING PUBLIC KEY.....	72
FIGURE 3.2.6.1: FIREWALL AT CLIENT’S PREMISES.....	89
FIGURE 3.3.1: FIRM SECURITY MANAGEMENT SYSTEM FOR WIMAX NETWORK UNDER ZA .....	90
FIGURE 4.2.2.1: X.509 CERTIFICATE .....	94

FIGURE 4.2.3.1: PUBLIC KEY EXCHANGE.....	96
FIGURE 4.2.3.2: PUBLIC KEY EXCHANGE.....	97
FIGURE 4.2.4.1: RANGING WITH X.509 CERTIFICATE.....	98
FIGURE 4.2.5.1: RANGING MESSAGES EXCHANGE WITH SHARED KEY PROTOCOL.....	99
FIGURE 5.1.1.1: NETSECSIM SIMULATOR STORYBOARD.....	100
FIGURE 5.1.2.1: NETSECSIM FLOWCHART.....	101
FIGURE 5.2.1: SIGNAL INITIALISATION MESSAGES.....	104
FIGURE 5.2.1.1: SCANNING PROCESS.....	104
FIGURE 5.2.2.1: DL-MAP AND DCD MESSAGES.....	105
FIGURE 5.2.3.1: DL-MAP AND UCD MESSAGES.....	106
FIGURE 5.2.3.2: UL-MAP AND DL-MAP MESSAGES.....	106
FIGURE 5.2.3.3: ACHIEVING UPLINK CHANNEL PARAMETERS.....	107
FIGURE 5.2.4.1: RNG-REQ AND RNG-RSP MESSAGES.....	108
FIGURE 5.2.5.1: SBC-REQ AND SBC-RSP MESSAGES.....	109
FIGURE 5.2.5.2: BASIC CAPABILITIES (SS).....	109
FIGURE 5.2.5.3: BASIC CAPABILITIES (BS).....	110
FIGURE 5.2.6.1: PKM-REQ AND PKM-RSP MESSAGES.....	111
FIGURE 5.2.7.1: REGISTRATION (SS).....	112
FIGURE 5.2.7.2: WAIT FOR REG-RSP.....	112
FIGURE 5.2.7.3: REG-REQ AND REG-RSP MESSAGES.....	113
FIGURE 5.2.8.1: DHCP MESSAGE.....	114
FIGURE 5.2.8.1.1: TIME OF DAY REQUEST AND RESPONSE.....	115
FIGURE 5.2.10.1: TFTP-CPLT AND TFTP-RSP.....	115
FIGURE 5.3.1: MAC MANAGEMENT MESSAGES.....	116
FIGURE 5.3.1.1: MAC PDU FORM.....	117
FIGURE 5.3.2.1: AUTHENTICATED REPLY FROM BS.....	119
TABLE 5.3.2.3: MAC PDU FRAME FIELDS.....	119
FIGURE 5.4.1.1: ZRSA AND MAC DIGEST CIPHERTEXTS.....	121
FIGURE 5.4.2.1: BW REQUEST MAC HEADER.....	122
FIGURE 5.4.2.2: SELECTING DOS ATTACK OPTION WITH NETSECSIM.....	123
FIGURE 5.4.2.3: SOLUTION TO DOS ATTACK.....	123

## Chapter 1: Introduction and Problem Definition

### 1.1 Overview

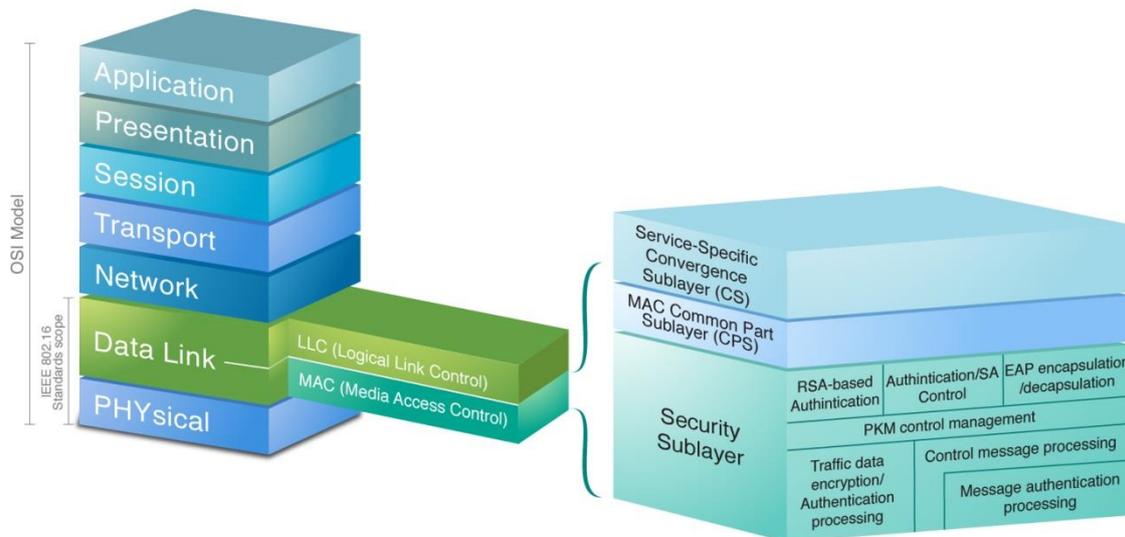
The current century has witnessed the development of an astonishingly-complex suite of protocols and endurable wireless network systems such as WiMAX and LTE. WiMAX is a wireless communication standard, which is well-structured and documented. WiMAX represents the state of the art of wireless networks at the current time. However, the main weakness of WiMAX is the compatibility issue, whilst LTE operators' and customers' have no problem to use the same device for the same coverage of 2G, 3G and 4G. It has been decided that this research is to be based on WiMAX protocols because of its early history, backed up by the soundness and formal definition of the published IEEE standard.

Wireless network popularity has led to its widespread use worldwide at a spectacular rate. The spectrum of use of wireless networks ranges from simple text messages to most confidential document transmissions to billions of pound transactions. The question is how secure is wireless networking and how strong is the law and enforcement to protect lawful usage of the internet. The answer is in the hand of hackers. More attacks result in less secure network systems. The security of wireless networking is a major issue and concern to all types of users: civilian, commercial, military and even the hackers themselves.

WiMAX is an evolving wireless technology based on IEEE 802.16 standard. The IEEE 802.16 standard defines the security mechanisms in WiMAX. The security sub-layer in WiMAX Networks is where authentication, authorization and encryption take place.

This thesis starts with a look at the WiMAX architecture and its components. Then, it addresses the IEEE 802.16 protocol layer associated with Security issues. Next, it tackles some of the WiMAX vulnerabilities that could cause DoS attacks. After that, it examines some of the up today encryption protocols. Finally, it suggests a way forward to tackle some of the possible causes of DoS attacks.

The WiMAX IEEE 802.16 standard uses the Open Systems Interconnection (OSI) network reference seven-layer model. The OSI model is shown in Figure 1.1.1. The OSI model divides the functions of different protocols into a series of layers. The two lowest layers are called the Physical (PHY) Layer and the Data Link Layer. The Data Link layer consists of Logical Link Control (LLC) and Media Access Control (MAC).



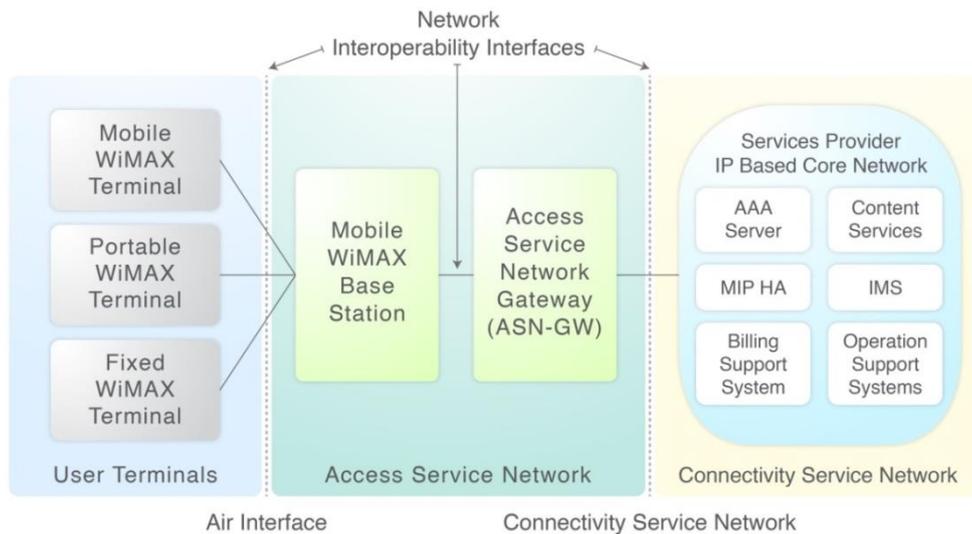
**Figure 1.1.1: IEEE 802.16 – 2004 Standards Scope Protocol Layers in OSI Model**

The physical connection between the communicated parties is created by the PHY layer while the establishment and maintenance of this connection is the responsibility of the MAC layer. The 802.16 standard defines only these two lowest layers (Nuaymi, 2007). Moreover, it divides the MAC layer into three sub-layers, Convergence Sub-layer (CS), Common Part Sub-layer (CPS) and Security Sub-layer (Nuaymi, 2007).

## 1.2 WiMAX Architecture

WiMAX Architecture is designed and developed for all-Internet Protocol (IP) platforms and employs only packet technology, without any legacy circuit telephony. Figure 1.2.1 presents the WiMAX architecture. This IP-based WiMAX network architecture consists of three main sections, namely:

- a) User Terminals,
- b) Access Service Network (ASN) and
- c) Connectivity Service Network (CSN).



**Figure 1.2.1: WiMAX network IP-based Architecture**

These three sections are explained briefly in section 1.2.1.

### 1.2.1 WiMAX main three sections

WiMAX consists of three main sections, User Terminals, Access Service Network and Connectivity Service Network.

#### 1.2.1.1 User Terminals

These are the terminals that work with the Base Station (BS) to provide wireless access functions and support mobility such as Mobile Station (MS) and Subscriber Station (SS). The MS and SS enable connection to a variety of servers through the ASN (NEC, 2012).

#### 1.2.1.2 Access Service Network (ASN)

The Access Service Network (ASN) sets and defines the network functions for providing WiMAX MS and SS with wireless access (NEC, 2012). The ASN consists of network elements (such as one or more BS and ASN Gateways) and provides the following functions:

- Layer 2 connectivity with MS and SS
- Transfers AAA messages (Authentication, Authorization and, Accounting) to WiMAX subscribers
- Relay functions for building MS and SS and Layer 3 connectivity
- Wireless resource management
- ASN and CSN secured mobility
- Paging

### 1.2.1.3 Connectivity Service Network (CSN)

CSN consists of various network elements, including AAA, home agents, routers, gateways and user databases. CSN provides WiMAX subscribers connectivity with the following functions:

- Allocates MS and SS IP addresses and endpoint parameters for user sessions
- Internet access
- AAA proxy/server
- Policy and authorization control based on user subscription profiles
- ASN-CSN tunnelling support
- Accounting among WiMAX subscribers and settlement among operators
- Tunnelling between CSNs for roaming
- Mobility between ASNs
- Services based on positional information, connectivity with peer-to-peer services, provisioning, etc. (NEC, 2012)

The end to end WiMAX network architecture significantly supports mobility and handover which includes:

- Vertical or inter-technology handovers under multi-mode operation
- IPv4 and IPv6 based mobility management
- Roaming between Network Service Providers (NSPs)
- Seamless handover up to vehicular speed satisfying bounds of service disruptions

WiMAX network architecture supports Quality of Service (QoS) via differentiated levels of QoS, admission control, bandwidth management, and other appropriate policies (Chen and De Marca, 2008)

### 1.2.2 Vulnerable parts of WiMAX to breach security

WiMAX is based on the IEEE 802.16 standard, which is recognised as having security flaws, including vulnerabilities in authentication and key management protocols. Message replay is one of the most well-known attacks on authentication and authenticated key establishment protocols. The idea behind replay based attacks is to flood a network with false management frames, which cause a denial of service (DoS) (Doe, 2011).

In IEEE the 802.16 standard, the Privacy Sub-layer sits on the top of Physical layer. Therefore, 802.16 networks are vulnerable to physical layer attacks such as jamming and scrambling.

Jamming is done by initiating a source of strong noise to significantly decrease the capacity of the channel, thus denying services (DoS) to all parties. Scrambling is another type of jamming, but it takes place for a short interval of time targeting specific frames. Control or management messages can be scrambled, but it is not possible with delay sensitive message i.e., scrambling Uplink slots are quite difficult, because the attacker has to interpret control information and to send noise during a particular interval (Barbeau, 2005).

The main purpose of the Privacy Sub-layer is to protect service providers against theft of service, rather than guarding network users. It is obvious that the privacy layer only guards data at the OSI layer two (data link), whereas it does not guarantee end to end encryption of user data. Likewise, it does not protect the physical layer from being intercepted (Boom, 2004).

Identity theft is another threat, which is reprogramming a device with the hardware address of another device. The address can be stolen over the air by interrupting management messages (Hasan, 2006).

There is clear evidence that the IEEE group does not have a solution to the weaknesses of security issues despite the launch of 802.16e in 2004.

This research project looks at the vulnerability of WiMAX 802.16e Subscriber Station/Mobile Station authentication, at various entry levels of the protocol and suggests a suite solution. For example, the initial entry part of WiMAX is inherited a weaknesses that will allow DoS attacks. To prevent Dos attacks, a solution has been suggested and will be presented in Chapter 3. The solution has secured the MAC layer from possible undesired attacks.

### 1.3 Problem Definition

Despite the security enhancements, security vulnerability is still an issue that exists with WiMAX.

### 1.4 The Aim of the Thesis

The aim of this thesis is to highlight some of the security threats that face today's wireless networks such as WiMAX, address these threats and present a design solution as a suite to tackle all known weaknesses to WiMAX.

### 1.5 Research Methodology

A research methodology describes the work undertaken by a project to make-up, manages and plan the process of certain project developments or describe the process of finding the answers to the research questions. At each of the operational steps in the research process,

one is required to select from a combination of “methods”, “procedures” and “models” of the research methodology which helps achieve the stated objectives more accurately.

This section gives an overview of the techniques used to analyse the security of wireless networks particularly taking WiMAX as a case study and considering the network entry process which is prone to Man-In-The-Middle (MITM) attacks which can lead to DoS attacks. In addition, as RSA is the de-facto standard of encryption/decryption algorithm and has been factorised or cracked with various bit lengths, a special study has been covered in the crypto analysis to identify its loopholes and perhaps suggest an alternative encryption/decryption algorithm.

The research specific study is to find security weaknesses in the security of wireless network protocols that are designated for 4G & possibly the upcoming 5G wireless network standards and propose effective solutions. One excellent example of such networks is IEEE 802.16e WiMAX. The data will be primarily collected from literature, case studies and articles. The target population will be Internet users and specifically WiMAX clients. The technique employed for evaluating the security protocol via the underlying RSA algorithm (named after its authors Rivest, Shamir & Adleman) is by simulation using C# programming. The simulator is suggested, designed and developed by the author. C# is used due to the availability of its encryption & decryption tool box and is known to be user friendly.

The network entry problem identified is resolved by applying a novel protocol called Zaabi Algorithm (ZA), used to secure the management messages and establish a secure channel, preventing unauthorised third parties accessing the channel. The details of this protocol are given in chapter 3 and 4 of this thesis. Two approaches are considered as Z Algorithm Version A and Z Algorithm Version B.

The authentication algorithm used for ZA, is ZRSA; and is evaluated for its effectiveness in the encryption and decryption processes using mathematical methods, before implementing the algorithm in code, for verification.

Finally, the algorithm ZA, is to be implemented and deployed using the C# environment and is to be tested with different character inputs, which will be encrypted and decrypted in a normal scenario case-study without any security threat. The simulation is to be repeated in a second scenario with different characters, but involving a DoS security threat which leads to the Mobile Station (MS) device, to time out through not receiving service as the Base Station (BS) resources are exhausted by the bogus messages sent by the attacker. The results of the attacks are to be evaluated based on the time taken to process each of the bogus messages

adding up to the overall down time of the system. A comparative analysis will be carried out with a published data set for WiMAX DoS attacks in Chapter 2.

During the process of developing this research, a set of milestone targets was identified. The main aims and targets of the research have been discussed and selected as shown in Table 1.5.1. In every target, different sources and processes were required; these have been identified in the third column of the table. Throughout the duration of the research, each of the targets was justified and explained with evidence and sometimes with manual calculation or prototypes (mainly in the case of simulation). For example, the use of Excel was justified because of the pre-examination of ZRSA and Brute Force attack before implementing the algorithm in the Network Security Simulator. Table 1.1 encompasses the agreed strands to achieve or to follow a sound research methodology for this project.

**Table 1.5.1:** Thesis Research Methodology

#	Target	How to get to the target?	What is required?	Justification
1	Undertake a literature review	Electronic resources; Journals; Books; Listening/speaking to experts	Keywords such as Wireless Networks (such as WiMAX 802.16 standard and LTE), security threats to WiMAX, cryptosystems, RSA and 3-DES algorithms	To gain knowledge/expertise in the subject area. To decide how the project could best be implemented and if it is feasible given the time frame
2	<b>Decision:</b> WiMAX or LTE?	Which one is based on IEEE standard?	WiMAX	To design and develop a security simulator around WiMAX within the designated time
3	Understand the WiMAX standard 802.16e network structure and capabilities	IEEE standard 802.16 Local and metropolitan area networks	WiMAX 802.16e standard manual, Publications on WiMAX via papers or articles	To help to: i) Find and define solutions with security weaknesses & ii) Design and develop a wireless security simulator
4	Improve the Encryption/Decryption Algorithm	Define the required mathematical formulae behind RSA	Define RSA weaknesses	Develop a new Encryption/ Decryption Algorithm (ZRSA)

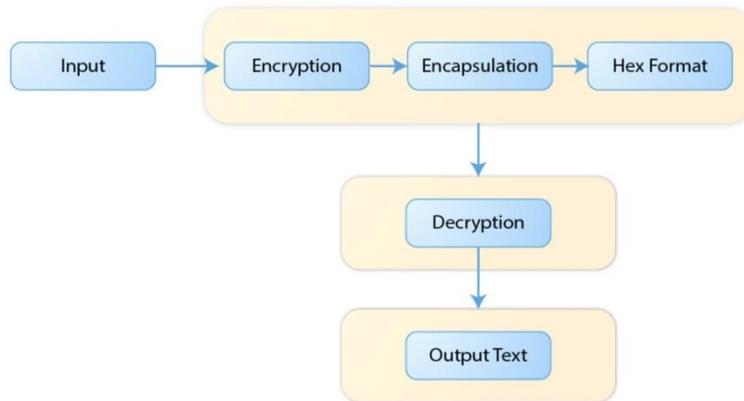
5	Evaluate ZRSA	Define BFA	Excel	To be applied on RSA and ZRSA
6	Tackle DoS attacks	Review the processes of Signal Initialisation (SI) and Management Messages (MM) of WiMAX	Define the SI & MM weaknesses	Suggest new processes to eliminate DoS attacks: i) Via SI ii) Via Pre-registration directory
7	<b>Decision:</b> Include in the simulator i) SI solution OR ii) Pre-registration directory	Which one is software dependent?	SI	To design and develop a new SI and include it in the security simulator
8	Familiarisation on Microsoft Visual Studio 2012 platform	i) Making use of tutorials and manual provided by the developers & ii) Trial and error	i) Microsoft Visual Studio 2012 version ii) Basic knowledge of using the software	Unless the software is tried and tested, errors could potentially arise once debugging; and this can be time consuming
9	Learn and practice on the software tool	Understand how to program using C# languages in the software tool to design applications	i) PC with Microsoft Visual Studio 2012 version installed & ii) C# programming materials	This will help to save time when implementing the designed simulator
10	Design and develop WiMAX standard security simulator	Determine the required security parameters to introduce into simulator	Knowing the possible security threats and attacks to the network	Designing the parameters of the security simulator stand as initial stage in this project
11	Implement the simulator using C# programming language	Creating a Windows forms application using Microsoft Visual Studio	Knowledge of Microsoft Visual Studio software and C# programming language	C# programming language was approved to be used for this project
12	Test and verify if the developed simulator output functions are as expected	Debugging the Implemented Windows forms application	Observing and testing the behaviour of the simulator. Making changes if required	Testing and verifying always prove to help achieve the best outcome results

13	Assist, evaluate the results	Setting standards for the simulator to meet	Assessing the designed simulator against the required standards	This is to help achieve the best results of the project
----	------------------------------	---	---	---

### 1.5.1 Simulation Model

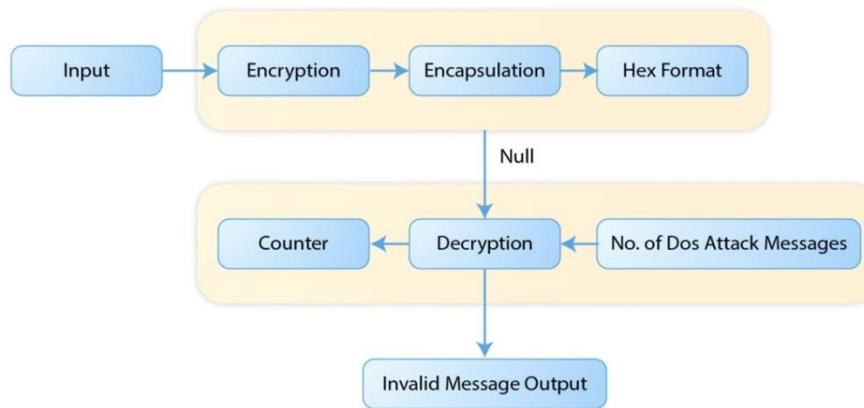
The following simulation models were used for the implementation of the two scenarios:

1. RSA without DoS Attack Model: The scenario is implemented with an input textbox which accepts all characters, encrypts the input and displays the encrypted text in a textbox, shows the encapsulation in the 802.16e WiMAX standard and displays the HEX format of the encrypted text. The encrypted text is then passed to a decryption method which decrypts the encrypted text to produce the plain text output which is displayed in an output textbox. The model is shown in Figure 1.5.1.1



**Figure 1.5.1.1:** Simulation Model for RSA without DoS Attack

2. RSA with DoS Attack Model: In this scenario the input, which accepts all character types, accepts an input text which is encrypted, shows the encapsulation of the encrypted text in 802.16e WiMAX standard format, and displays the HEX format of the encrypted text. On decryption a null value is passed on from the decrypt method and, based on the number of DoS attack messages specified for the system, processes that number of messages before throwing an invalid exception signifying timeout and no service received. This is shown in Figure 1.5.1.2



**Figure 1.5.1.2:** Simulation Model for ZA with DoS Attack

## 1.6 Report Guidance

The report structure is as follows:

1. Chapter 1: Introduction – The general history of the 802.16 standard is mentioned along with the “Problem Definition” under the Overview section. The research methodology which represents the road map to the future of this research project is laid down in the second section. Two simulation models at block level have been described in the third section.
2. Chapter 2: WiMAX Technology – The technology behind the 802.16e WiMAX standard specification is discussed in depth in this chapter including the security sub layer, the encapsulation standard and the authentication/authorisation specification.
3. Chapter 3: The Proposed Algorithm: Zaabi Security Algorithm (Zaabi Algorithm or ZA) – Presents and suggests a universal solution to the problem revealed and defined in Chapter 1. – A novel approach to securing the management messages in IEEE 802.16e protocols. The proposed solution is called Zaabi Algorithm. It is aimed at tackling all types of known attacks on WiMAX networks at the Transport and Application layers.
4. Chapter 4: The Certificate Authority Tools of the Proposed Algorithm: Zaabi Algorithm - The details of the Certificate Authority tools of the proposed algorithm, Z Algorithm are presented in this chapter.
5. Chapter 5: How to implement the Proposed Architecture and likely contribution to knowledge - Addresses the implementation issues of the Proposed Architecture. This Chapter also lists and discusses the likely contribution to knowledge.
6. Chapter 6 Conclusion – Is the conclusion of the report.

## 1.7 Summary

An overview of the thesis has been given in this Chapter. The chapter, as well, looked at the history of 802.16 technologies. This forms a background for the reader. The research has adopted a firm methodology with reasonable expectation with the given resources. Section 1.4, discussed the research methodology used and tabulated 4 items; i) Target, ii) How to get to the target? iii) What is required? and iv) Justification. As a matter of preparation, two simulation models were discussed in section 1.4.1; one without attack and the second one with attack. A brief report-guidance has been introduced in section 1.5.

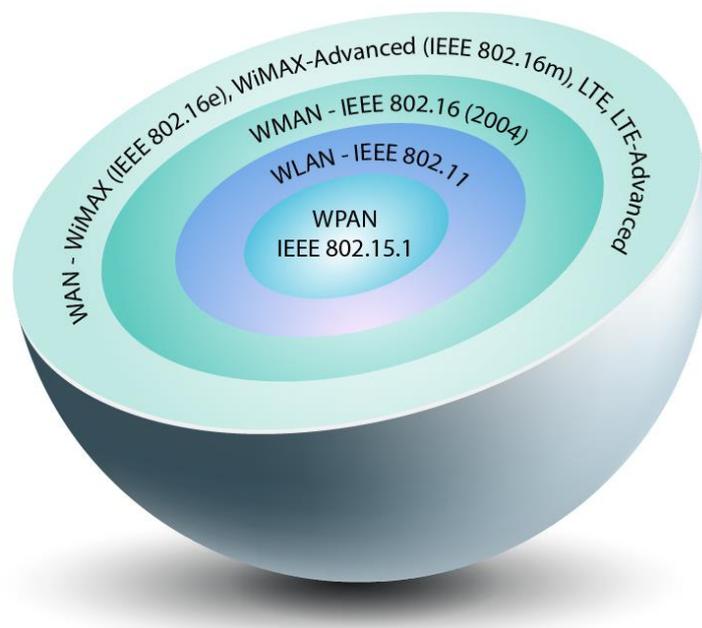
## Chapter 2: Wireless Data Networks

Wireless networking, whether it is 2G, 3G or 4G has been in use over the last three decades. It started with 2G and reached the most promising version, 4G. WiMAX-Advanced and LTE-Advanced protocol standards are supporting 4G and possibly the upcoming 5G

There are various types of network protocols that have been standardised internationally. A simple classification chart is presented in section 2.1. A brief history of wireless network technology is presented in section 2.2. This section detailed WiBro, WiMAX, GSM, LTE, WiMAX- Advanced and LTE- Advanced. The security architecture and features of each wireless network protocol have been released for GSM, WiMAX and LTE. The features of the security setup for these protocols do have common ground and share the same problems faced by users with hackers. The chapter ends with a brief summary.

### 2.1 Network Types

All existing and approved network protocols (Nuaymi, 2007) have been categorized in WAN, WMAN, WLAN and WPAN. The illustration shown in Figure 2.1.1 is relative to the distance coverage.



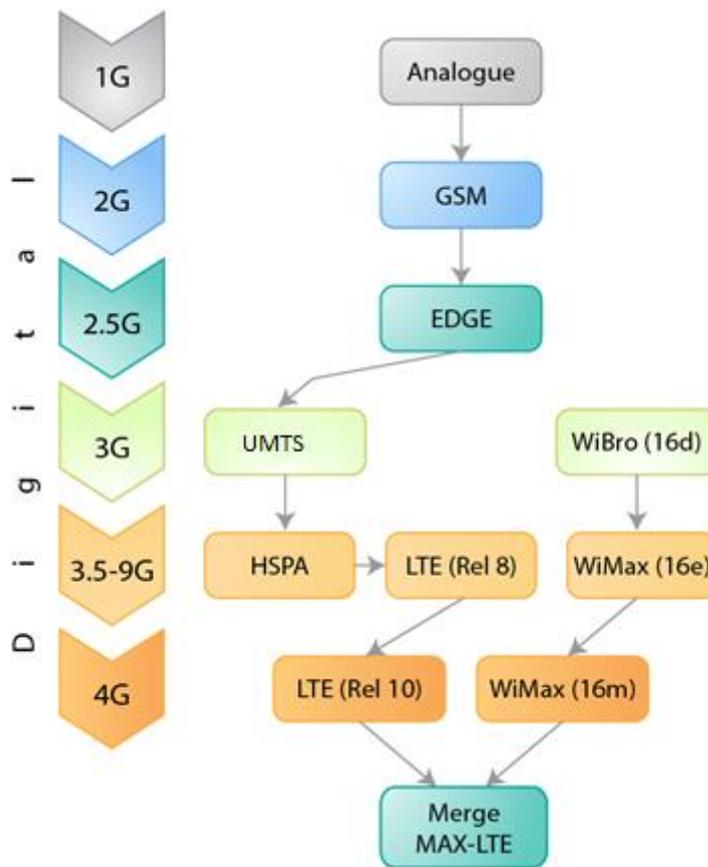
**Figure 2.1.1: Network Types**

An example of WPAN is Bluetooth, WLAN is WiFi, WMAN is WiBro and WAN is WiMAX (IEEE 802.16e).

## 2.2 Wireless Network Technology Brief History

Within the last three decades, the market launched four network technologies that deploy wireless communication protocols of data, namely WiBro, WiMAX, GSM and LTE. The following sections present a brief history to the four emerged wireless technologies (Kumar, et al., 2010) (Shukla, et al., 2013).

Figure 2.2.1 lists the evolution of the wireless technology from 1G to 4G.



**Figure 2.2.1:** Evolution of Wireless Standards

The progress from 1G to 4G was marked with millstones achievements and features. They are summarised below:

- 1G (1970): Based on Advanced Mobile Phone System (AMPS), Total Access Communication Systems (TACS) and Nordic Mobile Telephone (NMT) standards
- 2G (1990): Is digital and based on GSM standard
- 2.5G (1998): The first time that packet-switched data capabilities using Enhanced Data rates for GSM Evolution (EDGE), have been utilised
- 3G to 3.9G (2003): Is based on Universal Mobile Telecommunications System (UMTS) to deliver global roaming, with potential access to the Internet from any location. High

Speed Packet Access (HSPA) is used to extend and improve the performance of CDMA protocols

- 4G (2012): Is meant to deliver Quality of Services (QoS) to streaming and based on WiMAX (802.16m) and LTE (Release 10) standards

As they are related to each other, WiBro, WiMAX and LTE networks were chosen and discussed below.

### 2.2.1 WiBro

WiBro (Wireless Broadband) is a customised version of the 802.16e standard which is intended to be a wirelessly PtP communication system. WiBro is the fixed version of 802.16.

In 2002, WiBro was introduced by the Korean Communication Commission (group from South Korea Telecom Industry and the South Korean government) with the intent to deliver theoretical speed of 30 Mbps over a range of up to 5Km and provide a high speed internet solution to local consumers. The WiBro commissioner recommended OFDMA technology to allow multiple-access. So, with the speed of 30 Mbps, WiBro helped to overcome the existing data rate limitations of CDMA for mobile phones.

WiBro security architecture, similar to WiMAX, is based on IEEE 802.16e as it has two basic protocols; an encryption protocol (EP) and a privacy key management (PKM) protocol (Boavida, 2007).

One of the differences between WiBro and WiMAX is related to the MAC layer hand-offs. Though this difference is considered minor, it is suitable for simulation purposes. That means another version of WiMAX core has to be added to the simulator under development within this thesis. The discussion on the developed simulator is furnished in chapter 5.

In early 2000, Korea was enjoying the most extensive 3G deployments in the world, and its fixed broadband access per capita was the highest in the world. An improved mobile broadband was needed. As a result, WiBro was phased out and WiMAX mobile took over (WiMAX, 2015).

### 2.2.2 WiMAX

WiMAX (Worldwide Interoperability for Microwave Access) is a wireless communication protocol standardized by the IEEE. WiMAX was published as IEEE 802.16e Release 1 and later as IEEE 802.16m in Release 2. It was designed as an alternative to cable and DSL. Though, currently WiMAX is a niche market, the published standard by IEEE makes an ideal protocol to work with. Hence, WiMAX forms the basis of the research of this project.

### 2.2.2.1 WiMAX Protocols

This subsection looks at WiMAX technology, its security specification, the standard encapsulation format and the authentication specification. It is a thorough research study on WiMAX because it forms the main theme of the project of this thesis.

### 2.2.2.2 Overview

WiMAX 802.16 is an IEEE standard for high data transmission of up to 4Mbps for uplink and 46Mbps for downlink using Orthogonal Frequency Division Multiple Access (OFDMA) for uplink and downlink. It is purely based on IP. Security of 802.16e is provided at the MAC layer but still has some loopholes in the initial entry as the management messages are not encrypted. One of the objectives of this project is to stop DoS attacks at the MAC layer by a) analysis of user authentication and b) enhancement of the security at the entry point. One way to achieve this is by introducing encryption of the management messages through the use of a shared key known to both a) and b). The generated shared key could be achieved by a function based on the MAC address of the Subscriber Station (SS) or could be stored initially in the devices and the database maintained at the BS end for secure authentication. The WiMAX layout is given in Figure 2.2.2.2.1.

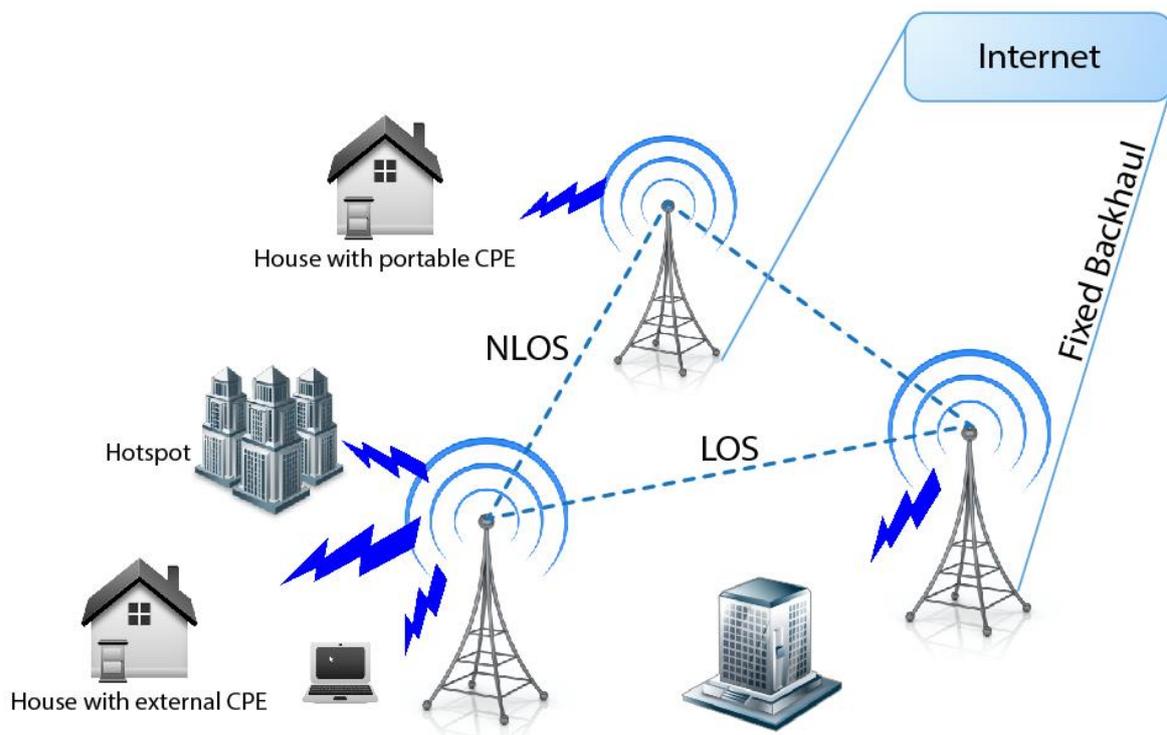


Figure 2.2.2.2.1: WiMAX Network

An experiment, conducted by Hong (2011) to simulate DoS attacks on WiMAX networks, attempted to exploit the vulnerabilities of the RNG-RSP messages to DoS attack. Due to the success of the experimental attack, it was concluded that there was still room for improvement to the security of the IEEE802.16e standard.

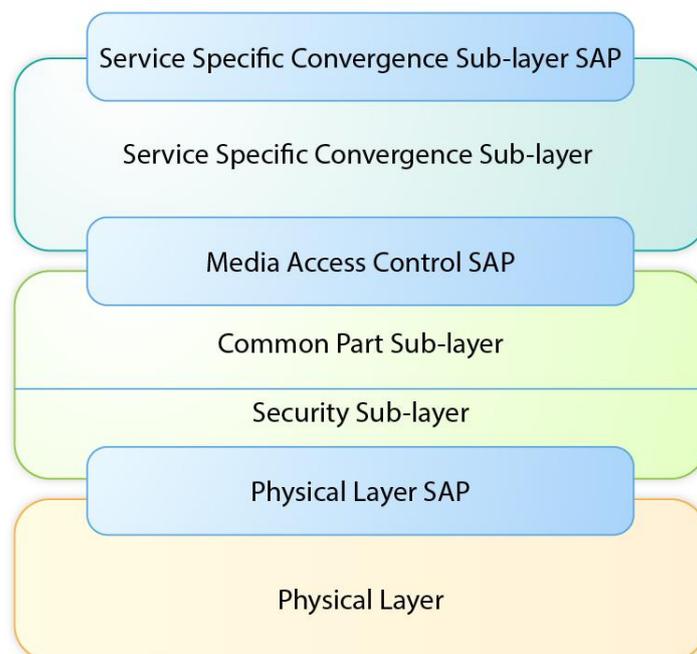
In an assessment of WiMAX security that was conducted by Sanjay P. Ahuja (2010), DoS attacks was listed as one of the main threats to WiMAX networks.

Shikha (2013), in their review of WiMAX security threats, identified DoS attacks as one of the threats that needed to be addressed to safeguard WiMAX networks.

Hence, there is enough proof that DoS forms a major vulnerability issue to the WiMAX standard.

### 2.2.2.3 Security Sub-Layer

In the 802.16e version of the WiMAX technology the Physical (MAC) layer is divided into sub-layers as shown in Figure 2.2.2.3.1.



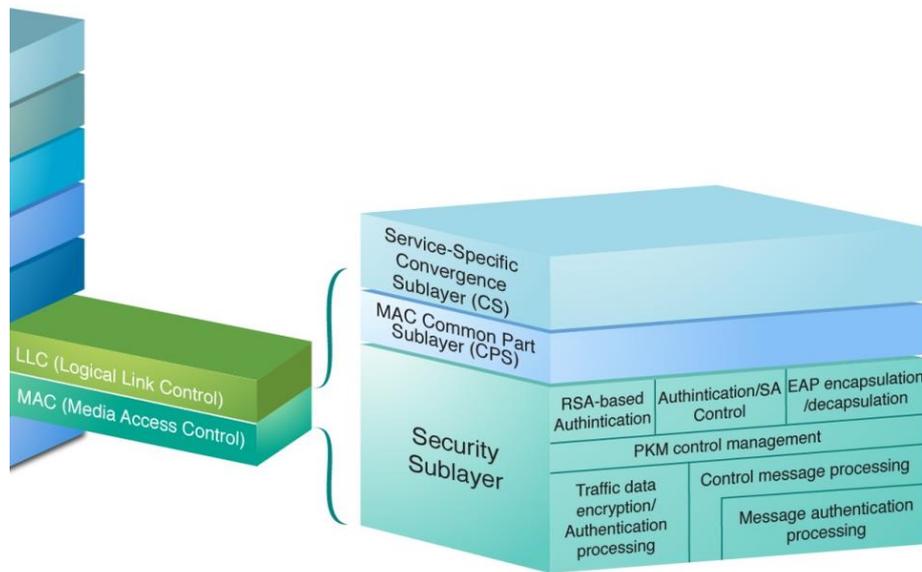
**Figure 2.2.2.3.1:** MAC & Physical layer of 802.16 showing MAC Privacy Sub-layer (Ulvan, et al., 2009)

The Physical (MAC) layer is divided into three sub-layers:

- The Service Specific Convergence Sub-layer (CS): This layer is concerned with mapping Service Data Units (SDUs) to appropriate service flows, bandwidth allocation and serves as a point of entry between the MAC layer and the upper layers.

- The Common Part Sub-layer (CPS): This sub-layer is concerned with the encapsulation of SDUs to form MAC Protocol Data Units (PDUs) as required.
- The Privacy/Security Sub-layer: This layer supports the security protocols to be used in authentication, encryption and key exchange.

The Security Sub-layer within the MAC layer supports various protocols and algorithms as shown in Figure 2.2.2.3.2.



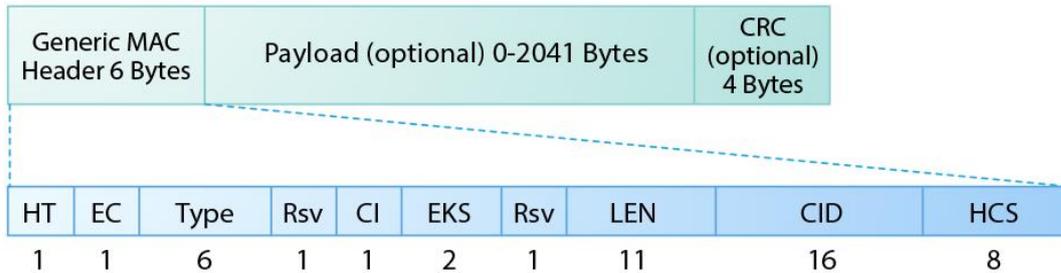
**Figure 2.2.2.3.2: Security Sub-layer**

All MAC PDUs pass through the Security Sub-layer before going to the Physical layer and the first entry point into the MAC layer from the Physical layer.

Both Authentication/Security Association (SA) control and Extensible Authentication Protocol (EAP) encapsulation/de-encapsulation are RSA based.

#### 2.2.2.4 WiMAX MAC – Protocol Data Unit (PDU)

The WiMAX MAC PDU is made up of a maximum size of 2051 bytes which comprises of a Header of 6 Bytes, Payload ranging from 0-2041 Bytes and a Cyclic Redundancy Check (CRC) trailer which is 4 Bytes in size (Nuaymi, 2007). The generic format is given in Figure 2.2.2.4.1.



**Figure 2.2.2.4.1: Generic MAC PDU**

The description for the various header fields is given as follows:

**HT** – Header Type: it is made up of a single bit and set to 0 for Generic MAC Header.

**EC** – Encryption Control Field: This indicates whether the payload is encrypted or not. It is set to 1 when the payload is encrypted and to 0 when it is not encrypted.

**Type** – this field indicates the sub-headers and special payload type up to five if present in the payload.

**Rsv** – Reserved: This field consists of one bit

**ESF** – Extended Sub-Header Fields: This field indicates if there is an extended sub-header present following the Generic MAC Header, it is of 1 bit size and is applicable to both uplink and downlink.

**CI** – CRC indicator: this field indicates if CRC is present or not, it is set to 1 if present and to 0 if CRC is not present.

**EKS** – Encryption Key Sequence: this field depends on the Encryption control field, if it is set to 1 then the EKS defines the sequential index of Traffic Encryption Keys (TEKs) and initialization vectors used to encrypt the payload, it is 2 bits in length.

**LEN** – Length: it specifies the length of the MAC PDU including the CRC if present and the Header, it is 11 bits in length.

**CID** – Connection Identifier: it represents the connection identifier of the user and is 16 bits long.

**HCS** – Header Check Sequence: it is used to detect errors in the header and is 8 bits long.

The Management messages are sent in MAC PDU with a Generic MAC Header followed by a management message type field of 1 byte, it may be followed by a CRC. Figure 2.2.2.4.2 shows the management message PDU format.



**Figure 2.2.2.4.2:** Generic MAC PDU

The MAC Management messages can neither be fragmented nor packed on the ranging connections, broadcast and basic connection.

### 2.2.2.5 WiMAX IEEE 802.16e Authentication/Authorisation Protocol

The initial connection, which is prior to the authentication and authorisation, is done by the exchange of MAC management messages for synchronisation. The management messages are defined and carried in the payload of the MAC PDU. After the initial synchronisation the authentication and authorisation protocols come into the picture.

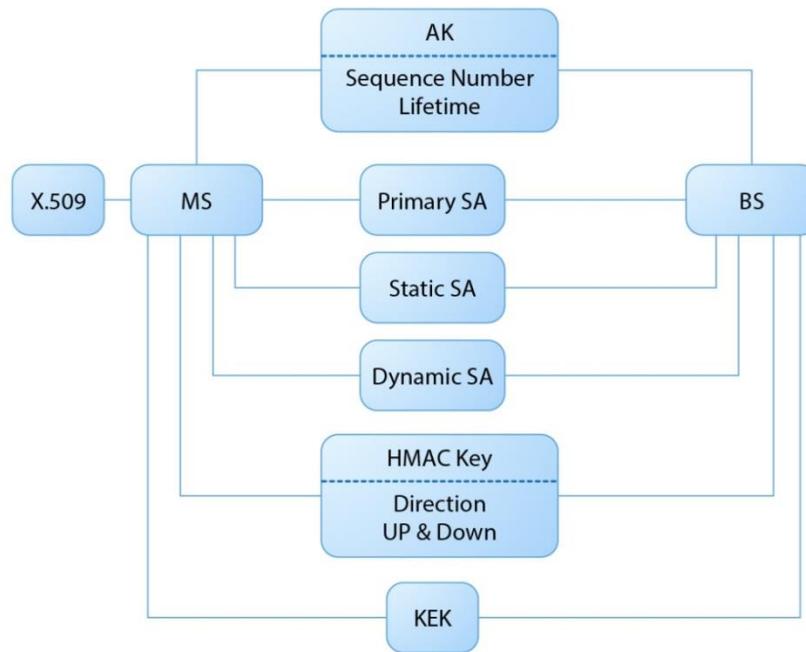
#### 2.2.2.5.1 Privacy Key Management (PKM) Protocol

With the Privacy Key Management (PKM) Protocol version 2, there is mutual authentication between the base station and the mobile subscriber, while in version 1 there is only unilateral authentication of the mobile subscriber to the base station. The protocol also supports periodic re-authentication, re-authorisation and key refreshing.

This project looks at the RSA authentication along with PKM. Other protocols like EAP and RSA authentication followed by the EAP authentication already exist. A shared secret is shared between the MS and the BS called the Authorisation Key (AK). AK is used by both the base station and mobile subscriber to generate the Key Encryption Key (KEK), which is used to encrypt further key exchanges. At this point the base station sends a Traffic Encryption Keys (TEK) challenge message to the MS which responds by activating the TEK state machine and sends a TEK request message to the base station. The base station responds by generating two TEK keys and sending them to the MS, the MS uses the TEK keys for sending data to the BS. The MS is responsible for timely refreshing keys as the TEK keys have a life based on time, this is ensured by activated TEK state machines.

The X.509 certificate is used to authenticate a subscriber identity and is associated to the data services.

A security model at a block level, as shown in Figure 2.2.2.5.1.1 to reflect the above steps, has been published by (Barbeau, 2005).



**Figure 2.2.2.5.1.1: Security Key**

### 2.2.2.5.2 PKM v1

A security association (SA) refers to the security information that a BS and an SS share in order to enable secure communication between them. There are three types of security associations in the IEEE802.16 specification: Primary SA, Static SA and Dynamic SA, as shown in figure 2.2.2.5.1.1 above. The SS establishes a primary security association during the SS's initialisation. Static SAs are prepared within the BS (IEEE, 2005). Dynamic Security Associations are made and terminated dynamically as the need arises in relation to the initiation and termination of service flows. Static and Dynamic SAs can be shared by multiple SSs.

The security association information that is shared between the BS and MS includes the cryptographic suite that is employed and the SA may include TEKs and initialization vectors. The specific content of the SA depends on the cryptographic suite used. Security associations SAs are identified by means of Security Association IDentities (SAIDs). The requirement is that every SS establishes a primary SA with a BS and the SAID of this association is equal to the Basic Connection Identifier (CID) of that SS (IEEE, 2005).

The SS requests keying material from the BS. This keying material might include the encryption standard and the initialization vector with a lifetime. The SS is responsible for requesting, from the BS, fresh keying material before the current keying material expires; if the current keying material expires the SS has to perform network entry again.

In some cryptographic suites, key lifetime depends on the exhaustion rate of a number space, in this scenario the keying material expires either when the lifetime expires or when the number space is exhausted whichever comes first.

#### 2.2.2.5.3 PKM RSA Authentication

The IETF RFC 3280, X.509 digital certificate and the RSA algorithm (PKCS#1) are used in the PKM RSA Authentication (Galbraith & Saarenmaa, 2005).

The client SS is authenticated during the initial authentication process via the X.509 certificate issued by the SS's manufacturer. The X.509 certificate contains the SS's MAC address and public key information, when the SS requests an AK, it presents its certificate to the BS. The BS verifies the X.509 certificate and encrypts the AK which it sends to the SS.

SSs using RSA authentication shall have an algorithm to install the public and private key pairs and also an algorithm that installs the X.509 certificate prior to AK request.

SSs that have factory installed RSA key pairs will also have a factory installed X.509 certificate.

The PKM protocols are of two types

- Privacy Key Management Version 1(PKM V1)
- Privacy Key Management Version 2(PKM V2)

The PKM v2 differs from PKM v1 by the following

- Mutual authentication of MS and BS
- Advanced Encryption Standard (AES) key derivation functions in contrast to Data Encryption Standard (DES) in PKM v1

The PKM v1 supports PKM-RSA authentication using X.509 certificate. With the IEEE802.16e amendment, PKM v2 protocol was added with support for Privacy Key Management – Extensible Authentication Protocol (PKM-EAP). The PKM-RSA alone is available in PKM v1 and PKM v2 supports both PKM-RSA and PKM-EAP as optional.

#### 2.2.2.5.4 Authorisation via RSA Authentication Protocol

The authorisation process involves sending the authentication message, which involves the X.509 certificate of the SS, to the BS. The authentication message is purely informative and the BS may decide to ignore the message. The SS immediately follows up with an

authorisation request message which the BS does not ignore and it contains the X.509 certificate of the SS, a request for AK and the SAIDs for the SS. The authorisation message includes the following:

- Manufacturer issued X.509 certificate
- The specific cryptographic algorithm supported by the SS and its capabilities is given to the BS in the form of cryptographic suite identifiers which describes the encryption and authentication of packet data that it supports
- The basic connection identifier CID is the first static CID assigned to the MS by the BS during the initial ranging

The BS sends an authorisation reply message to the MS after it verifies the identity of the SS and the cryptographic algorithm supported by the SS. The authorisation reply message contains:

- Authorisation Key encrypted by the SS's public key
- A sequence number which is of 4 bit length and used to identify the AK
- The lifetime
- The SAIDs and properties of primary or more static SAs for which the SS is authorised to have keying material

The authorisation reply message shall not identify any dynamic security associations (SAs) and also determine if the SS is authorised for basic unicast services and any additional statistically provisioned services (SAIDs) the subscriber has subscribed for. The protected service a BS provides to an SS depends on the cryptographic suites that both the MS and BS mutually support.

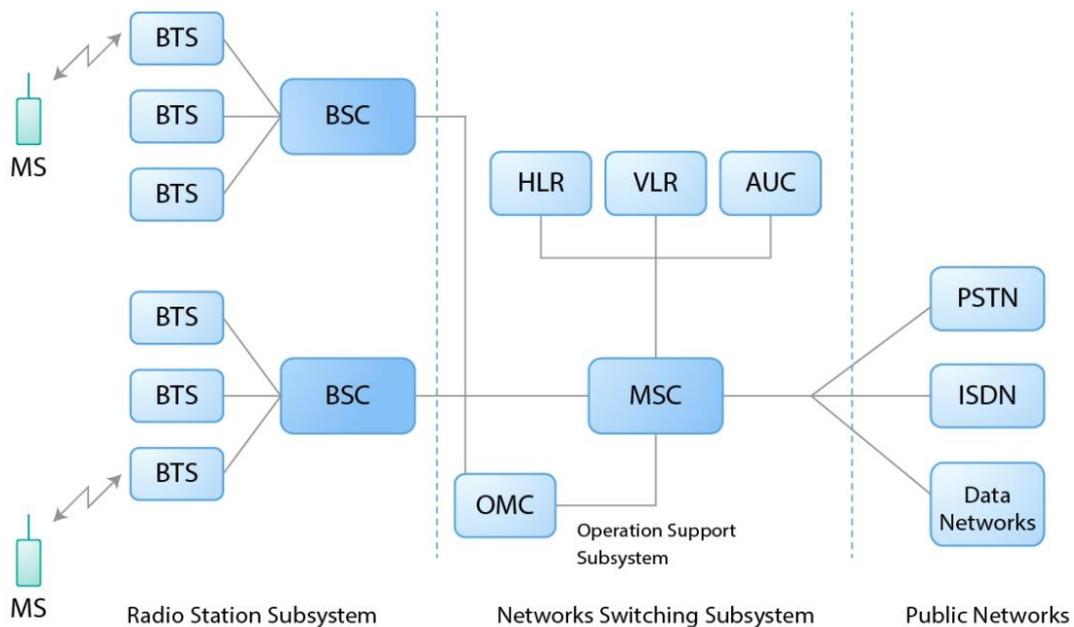
### 2.2.3 Global System for Mobile Communication (GSM)

GSM (Global system for Mobile communication) technology has been developed by ETSI (European Telecommunications Standards Institute) in July 1991. GSM is a frequency and time division system. GSM is a 2G cellular network developed to replace the analogue cellular networks known as 1G as a result of the increasing public demand to boost the capacity of the network with additional services to SMS and subscriber ID as well as reducing the cost.

Similar in requirement to the analogue generation, GSM is a circuit-switch based technology originally developed to support telephone voice communication. The development of the GSM has been continuously expanded to encompass wider frequency bands and to support data communication (Schille, 1999).

### 2.2.3.1 GSM Architecture

Though the GSM system was developed to mainly provide the user with a voice communication service, ongoing demand has driven GSM to provide more services such as data communication. Call handling, subscriber identity authentication, emergency calls, signalling information element confidentiality, voice service and short message service are all functions provided by the GSM network. Figure 2.2.3.1.1 shows the architecture of GSM system (Schille, 1999).



**Figure 2.2.3.1.1: GSM System Architecture**

GSM is a Circuit-switched technology as it is basically implementing a communication channel in which the communication between two devices takes place. The GSM system consists of major components and subsystems such as: Mobile Station (MS), Base Station (BS), Mobile Switching Centre (MSC) and the Location Registers (Visitor Location Register (VLR) and Home Location Register (HLR)) (Schille, 1999).

Mobile Station (MS) is a typical mobile device such as phone, Laptop etc. Mobile devices access and benefit from GSM features such as Terminal Adapter (TA), Terminal Equipment (TE) and Subscriber Identity Module (SIM).

Base station (BS) provides radio coverage to a specific area to receive and communicate with MS. BS is divided into two functions. First is Base Station controller (BSC) which provides the control function and second is Base Transmitter Station (BTS) to provide the radio transmitting function.

Mobile switching Centre (MSC) is the switching operation that is responsible for determining and providing the radio coverage needed by MS. MSC determines various locations of MS within the radio coverage area. MSC also manages the amount of the service needed by MS.

Home location register (HLR) is a data base consisting of the information of the subscribers. HLR is responsible for storing the International Mobile Subscriber Identity (IMSI), the mobile station ID serial number and the VLR address. The main responsibility of the HLR is to manage the subscribers of the network.

Visitor Location Register (VLR) is responsible for storing the subscriber location. In the case of a roaming mobile presence in an MSC area, VLR is warned by the MSC and ordered to register and store the roaming mobile. VLR also associates with HLR and provides information such as; mobile station roaming number, temporary mobile station identity and the location of the mobile presence area.

#### 2.2.3.2 GSM Security Algorithms

GSM systems use a set of algorithms (A3, A8 & A5) to provide security for information transmitted over the air:

- a) A3: is an authentication algorithm to authenticate the subscribers and allow connections between MS and BS
- b) A8: is a key generator algorithm and
- c) A5: is a cipher/decipher algorithm.

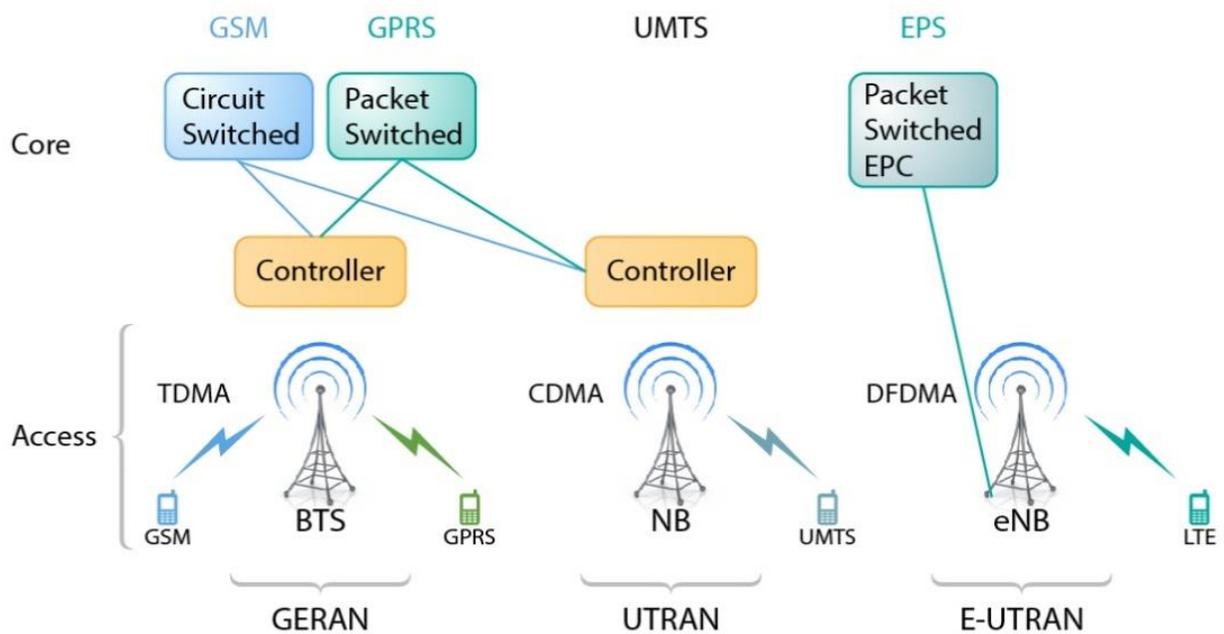
#### 2.2.4 Long Term Evolution (LTE)

With the new smart phones and the wide range of provided functions, services such as; voice calls and SMS became basic and hence data transmission enhancement became the main demand of mobile customers. Therefore, expanding the frequency bands and allowing more over air data transmission have been a serious demand of network customers around the world (Dahlman, May 10th 2011).

Hence, many organizations have made efforts to achieve the continuous customers' demands. In late 2009, 3GPP announced specifications and standards of a new over air interface technology called Long Term Evolution (LTE) or Evolved Universal Terrestrial Access Network (E-UTRAN). LTE is an extremely flexible radio interface technology that has been developed by 3GPP. LTE is a result of developing and enhancing the previous cellular network system, GSM. It uses orthogonal frequency division multiplexing (OFDM) for downlink and single carrier frequency division multiple access (SC-FDMA) for uplink. In 3GPP first LTE

release (Release 8), LTE provides a peak rates of 300 Mb/s as well as less than 5ms radio network delay. Moreover, enhancing the spectrum efficiency and reducing the cost have made LTE an excellent technology. LTE utilizes a number of systems such as; frequency division duplex (FDD) as well as time division duplex (TDD).

LTE is based on GSM. 3GPP enhanced and developed its first LTE release (Release 8) and introduced a series of additional releases. Release 9, 10 (LTE-Advanced) and further releases such as 11, 12, and the current 13 and 14 releases are all a result of developing the specifications of LTE. However, unlike GSM, LTE is an IP based packet switched technology that supports evolved packet system (EPS). Figure 2.2.4.1 illustrates the developed solutions from GSM to LTE.

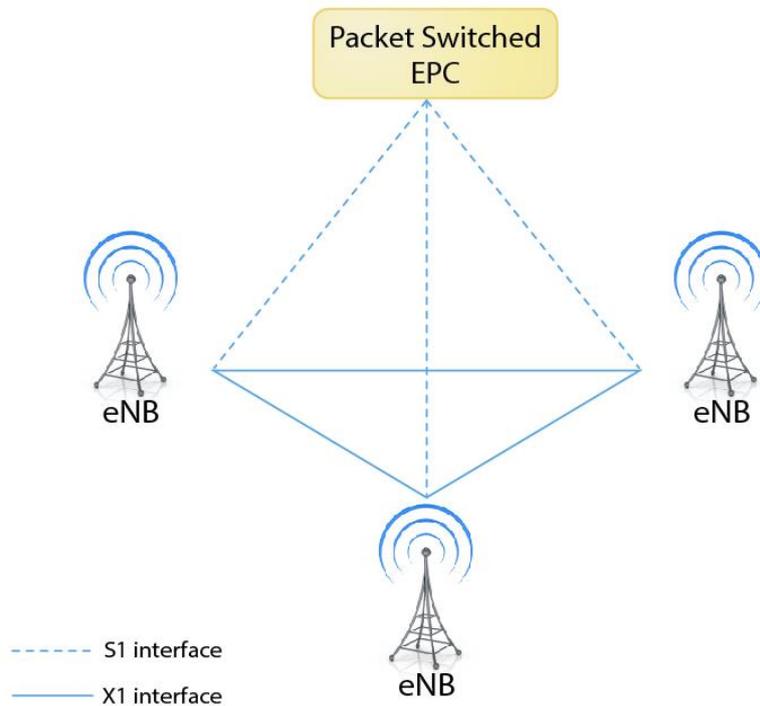


**Figure 2.2.4.1:** Network Solutions from GSM to LTE

### 2.2.4.1 LTE and LTE Radio Interface architectures

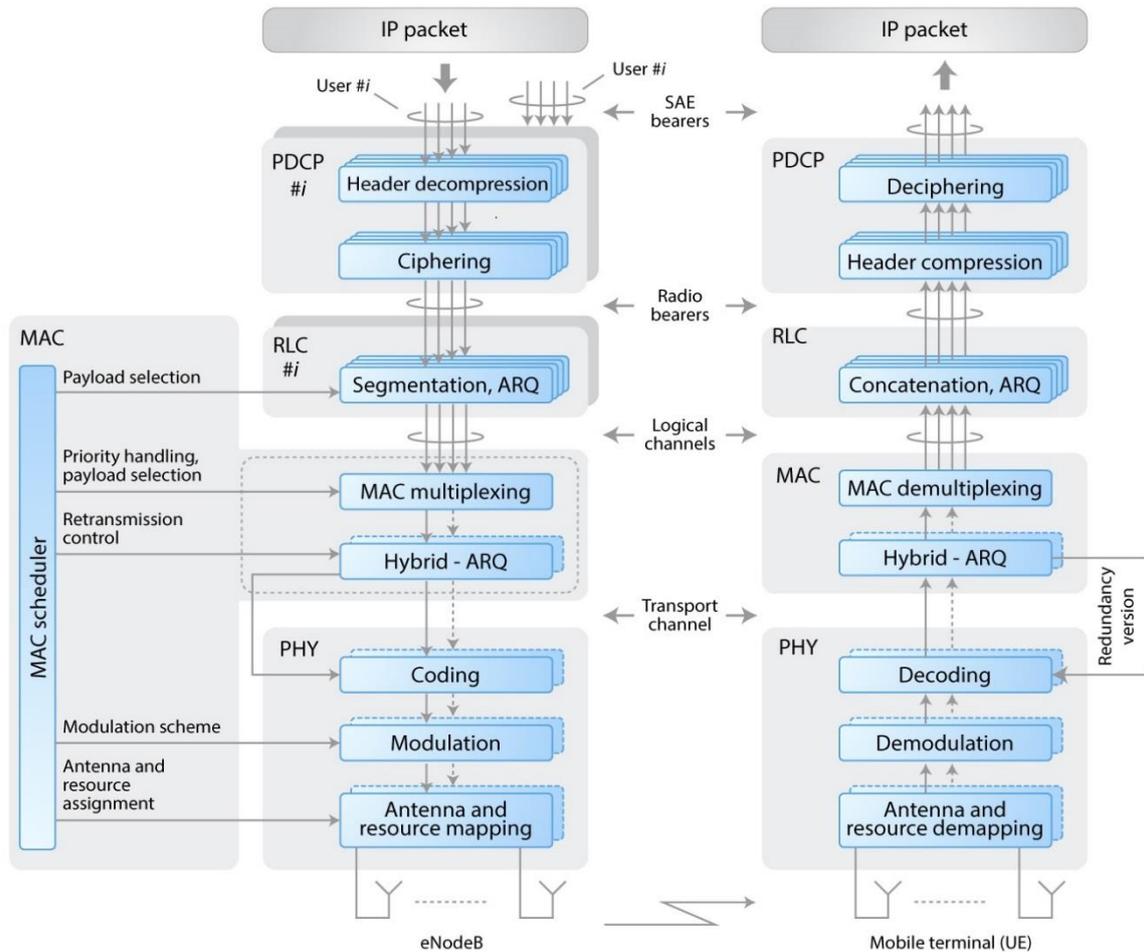
In a general and simple way, an LTE network consists of number of base stations (eNB) that are connected with each other through an interface called (X2). In LTE network there is no intelligent controller. Intelligence is distributed between eNBs. Intelligence distribution plays a main role in speeding the connection in LTE network. Using distributed intelligence, solves the problem of handover delay. Interaction between eNBs and EPC takes place through an

interface called (S1). Figure 2.2.4.1.1 illustrates an overall LTE architecture (Research, 3GPP Technologies, 2011).



**Figure 2.2.4.1.1: Overall LTE Architecture**

Since, the coding/encoding processes take place in the physical layers of the LTE radio interface and the focus of this research is the security algorithms used in LTE, Figure 2.2.4.1.2 illustrates the architecture of the LTE radio interface at protocol layers level.



**Figure 2.2.4.1.2: LTE Radio Interface Architecture**

Table 2.2.4.1.1 specifies each layer and the service it performs.

**Table 2.2.4.1.1: LTE Radio Interface Layers**

<b>Radio Interface Layers</b>	PDCP	This layer's responsibility is to reduce the transmission bit length over the radio interface by compressing the IP header. In the eNB side, the layer also provides the ciphering and integrity needed for the transmitted data. In the UE side, the layer perform decipher and decompression processes.
	RLC	This layer's responsibility is to segment the transmitted data.
	MAC	This layer's responsibility is to provide downlink/uplink scheduling and also to perform the data hybrid ARQ retransmission.

	PHY	In the eNB side, this layer is responsible for performing coding, modulation and antenna mapping processes. In the UE side, this layer provides decoding, demodulating and also antenna mapping processes. All physical functions are provided through this layer.
--	-----	--

#### 2.2.4.2 LTE Security

Similar to GSM, LTE uses A5, A3 and A8 algorithms to provide security to the functions it provides. LTE security features are summarised below and described in more details within the following section and subsections:

- Privacy: unauthorised processes are denied, hence illegitimate IMSI is not allowed
- Authentication: no masquerading by authorised user
- Data confidentiality at the Physical Layer: users are protected and prevented from any illegitimate access
- Connectionless data confidentiality: users are prevented from any illegitimate access
- Signalling information confidentiality: is intend to protect the transferred information from being accessed by illegitimates users or processes

#### 2.2.4.3 Security and Cryptography Algorithms

Cryptography algorithms used in GSM are A5, A3 and A8. These algorithms are intended to provide authentication and data protection to the user.

A5 is a ciphering/deciphering algorithm. A3 is an authentication algorithm. A8 is a cipher key generator. All three algorithms work together to provide the protection needed for the security features and functions. A minimum cryptography level is specified by the security algorithms. Without this cryptography level, the security features cannot work.

##### 2.2.4.3.1 Security Features and Functions

The following security features and functions are compulsory to be implemented in both the base station and the mobile station. Compulsory means that all GSM and mobile stations must support there security features.

###### 2.2.4.3.1.1 Subscriber Identity Confidentiality

This feature is to make sure unauthorised processes are not available to illegitimate IMSI. The feature provides for the privacy of the GSM subscribers. The main concern of this feature is preventing the subscriber Identity, location and more private information that is transferred through the radio channel from being traced. This feature plays a main role in improving the other security features and functions.

#### 2.2.4.3.1.2 Subscriber Identity Authentication

The purpose of the subscriber identity authentication feature is to ensure that the information that the IMSI or the TMSI are willing to transfer through the radio path, is the one claimed to be. The feature also protects the GSM from illegitimate use. This feature denies any unauthorised user from masquerading as an authorised user.

#### 2.2.4.3.1.3 User Data Confidentiality on Physical Connections (Voice and Non-voice)

The user Data confidentiality on physical connections feature ensures the user's information transferred through the radio path is protected and protected from any illegitimate access. The feature's main property is to keep the privacy of the users information transferred through traffic channels.

#### 2.2.4.3.1.4 Connectionless User Data Confidentiality

The connectionless user data confidentiality feature is to ensure that the users' information transferred through the signalling channel is protected from any illegitimate access. The main purpose of this feature is to make the signalling channels safe and privates for exchanging the users' Information.

#### 2.2.4.3.1.5 Signalling Information Element Confidentiality

Signalling information is the information transferred between the BS and the MS through the radio channel. The signalling information element confidentiality feature is intended to protect the transferred information from being accessed by illegitimates users or processes.

### 2.2.4.4 Authentication and Cipher Key Generator Algorithms

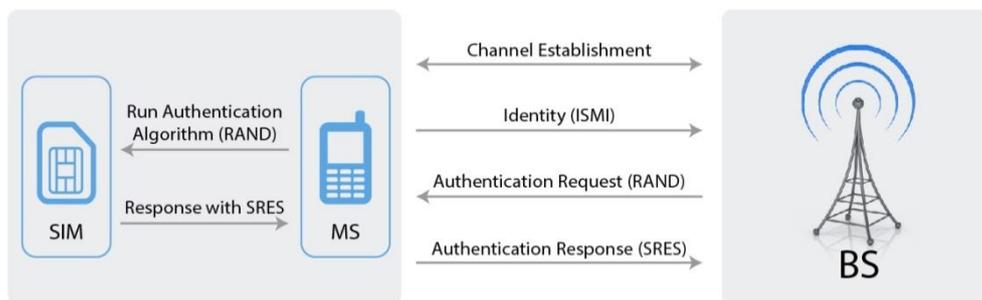
As mentioned earlier, the cryptography algorithms A3, A8 and A5 work all together to provide the protection for the security features and functions. Based on that, a fair understanding of each of the cryptography algorithms is essential to understand the other algorithms.

#### 2.2.4.4.1 A3 Authentication Algorithm

A3 authentication is the algorithm responsible for authenticating the MS. This algorithm purpose is to ensure that the subscriber is the one who is claimed to be. The authentication process follows many steps to perform the authentication of the MS. The authentication starts with a request sent from the MS to the BS. The BS sends the MS a 128 bit random number (RAND). The MS computes the RAND received from the BS with the Individual Subscriber Authentication Key (Ki) and generates a number called (SRES). The Ki is a 128-bit subscriber identification module (SIM) number which is already known by the BS. The SRES is a 32-bit signed response which is computed by the MS using the Ki. Then the MS sends the BS the SRES.

The BS investigates the SRES number sent by the MS and ensures it matches the right data. The BS investigation leads to two options. The first option is when the SRES does not match the data that exists in the BS and in this case the request is refused and the connection is terminated. The second option is when SRES matches the existing data and in this case the BS verifies the SRES and generates a 64-bit session key ( $K_c$ ) using the A8 algorithm. This  $K_c$  number is then sent to the MS in order to use it in the A5 ciphering/deciphering algorithm.

Figure 2.2.4.4.1.1 shows the operation of the communication between the MS and the BS to achieve authentication.



**Figure 2.2.4.4.1.1:** Process of Authentication between BS and MS

#### 2.2.4.4.2 A8 Cipher Key Generator Algorithm

A8 is an algorithm intended to generate a key that can be used in the A5 algorithm. As mentioned earlier in this section, understanding one algorithm requires an understanding of the other algorithms as they all work together. Basically the operation of this algorithm is related to the A3 authentication algorithm. Hence, the beginning of the A8 key generation key is already triggered with the authentication operation. A8 is the responsible algorithm for generating the session key  $K_c$ .

The session key  $K_c$  is generated by the A8 key generator algorithm in the BS after verifying the user.  $K_c$  is a 64-bit binary key that is computed by the A8 algorithm after ensuring the  $K_i$  number is verified. Handing the  $K_c$  to the user means that the user is authenticated and can start using  $K_c$  in A5 ciphering/deciphering algorithm. Using  $K_c$ , the MS can perform the ciphering/deciphering process. The  $K_c$  can be stored in both the MS and the BS. Storing the  $K_c$  in the MS allows the ciphering process to be performed. While storing the  $K_c$  in the BS allows the ciphered information (e.g. text) to be deciphered and accessed.

The BS is where the  $K_c$  is managed. Storing the  $K_c$  in the MS for the purpose of performing the ciphering/deciphering process can be done as long as the  $K_c$  is still active. Updating or deactivating the  $K_c$  by the BS means that the  $K_c$  can no longer be used in the A5

ciphering/deciphering algorithm. The process of the key generator algorithm is shown in Figure 2.2.4.4.2.1.

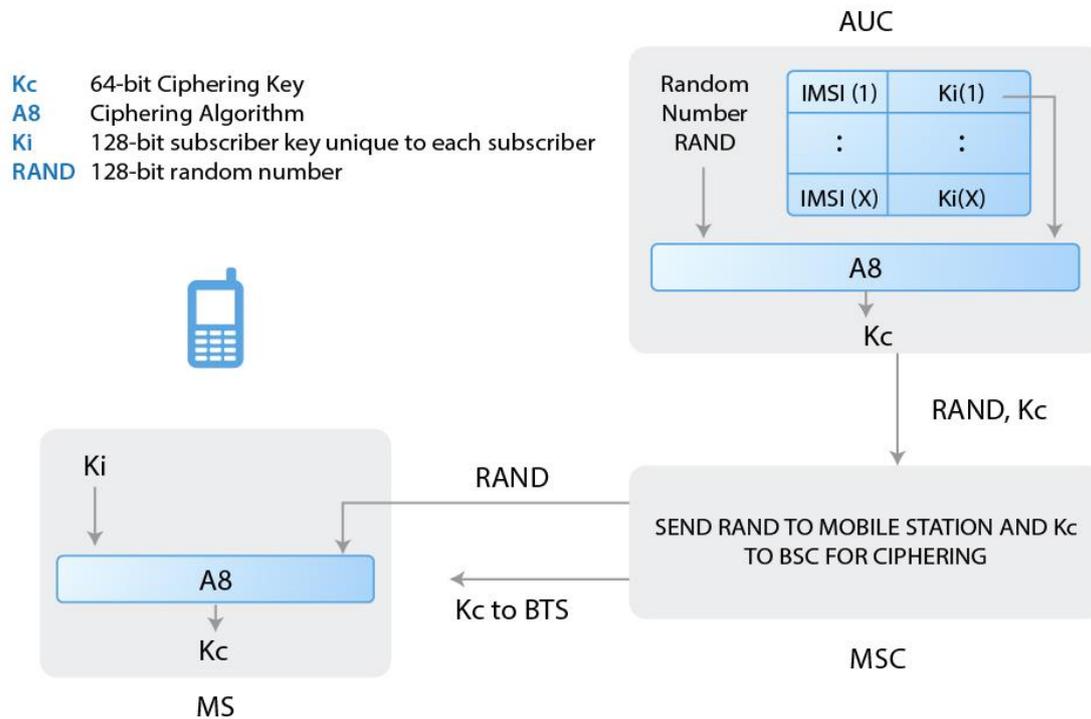
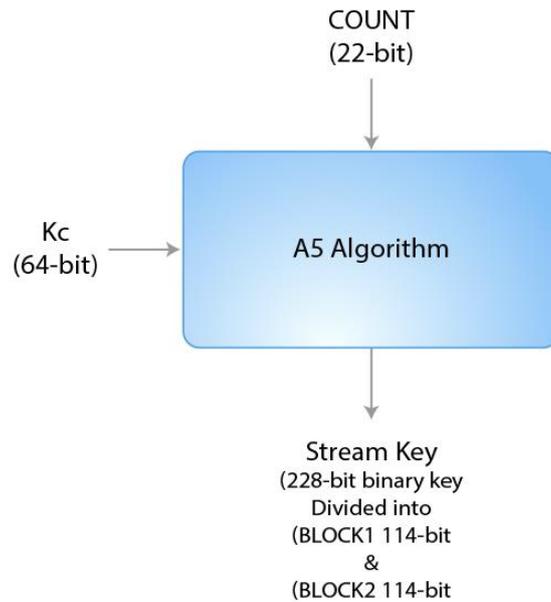


Figure 2.2.4.4.2.1: **Process of Key Generator Algorithm**

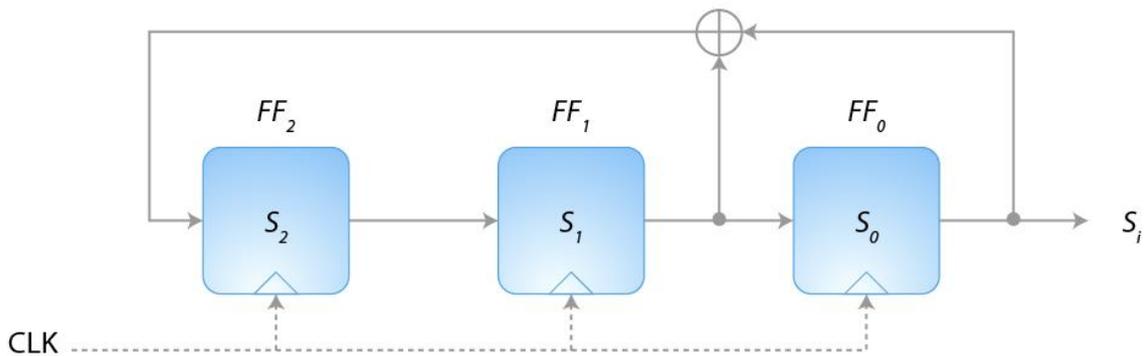
### 2.2.4.4.3 A5 Ciphering/Deciphering Algorithm

The A5 is the algorithm responsible for performing the ciphering/deciphering operation. The A5 algorithm's main responsibility is to provide encryption to the data transferred over the radio path. The A5 algorithm is implemented in both the base station (BS) and the mobile station (MS). Both ciphering and deciphering operations take place at the MS. However, triggering the ciphering/deciphering operations is managed by the BS. The BS orders the MS to perform the ciphering or deciphering.

A5 is a stream cipher based algorithm. A stream cipher is basically combining the information (e.g. plain text) with a fixed binary number resulting in ciphered information (e.g. text). This process is performed through several steps. The A5 algorithm generates a key called (Stream Key). The stream key is a result of performing several steps using the K<sub>c</sub>. The K<sub>c</sub> is processed through several steps. The K<sub>c</sub> is considered as an input to the A5 algorithm as well as a 22-bit frame number (COUNT). These two inputs are processed by the A5 algorithm using a combination of three feedback shift registers. Figure 2.2.4.4.3.1 describes the algorithm at block level along with the inputs and output parameters. Figure 2.2.4.4.3.2 shows the basic design of linear feedback shift registers.

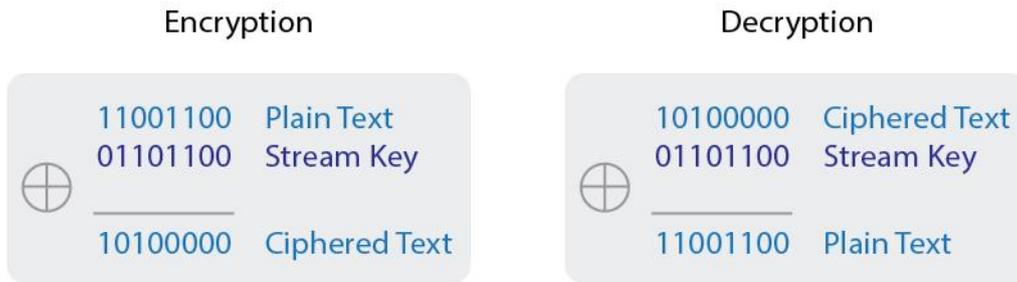


**Figure 2.2.4.4.3.1:** Inputs and output of A5 algorithm



**Figure 2.2.4.4.3.2:** Basic Design of Linear Feedback Shift Register with 3 flip-flop clocks

The A5 algorithm is a stream key generator. The stream key is a 228-bit binary number used to be computed with the information to be ciphered (e.g. plain text). The ciphering process is achieved by computing the information to be ciphered (e.g. plain text) with the stream key. The result is ciphered information. The A5 algorithm decipheres the ciphered information by computing Kc, the Stream key with the ciphered text. An example of Cipher/decipher simple text, is given in Figure 2.2.4.4.3.3.



**Figure 2.2.4.4.3.3:** Cipher/Decipher Example

## 2.2.5 Vulnerability of GSM and LTE networks

As noticed above that GSM deployed security algorithms were subsequently enhanced and deployed with LTE but vulnerabilities still exist in the systems.

### 2.2.5.1 GSM

Among a few hacker attacks, (Atkinson, 2015) has listed two of them; IMSI and SS7 attacks. Both have been briefly described below.

#### 2.2.5.1.1 GSM networks are vulnerable to IMSI detach attacks

In an article published in (Atkinson, 2015), it was listed that an **IMSI detach attack** near the Norwegian Parliament building began with suspicious mobile activity. The investigators were able to detect fake transmitters, which had the ability to register all mobile phones within their reach. The fake base stations first collect data from the mobile phone's SIM card. This allows eavesdropping of certain conversations. It will then transmit the call on to the real GSM-system, but anyone listening in can hear the entire conversation. In addition, the fake base station may allow the attacker to register SMS messages and install spyware. Some of this spyware activated the microphone to listen to conversation or the camera to monitor venues.

#### 2.2.5.1.2 SS7

The SS7 protocol suite is the approved standard by ANSI and ETSI. It is meant to control the information on public switched telephone networks. It renders the calls to fine details.

SS7 is deployed in GSM but the inherited vulnerability comes with it that allowed hackers to "track the movements of cell phone users from virtually anywhere in the world with a success rate of approximately 70%" (Timburg, 2014).

In 2014, it was reported (Atkinson, 2015) that "some subscribers on the MTS Ukraine mobile phone network were affected by suspicious/custom SS7 packets from telecom network elements with Russian addresses, causing their location and potentially the contents of their phone calls to be obtained". In this particular example, the affected subscribers were

intercepted as their calls were forwarded to a physical landline number in St Petersburg, Russia.

### 2.2.5.2 LTE

The demand for faster, cheaper and more secure network services has forced LTE to be developed. However, the security algorithms that were used in 3G have been recommended to be used in LTE. These algorithms are A3, A8 and A5 which have been described in the previous section (Banescu, 2009).

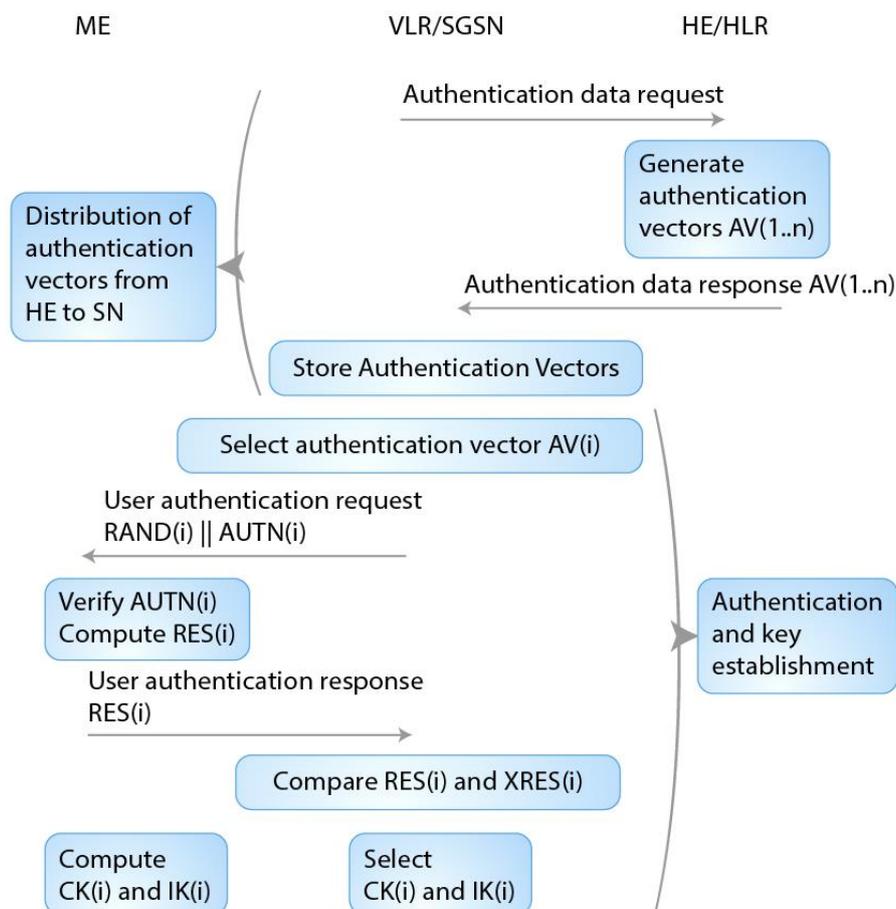
LTE/UTMS do differ with GSM in respect to:

- The extension of encryption and integrity protection coverage from the Mobile Equipment (ME) to the RNC, not at the base station (NodeB) such as the case of GSM
  - Protection of the signaling infrastructure in the core network by specifying mechanisms to allow operators to protect signaling between and within networks
  - The cryptographic keys derived on the User Services Identity Module (USIM) are longer in UMTS (128-bits) than in GSM (64-bits). Moreover the standard encryption algorithms used by 3G were openly published in order to be analyzed by a large community of experts as opposed to their predecessors
  - The definition of standard UMTS authentication algorithms, in order to avoid vulnerabilities of insecure solutions (e.g. COMP128) chosen by mobile operators
- UMTS/LTE provides a set of security features. They are classified as follows:
- *Network access security* (NAS) represents the set of security features that provide users with secure access to services, and which in particular protect against attacks on the (radio) access link
  - *Network domain security* (NDS) represents the set of security features that enable nodes to securely exchange signaling data, user data, and protect against attacks on the wire-line network
  - *User domain security* (UDS) represents the set of security features that secure access to mobile stations
  - *Application domain security* (ADS) represents the set of security features that enable applications in the user and in the provider domains to securely exchange messages
  - *Visibility and configurability of security* represents the set of features that enable the user to inform him whether a security feature is in operation or not and whether the use and provision of services should depend on the security feature

The three entities that form the security protocol (3GPP, Security architecture (Release 11), 2011) are:

- The Authentication Centre (AuC) in a user's Home Environment (HE) or Home Location Register (HLR);
- The Visitor Location Register (VLR) or the Serving GPRS Support Node (SGSN);
- The USIM in the ME.

The communication of the three entities is shown in Figure 2.2.5.2.1.



**Figure 2.2.5.2.1: Authentication and Key Agreement Protocol**

## 2.2.6 The current Status of LTE Security

LTE is still open for hackers attack. For example, DoS attacks and the length limitations of the security keys leave the system wide open for attack (Banescu, 2009). The author believes that similar solutions offered to WiMAX could work for the LTE security setup. Part of the development simulator, NetSecSim architecture is ready for rendering LTE security technology along with WiMAX.

## 2.2.7 WiMAX-LTE Comparison

WiMAX and LTE are two competing technologies. There are similarities and minor differences as shown in Table 2.2.7.1.

**Table 2.2.7.1:** Technical Differences between WiMAX (IEEE 802.16m) and LTE-Advanced to Deliver 4G

	<b>WiMAX (IEEE 802.16m)</b>	<b>LTE-Advanced</b>
<b>Bandwidth</b>	Up to 40 MHz	Up to 100 MHz
<b>IP Architecture</b>	Flat-IP	Flat-IP
<b>Access Technology</b>	Downlink: SOFDMA Uplink: SOFDMA	Downlink: OFDMA Uplink: OFDMA, SC-FDMA hybrid
<b>Spectrum Options for Deployment</b>	limited	It is easy to migrate
<b>Mobility Speeds</b>	Up to 120 Km/Hour	Up To 450-500 Km/Hour
<b>2G &amp; 3G Handover</b>	Not Possible	Possible
<b>MIMO Configurations</b>	Advance	Advance
<b>Frame Duration</b>	5 ms	10 ms
<b>Cost (Chang, et al., 2010)</b>	Low	High

The cost of building WiMAX is much cheaper than building LTE-Advanced but the advantage of backward compatibility made LTE to have the major impact politically.

## 2.2.8 Merging

Since there are large number of customers to both operators, merging the two standards makes sense. A new vision to a new modem has been developed by Green Packet. Green Packet is a leading mobile wireless network provider who has used their expertise in WiMAX and LTE to merge both technologies to provide a Duo band modem solution (Green Packet, 2015). Duo provides two modes of operation, one for WiMAX and the other for LTE.

The research of this thesis has recognised the fact that both standards, WiMAX and LTE, are encountering similar problems and threats. The security vulnerabilities to WiMAX and LTE pushed researchers and industry to find similar solutions to both standards. A few techniques used by hackers that pose a constant threat to WiMAX and LTE include DoS, jamming, masquerading etc. If there is a solution to one protocol, it may be a solution for the other standard.

## 2.3 Summary

This Chapter listed all available wireless standards since 1G. A chart of the protocols of wireless classification has been exemplified in this Chapter. In addition, the chapter concluded that features of the security setup for these protocols do have common ground and share the same problems faced by users with hackers.

Moreover, this chapter looked at the specification of the IEEE 802.16e standard and its revision in the MAC security layer. The next chapter presents and suggests a universal solution to the problem revealed and defined in Chapter 1. The proposed solution is called Zaabi Algorithm. It is aimed at tackling all types of known attacks on WiMAX networks at the Transport and Application layers.

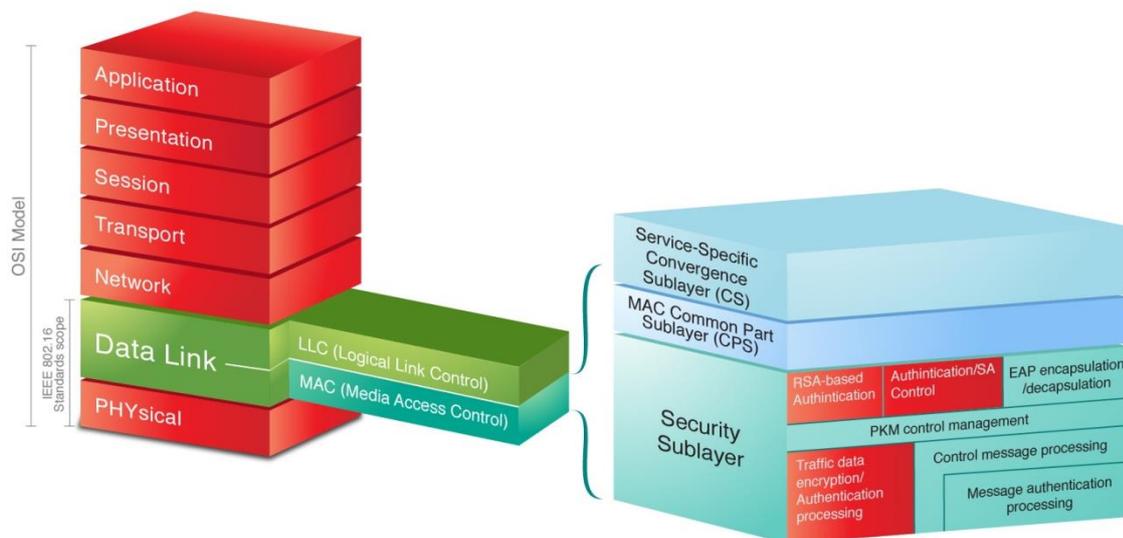
## Chapter 3: The Proposed Algorithm: Zaabi Security Algorithm (Z Algorithm or ZA)

### 3.1 Overview

WiMAX security is essential in the MAC (Media Access Control) layer although there are security protocols in the upper layers and in the physical layer which further enhances the security robustness of WiMAX. Despite the efforts and implementations of security features, weaknesses still exist in the overall architecture. For example, at the initial entry into the network, the management messages are sent without any encryption and hence make the network susceptible to DoS attacks.

Novel ideas on security issues have arisen via active research to enhance the WiMAX security protocol, but have still fallen short of completely securing the initial entry into the network.

A new proposed algorithm, ZA, is designed to provide universal solutions to WiMAX security threats such as DoS, securing of management messages and masquerading as well as standard security requirements as shown in Figure 3.1.1. This is covered in section 3.2.

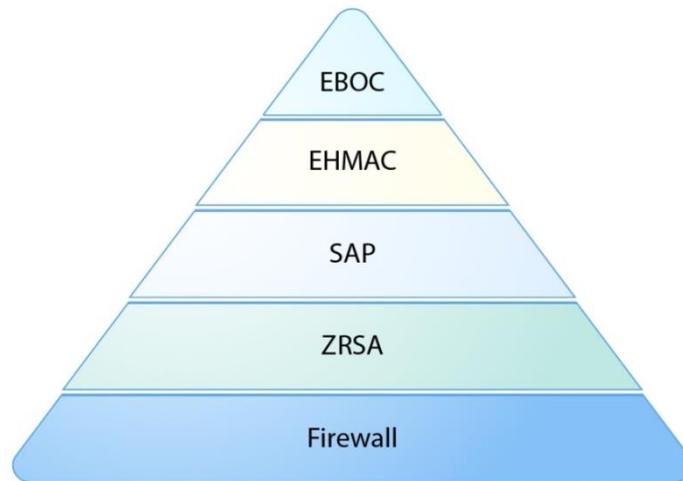


**Figure 3.1.1:** The security threats on IEEE 802.16 – 2004 Protocol Layers in OSI Model

This chapter provides a summary of the research carried out on WiMAX security and the new proposed algorithm.

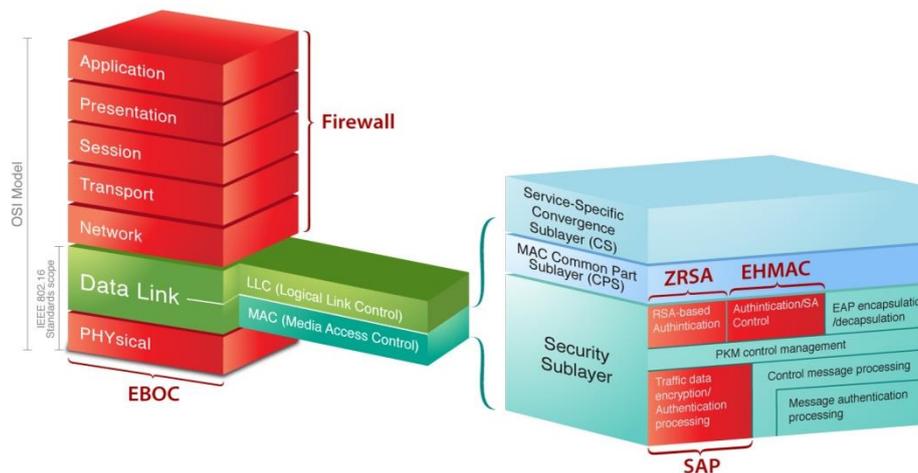
### 3.2 Overall ZA

The overall Z Algorithm is a suite of solutions that consists of Exponential Back Off Counter (EBOC), Enhanced HMAC algorithm (EHMAC), Simple Authentication Protocol, ZRSA algorithm and Firewall at Client's Premises for WiMAX is shown in Figure 3.2.1.



**Figure 3.2.1: Z Algorithm**

The components of the Z Algorithm over WiMAX are shown in Figure 3.2.2.



**Figure 3.2.2: Overall Z Algorithm Over WiMAX Architecture**

The following subsections describe each of these components in details:

- EBOC
- Enhanced HMAC algorithm,
- Simple Authentication Protocol,
- ZRSA algorithm, and
- Firewall at Client's Premises

### 3.2.1 Exponential Back Off Counter (EBOC)

The initial signalling process in WiMAX is carried out before sending the data packet Request to Send / Clear to Send / Acknowledge (RTS, CTS, ACK). This technique introduces the hidden terminal problem that can be used by the rogue attacker to mess up the system.

802.16e is “connection oriented technology” and it uses Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA) protocol to avoid collisions and sense the medium. In this system, the rogue attacker can easily bring down the system by continuously transmitting some useless data packets to a server or a host.

Once the rogue attacker gets the chance to send a packet he can use the medium till he decides to stop. That means nobody can stop him due to the unfairness back off counter system related to CSMA/CA. This is the case because CSMA/CA uses a binary back off algorithm and if any sender wins the competition to send the data, they can use the medium till they want to stop sending data. Hence, all other users would have to wait till the current sender stops its transmission due to the low back off counter value of the current sender. So if the attacker keeps sending messages to the server, it means other users detect that the server is busy or in fact the medium is busy.

To overcome this problem, Exponential Back Off Counter (EBOC) is introduced here. This technique is used in Multiple Access with Collision Avoidance (MACA) and Multiple Access with Collision Avoidance for Wireless (MACAW) protocols. With this implementation all users get a chance to send packets even if they fail to win the contest at the beginning but this adds more overhead to the network.

### 3.2.2 Enhanced Hash-Based Method Authentication Code: (EHMAC)

This is an algorithm that provides integrity to the message being sent. The message is constructed and passed through a MAC algorithm and the output is concatenated with the constructed message and sent to the receiver who then performs the same procedure with the message and compares it to the received MAC output, if they are the same, the integrity of the message is verified else the receiver discards the message. It is normally used in conjunction with other algorithms and protocols that provide confidentiality, authentication, non-repudiation and identity.

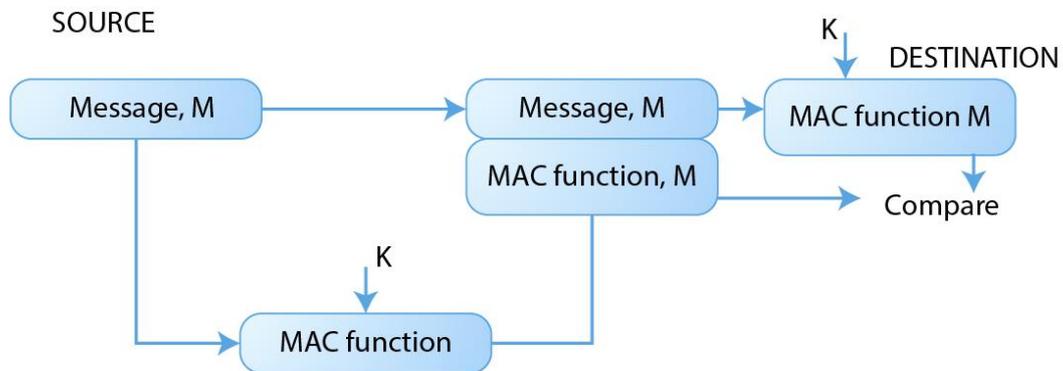
The requirement for the algorithm would be to share a secret key between two parties P and Q intending to exchange messages. The exchange of message involves the passing of the message through a MAC algorithm which uses the shared secret key to execute the MAC function. For message M, key K and the MAC function MAC, the function is calculated as follows:

$$MAC(M) = MAC(K,M) \text{-----}(3.2.2.1)$$

The message  $M$  and  $MAC(M)$  calculated by  $P$  is sent to  $Q$  which then performs the same function on the received message and gets  $MAC(M)'$  and compares it to the received  $MAC(M)$ , if they are the same then the receiver  $D$  can be assured that the message is genuine as sent by  $C$ . If the message is altered before getting to  $Q$ , the output of the MAC function at  $Q$  would be different compared the one sent by  $P$ . The requirement for the MAC function is that it is irreversible and any unauthorized party  $X$  cannot deduce  $M$  from  $MAC(M)$ . The MAC function calculated by  $Q$  is as follows:

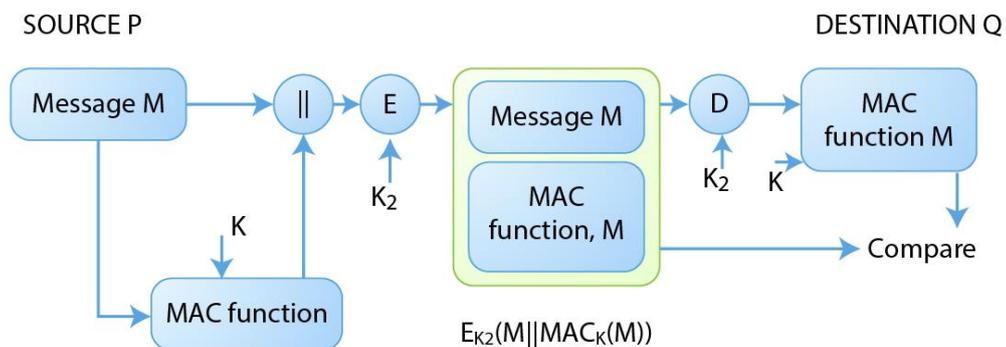
$$MAC(M)' = MAC(K, M) \text{ -----(3.2.2.2)}$$

The protocol for MAC authentication is shown in Figure 3.2.2.1.



**Figure 3.2.2.1: Message Authentication with MAC**

The MAC function can be used with other protocols and algorithms to incorporate all the required security services. The process of figure 3.2.2.1 provides message authentication only but no confidentiality, because the message is not encrypted. Figure 3.2.2.2 shows encryption of the message along with its MAC functions for authentication.



**Figure 3.2.2.2:** Message based Message Authentication and Confidentiality

As in Figure 3.2.2.2 the message  $M$  is passed through a MAC function involving a key  $K$ , the output is then concatenated with the message  $M$  and this is passed through an encryption function involving  $K_2$ , the output is sent to the receiver  $Q$  which decrypts the received message using  $K_2$  and retrieves  $M$  from it and performs the MAC function involving  $K$  on the message  $M$  to get  $MAC\ M'$ .  $Q$  compares the MAC output to the received MAC output from sender  $P$  in order to authenticate the message. Since the message  $M$  is encrypted and the MAC function is performed on the message  $M$  by the sender  $P$ , message confidentiality and authentication is provided by the protocol. In this scenario the protocol works as follows:

At Source  $P$  performs

$$P-Q: E_{K_2}(M||MAC_K(M)) \text{-----} (3.2.2.3)$$

At destination  $Q$  performs

$$Q: D_{K_2}(E_{K_2}(M||MAC_K(M))) \text{-----} (3.2.2.4)$$

To get

$$Q: M||MAC_K(M) \text{-----} (3.2.2.5)$$

$Q$  performs

$$Q: MAC_K(M)' \text{-----} (3.2.2.6)$$

$Q$  compares  $MAC_K(M)'$  from equation 3.2.2.6 with received  $MAC_K(M)$  from equation 3.2.2.5 to verify the integrity of the message.

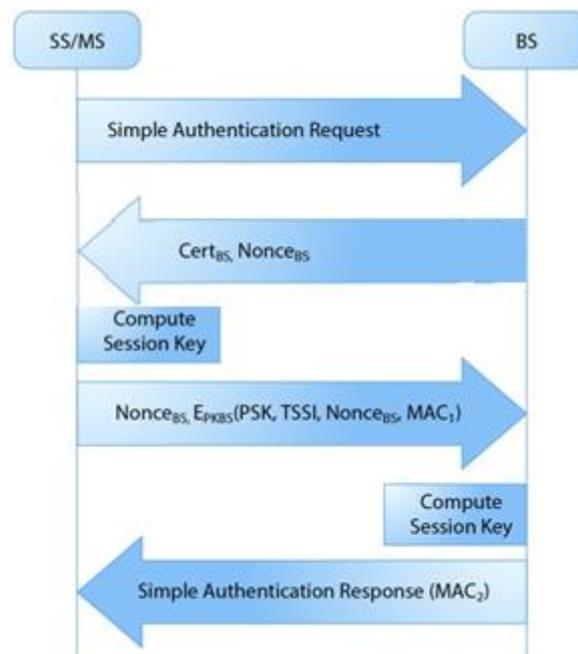
The confidentiality and message authentication is tied to the message in the protocol above, this is most preferable but there are other ways to use the MAC function to get authentication by tying the MAC function to the encrypted message.

According to (William, 2011), a MAC function is similar to encryption with one difference that a MAC function need not be reversible as it must be for decryption. If an  $n$ -bit MAC is used, then there are  $2^n$  possible MACs, whereas there are  $N$  possible messages with  $N \gg 2^n$ . Furthermore with a  $k$ -bit key, there are  $2^k$  possible keys; therefore if we have a 100-bit messages and a 10-bit MAC. Then, there are a total of  $2^{100}$  different message but only  $2^{10}$  different MACs. So on average each MAC value is generated by a total of  $2^{100}/2^{10} = 2^{90}$

different messages. Therefore due to the mathematical properties of the MAC function it is less vulnerable to being broken than encryption.

### 3.2.3 Simple Authentication Protocol

In the paper presented by (Li, et al., 2010), it was suggested to use a protocol supported by both the SS/MS and the BS for securing the management messages and eliminate DoS attacks via the management messages. Figure 3.2.3.1 below explains the “Simple Authentication Protocol”.



**Figure 3.2.3.1:** Simple Authentication Protocol

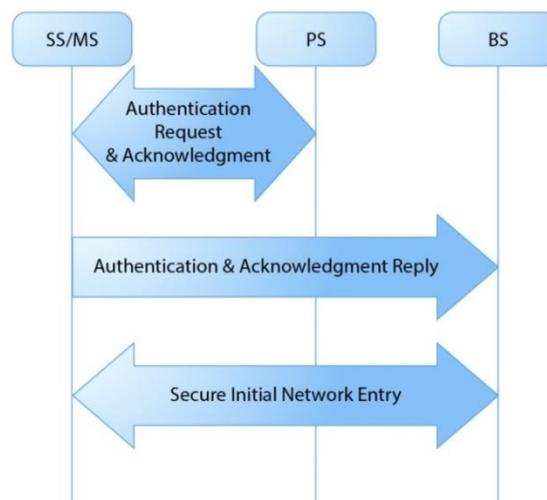
The mobile station sends a Simple Authentication Request to the BS, the BS responds by sending it an X.509 certificate and its' nonce. The MS/SS verify the authenticity of the BS from the X.509 certificate and computes the PSK (Pre Shared Key) the session key and sends, a PAK (Product Authorization Key) for authentication and PEK (Product Encryption Key) for subsequent encryption, the MS/SS computes MAC of the PAK for integrity and encrypts the PSK with TSSI (Temporal Subscriber Station Identity) and the BS's nonce with the BS's public key and sends it along with the MAC and the BS's nonce to the BS. The BS decrypts the encrypted message with its private key and checks the nonces for consistency and derives the PAK and PEK and computes the MAC' of the PAK and sends it to the MS/SS in the reply message.

### 3.2.4 Authentication using Proxy Base Station

(Bogdanoski, 2008) mentioned that the security issue in the physical layer cannot be completely eliminated and the MAC layer security can be vulnerable by mounting a DoS attack on the MS by flooding it with multiple bogus registration messages, the MS will not be able to process so many messages and this will temporarily cause Denial of Service.

(Nguyen, 2009) described in his paper the various MAC layer attacks on WiMAX such as masquerading, Eavesdropping of management messages and various types of Denial of Service attacks. These were verified by works of (Altaf, et al., 2008) and (Elleithy, et al., 2008) on the existing version of 802.16, and probably in the future the standard will mature to cover all security issues.

(Tshering, et al., 2013) have described in their paper the novel method of authentication by means of a Proxy Server (PS) which allows for MS's to authenticate to the PS before being handed over to the BS for authorization. This procedure is still vulnerable to DoS attacks between the MS and PS. Although it eliminates DoS at the BS by flooding of authentication messages, it does not eliminate it completely as the PS is still vulnerable. The procedure is shown in Figure 3.2.4.1.



**Figure 3.2.4.1:** Authentication using Proxy Base Station

The MS/SS sends its X.509 certificate to the PS for authentication and once the PS has verified the authenticity of the SS/MS it hands over the authorization process to the BS for authorization key exchange and data transfer. The management messages exchanged between the SS/MS and PS are susceptible to MITM attack and DoS attacks. Hence the problem might be eliminated at the BS but it is not completely eliminated, as the initial entry into the network via the PS is not secured.

The enhanced version of the authentication using proxy Base Station is described in section 3.3.

### 3.2.5 Zaabi RSA (ZRSA)

ZRSA is a cryptography algorithm developed with new parameters to generate secret keys that is formidable to crack within our current computing capabilities.

#### 3.2.5.1 Cryptography

Cryptography is part of the art of protecting information. The modern uses for cryptography are encrypting and decrypting emails, credit cards and other confidential data.

Cryptography is known as “symmetric key system” that generate one secret key to be shared between the sender and the receiver and “public key system” (asymmetric) that generates two keys, a public key which is known by all and a private (secret) key which is known by the receiver.

##### 3.2.5.1.1 Encryption Techniques

While doing encryption with the help of computers, there are two main approaches. These two approaches are symmetric encryption and asymmetric encryption systems.

###### 3.2.5.1.1.1 Symmetric Encryption

Symmetric encryption, also known as secret key cryptography is based on using the same key to encrypt and decrypt a message.

This technique requires great care in key distribution because the same key both encrypts and decrypts the message (Delfs, 2007). The problem is how to distribute keys and the solution varies. One can get along using a key when you physically meet or remotely using any type of media. Here, it is appropriate to select a media other than the one used for the encrypted information.

###### 3.2.5.1.1.2 Asymmetric Encryption

The form of cryptosystem that uses encryption and decryption with two separate keys is called asymmetric encryption. One of the keys is called public key and the decryption key is called private key (Ferguson & Schneier, 2003).

Asymmetric encryption is thus based on a user to use different keys to encrypt and decrypt a data set. The technique is also called Public Key Cryptography.

Each user has a key pair consisting of a public and a private key. The public key can be made available to other users via a database. The private key, as the name suggests, is available only to its owner and the user should not give it to anyone else.

Asymmetric encryption algorithms based on the use of a reverse approach are called trap door encryption (Diffie & Hellman, 1976). A trapdoor is a process that is easy to implement in one direction, but very difficult to do if one is to go back in the other direction. The asymmetric encryption algorithms that are considered safe today use all the trap doors that are taken from mathematics.

An example of an asymmetric encryption algorithm is the RSA algorithm, to be described in section 3.2.5.1.1.2.1. It is a block cipher algorithm that divides the plaintext into blocks that are encrypted separately. The size of the blocks depends on the length of the key that is used.

It is an asymmetric encryption algorithm because it uses public and private keys. Therefore, a user must first generate public and private keys to be able to make use of the RSA algorithm. The keys are calculated according to a certain pattern to be discussed in this section.

When a message is encrypted with asymmetric technology, it is done in the following way. Subscriber A wants to send an encrypted message to subscriber B, A encrypts the message with B's public key. When B receives the message, he can decrypt it with his private key.

The RSA algorithm easily multiplies two large prime numbers  $p$  and  $q$  to power  $n$  (Li, 1998) but much more difficult to factor  $n$  to  $p$  and  $q$ .

#### 3.2.5.1.1.2.1 RSA ALGORITHM

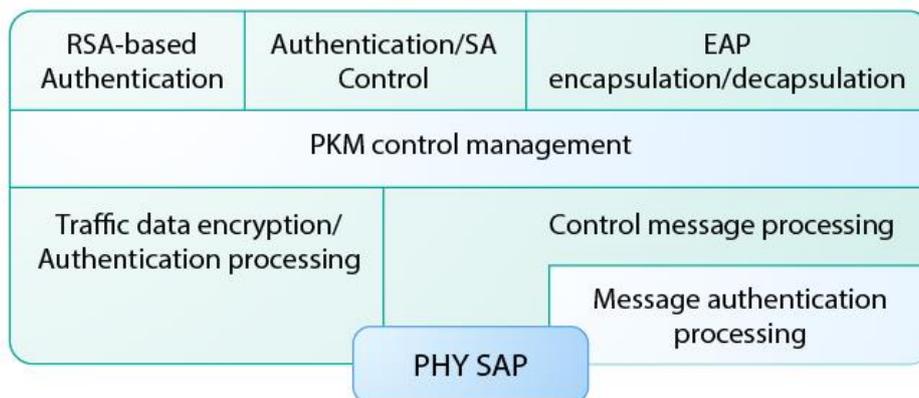
RSA algorithm is an asymmetric encryption algorithm that was developed in 1977. The algorithm gets its name from its three creators' surnames, Rivest, Shamir and Adelman. It is a block cipher algorithm, divides the plaintext into blocks that are encrypted separately. The size of the blocks depends on how big the key is used.

The RSA algorithm is an asymmetric encryption algorithm that uses public and private keys. Therefore, a user must first create a public and a private key to be able to make use of the RSA algorithm. The keys are calculated according to a certain pattern.

#### 3.2.5.1.1.2.1.1 ARCHITECTURE OF RSA BASED ON MAC LAYER

The Architecture of an RSA network is more useful in transmitting secured access of data, which can include, voice, video, and multimedia and the service quality can be unbeatable depending on its long-range transmission of information in the sequence of bits (Kapil, 2012). The MAC layer plays an important role in authentication and other encryption standards used

in the RSA algorithm, which is mainly functional to deploy the WIMAX Network IEEE 802.16 standard. The RSA standard is an Asymmetric key algorithm, which generally works on the principle of digital signature technique, and performs key exchange of valuable information between the MS (Mobile Subscriber or Subscriber) and the BS (base Station). The MAC layer is mainly useful for the service delivery of packets from the source to destination. RSA Based MAC Layer in IEEE 802.16 is shown in Figure 3.2.5.1.1.2.1.1.1 (Molisch, 2010).



**Figure 3.2.5.1.1.2.1.1.1:** RSA Based MAC Layer in IEEE 802.16

### 3.2.5.1.1.2.2 Working Principle of RSA

The working Principle and the key establishment process of RSA is described as follows. The RSA Algorithm revolves around the three basic step wise procedures and they are classified accordingly as Key Generation, Encryption and Decryption.

### 3.2.5.1.1.2.3 Key Generation

There are generally two types of keys in RSA (Ferguson & Schneier, 2003). They are classified into public and private keys. The public key can be disclosed to anyone but the private key is confidential and never disclosed as it's primarily very important in decrypting the data encoded using the RSA system. The keys are generated from the below factors.

Initially the user opts for two distinct prime numbers  $p$  and  $q$ . For security purposes, the integers,  $p$  and  $q$ , should be chosen at random, and should be the same bit-length. Prime integers can be efficiently found using a primality test.

The next stage involves in computing and finding the value of " $n$ " which is equal to " $n=p*q$ "

The third iteration process is using moduli for the asymmetric cryptographic process.

The next step is computing the values of " $\phi$ " which is  $\phi = (p-1)(q-1)$

The Fifth step involves in choosing an integer  $e$  such that  $1 < E < \phi$ ,  $(E, \phi) = 1$

We then find the value of the  $d = E^{-1} \text{ Mod } \phi(n)$ ; i.e.  $d$  is the multiplicative inverse of  $E \text{ mod } \phi$ , and  $d$  is kept as a private exponent encryption secret.

#### 3.2.5.1.1.2.4 Encryption

Assume a subscriber sender at "X", transmitting their public key  $(n,E)$  to another subscriber sender at "Y", the destination user keeps the private key as secret and does not disclose any information to the system. The text message is encrypted and sent to the destination as a cipher text to the user at the receiving end:

$$C = M^E \text{ Mod } n$$

This is shown in Figure 3.2.5.1.1.2.4.1.

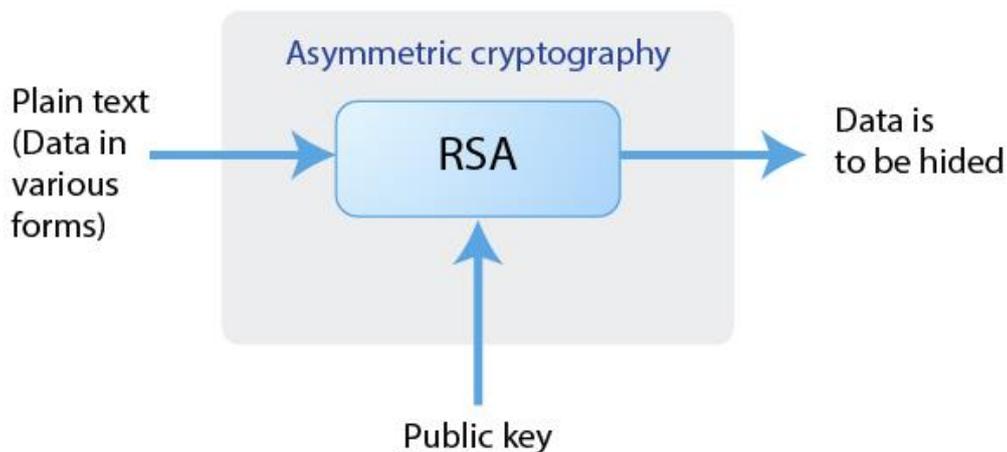


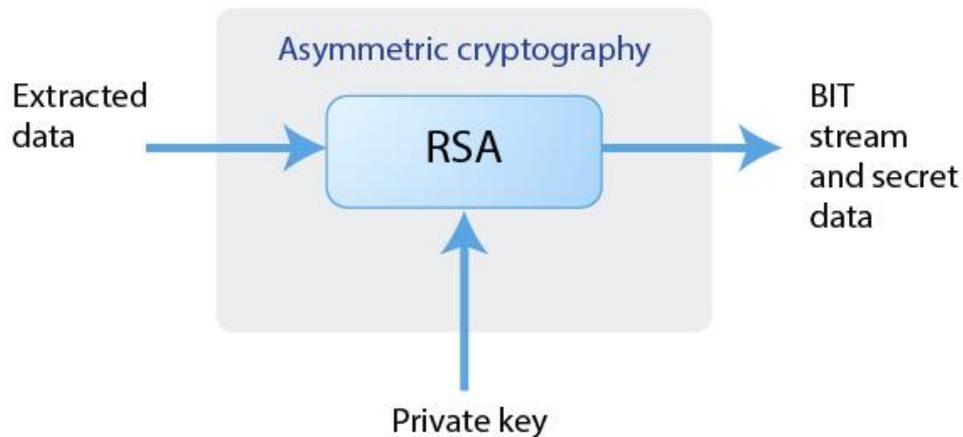
Figure 3.2.5.1.1.2.4.1: Encryption Using Public Key

#### 3.2.5.1.1.2.5 Decryption

The Cipher text can be transformed into the original text through a recovering technique using the factor  $d$  of the cryptographic component.

The Message which is obtained can be transformed into  $M = C^d \text{ Mod } n$ .

Generally, the length of the bit string of  $n$  should be 512 bits at least. The decryption using public at block level key is depicted Figure 3.2.5.1.1.2.5.1.



**Figure 3.2.5.1.1.2.5.1: Decryption Using Public Key**

RSA with its basic form has been factorised with different sizes of bit length. Hence, researchers and industry are seeking more secure algorithms.

#### [3.2.5.1.1.2.6 Modified RSA with Two Random Numbers and Three Prime Numbers](#)

The following two algorithms are aimed to help in increasing the security of the RSA algorithm.

#### [3.2.5.1.1.2.7 Modified RSA with Two Random Numbers](#)

Modified RSA with two random numbers is similar to RSA apart from using  $z_1$  and  $z_2$  to replace  $z$ .  $z_1 = x * y$  and  $z_2 = x * y * a * b$ .  $a$  and  $b$  are considered two random numbers. With this method,  $z_2$  is linearly bigger than  $z$ . Hence, the modified RSA is safer than the original RSA as the de factorization would require longer time. The paper provided an example to prove its accuracy.

#### [3.2.5.1.1.2.8 Modified RSA with Three Prime Numbers](#)

Paper published by (Patidar, 2013) listed a new RSA algorithm with three prime numbers, instead of the usual two prime numbers.

The three prime numbers are:

$p, q$  &  $r$ .

$\phi$  and  $n$  are calculated as follows:

$$n = p * q * r$$

$$\phi(n) = (p-1) * (q-1) * (r-1)$$

The two conditions  $\text{GCD}(E, \phi(n))=1$  and  $1 < E < \phi(n)$  are applied to this method, which is similar to the original RSA. The sender encrypts the message using the following standard formula:

$$C = M^E \text{ Mod } n$$

While the receiver computes the private key  $d$  using:

$$d = E^{-1} \text{ Mod } \phi(n)$$

The value of  $d$  would allow the receiver to decrypt the message using the following standard formula:

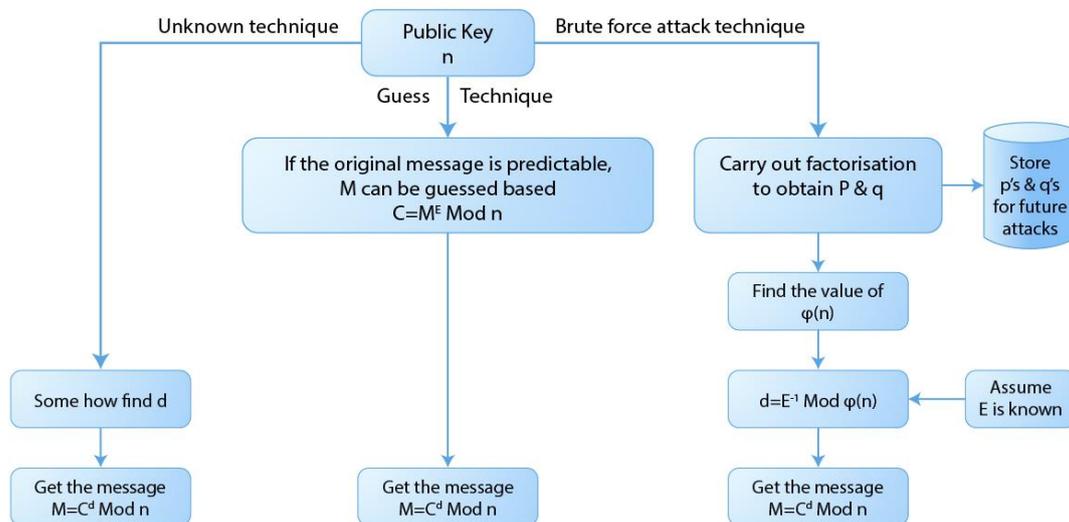
$$M = C^d \text{ Mod } n.$$

So, by inducing three prime numbers, this method has boosted the value of  $n$ . Since the number is bigger, this technique has met the same aim for the algorithm that uses two random numbers explained above.

### 3.2.5.1.1.2.2 Possible ways to attack RSA

The possible ways for attackers to obtain the private key  $d$ , is to view all parameters associated with the RSA algorithm and identify their weaknesses and strength.

The knowledge of the RSA algorithm is readily available to attackers/hackers. The parameters that are associated with the RSA algorithm are  $n$ ,  $E$ ,  $C$  and  $d$ . As disclosed above in section 3.2.5.1.1.2.1,  $n$  is public key and  $d$  is secret key. Worst case scenario, all other information of RSA is public, i.e. attackers may be able to get the values of  $n$ ,  $E$  and  $C$ . Figure 3.2.5.1.1.2.2.1 exposes RSA security issues.



**Figure 3.2.5.1.1.2.2.1: RSA security issues**

Finding  $d$  is the first technique, the Unknown Technique, which may lead to decrypting the message and publishing it in the public domain.

With the second technique, if the message is predictable, the attacker makes a guess at the original message, by giving a value to  $M$ . Therefore,  $M^E \text{ Mod } n$  can be calculated. If the result matches the encrypted version, the iterative process is to be stopped, otherwise, another

guess is given and the process starts again. This technique pushed computer manufacturers to append random padding to make the message unpredictable.

The third technique is Brute Force Attack. Here  $n$  is factorised to reach  $p$  &  $q$ . Hence,  $\phi$  can be calculated. Using the Extended Euclidian algorithm,  $d$  can be calculated and the message can be decrypted. If  $n$  is large this technique could take a long time.

Figure 3.2.5.1.1.2.2.2 defines  $n$  as small, medium and large numbers relative to the size of bits.

n is considered as:	If:
Small	$n < 2^{136}$
Medium	$2^{136} \leq n < 2^{512}$
Large	$2^{512} < n$

**Figure 3.2.5.1.1.2.2.2:**  $n$  is defined as small, medium or large depending on the size of bits

It has been reported that, if  $n$  is small or medium, RSA has been factorised (Al-Hamami, 2012). As computing power is continually increasing, it is only a matter of time for large  $n$  to be factorised. Hence, a new RSA algorithm has to be developed to make the factorisation impossible or near impossible. This is the subject of section 3.2.5.2

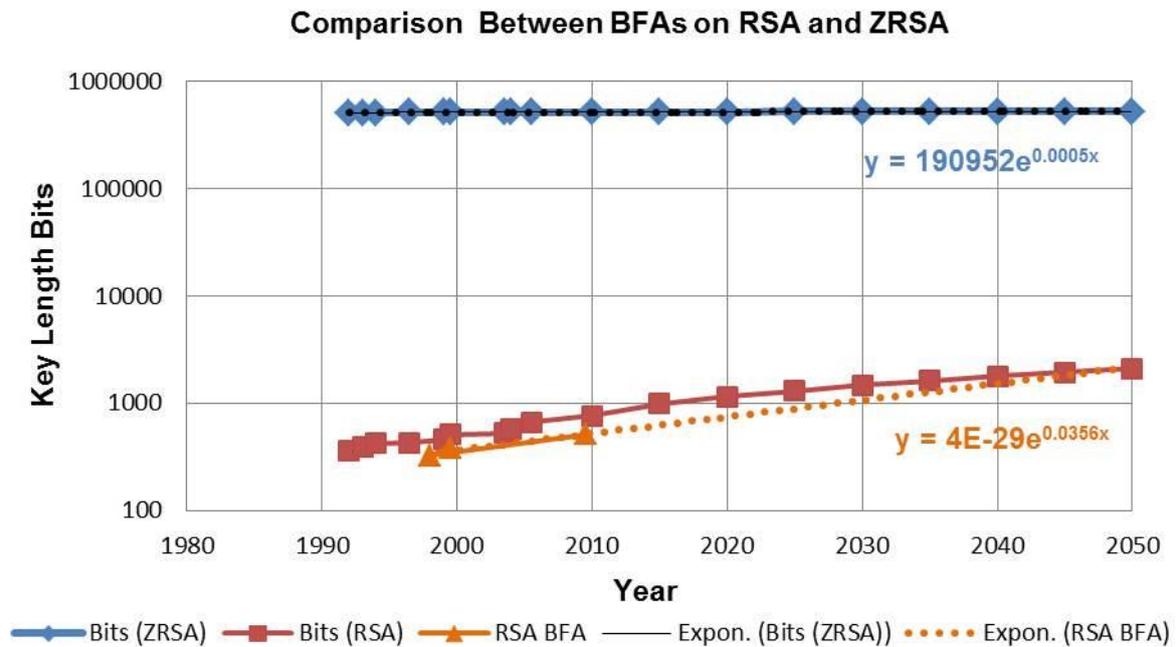
### 3.2.5.2 ZRSA

The two new RSAs listed above have provided techniques to show that the value of  $n$  is increasing linearly with the introduction of new parameters. The increase in value  $n$  may make the factorisation difficult but not different from the original RSA, at least for the time being. However, time within the last two/three decades has proven that computing power is increasing continually. So, it is a matter of short time and hackers would be able to crack RSA and the above suggested RSAs, with ease.

The suggested alternative to the original RSA algorithm in this project is referred to as ZRSA. ZRSA uses three prime numbers and three random numbers which is the main difference with the original RSA algorithm. The third random number is expressed as:

$$z = e^{1\text{st Random Number}} + e^{2\text{nd Random Number}}$$

The new parameters within ZRSA affect the three components of the generated public and private keys (E,n) & (d,n) , i.e. E, n & d. However, the values of the suggested two parameters  $e^{1st}$  and  $e^{2nd}$  are too large to handle by the available PC, z was replaced with small prime random numbers.



**Figure 3.2.5.2.1:** Comparison Between Brute Force Attacks on RSA and ZRSA

Figure 3.2.5.2.1 is explained fully later within Chapter 3.

The following section outlines the theory behind ZRSA and gives an example to verify its accuracy.

### 3.2.5.3 Working Principle of One Part of ZRSA

The working Principle and the key establishment process of ZRSA is described as follows. The ZRSA Algorithm revolves around the three basic stepwise procedures and they are classified accordingly as Key Generation, Encryption and Decryption.

#### 3.2.5.3.1 Key Generation

There are generally two types of keys in ZRSA, they are classified into public and private keys. The public key can be disclosed to anyone but the private key is confidential and never disclosed as it is primarily very important in decrypting the data encoded using the RSA system. The keys are generated from the below factors.

Initially, the user for ZRSA opts for three distinct prime numbers p, q and r. For security purposes, the integers, p, q and r, should be chosen at random, and should be of the same bit-length. Prime integers can be efficiently found out using a primality test. The user for ZRSA opts for two distinct random numbers a and b. The third random number labelled as z is expressed as:

$$z = e^{1\text{st Random Number}} + e^{2\text{nd Random Number}}$$

The next stage involves computing the value of "n" which is equal to "n=p\*q\*z"

The third iteration process is using moduli for the asymmetric cryptographic process.

The next step is computing the values of "phi" which is  $\phi(n) = (p-1)(q-1)(z-1)$

The fifth step involves choosing an integer E such that  $1 < E < \phi$ , provided  $\text{GCD}(E, \phi(n)) = 1$

The sixth step is to find the value of  $d = E^{-1} \text{ Mod } \phi(n)$ . From the formula d is the multiplicative inverse of E Mod  $\phi$ . d is kept as the private exponent encryption secret.

To find the modular inverse with respect to the ' $\phi$ ', d which is one element of the ZRSA private key, the following set of equations has to be used:

$$d = E^{-1} \text{ Mod } \phi$$

or

$$d * E = 1 \text{ Mod } \phi$$

The extended Euclidian algorithm has to be used to calculate the value of 'd', as follows:

$\text{GCD}(\phi, E) = \phi x + E y$ , where  $\text{GCD}(\phi, E) = 1$  in ZRSA, and

$y = d$  if  $d < \phi$  and  $d \neq$  negative integer.

How to find the inverse of  $\phi \text{ Mod } E$ ?

If  $\text{GCD} = x^*E + y^*\phi$ ,  $E = n$  and  $\phi = m$ ,  $\text{GCD}$  in ZRSA is = 1.

Hence:

$$1 = x^*E + y^*\phi$$

**If  $E = r_{i-1}$  and  $\phi = r_{i-2}$ , use:**

$r_{i-2} = q_i r_{i-1} + r_i$ ,  $q_i$  is quotient and  $r_i$  is the remainder ---- (1)

Equation (1) is used to generate the quotients,  $q_i$ 's.

$$d = x_{i-2} - (x_{i-1} * q_{i-2}) * \text{Mod } \phi, \text{ where } x^{-2} = 0, x^{-1} = 0, q^{-2} = 0 \ \& \ q^{-1} = 0$$

$$r_{i-2} = q_i * r_{i-1} + r_i$$

The above set of equations has been formalised in Excel file to produce  $n$ ,  $E$  and  $d$ .

Hence, the public key and the private key are:

Public Key =  $(n,E)$  and

Private Key =  $(n,d)$

### 3.2.5.3.2 Encryption

There are conditions to ZRSA, prior to transmitting and receiving, which have to be fulfilled to ensure security. These conditions are similar to the standard RSA conditions. The first condition is that a sender, subscriber (SS1), has to transmit a public key  $(n,E)$  to the receiver, the other subscriber (SS2). The second condition is that the destination user, SS2 keeps the private key unit  $d$  as secret and does not disclose it to the WiMAX system. The text message  $M$  is padded, encrypted and sent to the destination as a cipher text,  $C$  to the user at the receiving end. The encryption formula is:

$$C = M^E \text{ Mod } n$$

The conceptual idea of ZRSA encryption is shown in Figure 3.2.5.3.2.1.

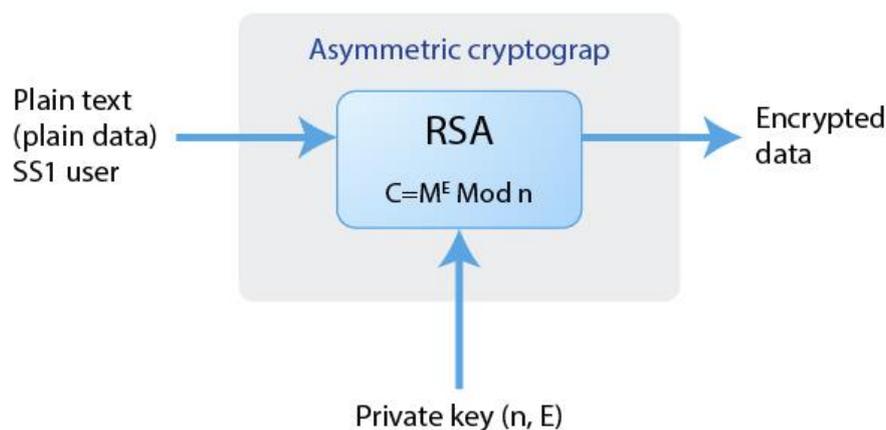


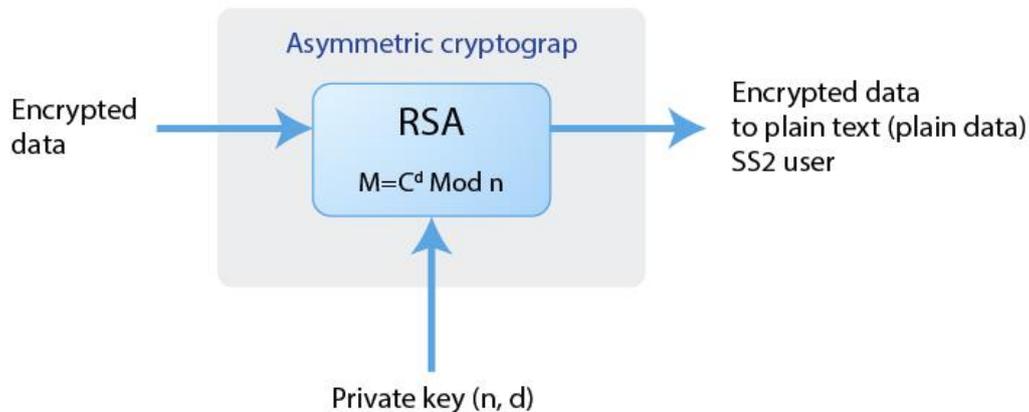
Figure 3.2.5.3.2.1: Encryption Using Public Key

### 3.2.5.3.3 Decryption

The Cipher text can be transformed into the original text using the factor  $d$  of the cryptographic component. The decryption formula is:

$$M = C^d \text{ Mod } n$$

M is transformed into plain text with a specified padding. The conceptual idea of ZRSA decryption is shown in Figure 3.2.5.3.3.1.



**Figure 3.2.5.3.3.1:** Decryption Using Private Key

#### 3.2.5.3.4 Working Example for ZRSA Algorithm

As mentioned above 'p', 'q' and 'r' are prime numbers which are supposed to be very large, but selecting two small prime numbers for 'p', 'q' and 'r', makes the calculation easier to understand.

Step 1 - select three prime numbers  $p = 7$ ,  $q = 11$  and  $r = 13$  (7, 11 and 13 are co-prime numbers).

Hence,  $\text{GCD} = 1$ .

Step 2 - calculate the ZRSA modulus:  $n = p * q * r = 7 * 11 * 13 = 1001$

Step 3 - calculate the totient:  $\phi = (p-1) * (q-1) * (r-1) = 6 * 10 * 12 = 720$

Step 4 - Select a value for 'E' ( $1 < E < 720$  ( $\phi$ ) and should be a prime and co-prime with ' $\phi$ '). Possible prime numbers above 1 are  $E = 3, 5, 7, 11, 13, 17, 19, 23, 27, 29, 31, 37, 41, 43, 53$  and 59.

3 and 5 cannot be taken as 'E' because these two numbers are not co-prime with ' $\phi$ ', any other value can be selected as 'E'. For this example,  $E = 17$  is chosen.

Step 5 - calculate the modular inverse with respect to the ' $\phi$ '

$$d * E = 1 \text{ Mod } \phi$$

Using the extended Euclidean algorithm as discussed in the previous subsection:

$GCD(\phi, E) = \phi x + Ey$ , where  $GCD(\phi, E) = 1$ ,  $\phi = 720$  and  $E = 17$

Step 1: Euclidean algorithm

$$720x + 17y = 1$$

Here, the Euclidean Algorithm is applied to 720. The listed functions were programmed within Excel. Excel has produced the value  $d = 593$ , as shown in Figure 3.2.5.3.4.1.

e		Phi		$r_{i-2} = q_i * r_{i-1} + r_i$										
Step i	a	b	$r_{i-2}$	$q_i$	$r_{i-1}$		$r_i$ (If $r_i = 0$ , stop Step $i+1$ )	$x_{i-2}$	$x_{i-1}$	$q_{i-2}$	$x_{Modi} = x_{i-2} - (x_{i-1} * q_{i-2})$	$x_i = x_{Modi} * \text{mod } b$		
0	17	720	720	42	*	17	+	6	0	0	0	0	0	0
1		720	17	2	*	6	+	5	0	0	0	1	1	1
2		720	6	1	*	5	+	1	0	1	42	-42	678	678
3		720	5	5	*	1	+	0	1	678	2	-1355	85	85
4		720							678	85	1	593	593	593

**Figure 3.2.5.3.4.1:** Euclidean Algorithm is applied to 720

Therefore, the encryption key is [1001,17] and the decryption key is [1001,593].

Encryption and decryption procedure:

For example, assume that the base station needs to transmit the message "HELLO". First, the message needs to be converted to a numeric value. The character set can be padded as:

2	3	4	6	7	8	9	12	13	14	16	17	18
A	B	C	D	E	F	G	H	I	J	K	L	M
19	21	23	24	26	27	28	29	31	32	34	36	37
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
38	39	41	42	43	46	47	48	49	51	52	53	
sp	0	1	2	3	4	5	6	7	8	9	*	
336	584	712										

Ext1= £ Ext2= ¥ Ext3= α

Hence, according to the above character set, H = 12, E = 7, L = 17 and O = 21 (HELLO).

Encryption for H →  $C = (12)^{17} \pmod{1001}$

$C = 584$  (Remainder)

Encryption for E →  $C = (7)^{17} \pmod{1001}$

$$C = 336 \text{ (Remainder)}$$

$$\text{Encryption for L} \rightarrow C = (17)^{17} \pmod{1001}$$

$$C = 712 \text{ (Remainder)}$$

$$\text{Encryption for L} \rightarrow C = (17)^{17} \pmod{1001}$$

$$C = 712 \text{ (Remainder)}$$

$$\text{Encryption for O} \rightarrow C = (21)^{17} \pmod{1001}$$

$$C = 21 \text{ (Remainder)}$$

The encrypted message  $\text{¥} = 584$ ,  $\text{£} = 336$ ,  $\alpha = 712$ ,  $\alpha = 712$  &  $\text{O} = 21$  ( $\text{¥£}\alpha\alpha\text{O}$ )

$$\text{Decryption for ¥} \rightarrow M = (584)^{593} \pmod{1001}$$

$$M = 12 \text{ (Remainder)}$$

$$\text{Decryption for £} \rightarrow M = (336)^{593} \pmod{1001}$$

$$M = 7 \text{ (Remainder)}$$

$$\text{Decryption for } \alpha \rightarrow M = (712)^{593} \pmod{1001}$$

$$M = 17 \text{ (Remainder)}$$

$$\text{Decryption for } \alpha \rightarrow M = (712)^{593} \pmod{1001}$$

$$M = 17 \text{ (Remainder)}$$

$$\text{Decryption for O} \rightarrow M = (21)^{593} \pmod{1001}$$

$$M = 21 \text{ (Remainder)}$$

Hence, the decrypted message is HELLO.

#### 3.2.5.4 Brute Force Attack on ZRSA

Brute force attack (BFA) is a trial and error technique via exhaustive effort to decrypt the private key. BFA has been applied on RSA and Data Encryption Standard (DES) keys with success. BFA is successful because of the structural weaknesses of RSA and DES algorithms.

Here is a description of BFA applied to ZRSA to find the value of  $d$ .

From the public key, the values of  $n$  and  $E$  are known to the hacker. This is possible by looking up the message's destination in the public-key directory.

Another known fact is:

$$d * E = 1 \text{ Mod } \varphi(n)$$

or

$$d * E = K * \varphi(n) + 1$$

then

$$d = (K * \varphi(n) + 1) / E$$

Now  $d$ ,  $E$  as well as  $K$  are three unknowns. By trial and error,  $K$  can be worked out. The following example illustrates this technique.

From the above example:

$$E = 17 \text{ and}$$

$$\varphi(n) = 720$$

Hence:

$$(1 * 720 + 1) / 17 = 721 / 17 \quad (\text{doesn't divide evenly})$$

$$(2 * 720 + 1) / 17 = 1441 / 17 \quad (\text{doesn't divide evenly})$$

$$(3 * 720 + 1) / 17 = 2,161 / 17 \quad (\text{doesn't divide evenly})$$

$$(4 * 720 + 1) / 17 = 2,881 / 17 \quad (\text{doesn't divide evenly})$$

$$(5 * 720 + 1) / 17 = 3,601 / 17 \quad (\text{doesn't divide evenly})$$

...

$$(14 * 720 + 1) / 17 = 10,081 / 17 = 593 \quad (\text{This is } d!)$$

From the formula  $d = (K * \varphi(n) + 1) / E$ , each search step with BFA would require 1 multiplication, one addition and one division. Assume, in an ideal world, a processor with the best speed would require:

<u>Operation</u>	<u>Cycle</u>	<u>Time per cycle in ns</u>
Addition	1	1
Multiplication	1	1
Division	16	16
Total		18

Overhead processor operation that covers reading and writing to memory and other requirements is assumed to be 100ns. Hence, the assumed total required time is 118ns.

So, the total length of time to BFA ZRSA on the example above takes:

$$118\text{ns} \times 15 \text{ (steps)} = 1,770\text{ns} = 0.00177\text{ms} = 0.00000000177\text{sec}$$

The bit length for the above example is 10 bits and it takes approximately 0.00000000177sec.

It has been indicated (Exchange, 2012) that for every extra 10 bits, the duration of a BFA crack would double. If  $T_{\text{Length of Bits}}$  is the time BFA takes to find the value of  $d$  and  $T_{10} = 0.00000000177\text{sec}$ , then:

$$T_{\text{Length of Bits}} = T_{10} + T_{10+10} + T_{10+10+10+10} + \dots$$

$$T_{\text{Length of Bits}} = T_{10} + 2T_{10} + 4T_{10} + 8T_{10} + \dots$$

$$T_{\text{Length of Bits}} = T_{10} (1 + 2 + 4 + 8 + \dots)$$

$$T_{\text{Length of Bits}=10x} = T_{10} (1 + 2 + 4 + 8 + \dots)$$

The series  $1 + 2 + 4 + 8 + \dots$  is the Geometric Sequence (GS) and if  $n = 2$ , the sum of the sequence is calculated using:

$$Sum_{GS} = \frac{1 - r^{n+1}}{1 - r}$$

$$Sum_{GS} = -(1 - 2^{n+1})$$

$$T_{\text{Length of Bits}=10*x} = T_{10} * (2^{n+1} - 1)$$

So, if the length of bits is 100, Length of Bits =  $10*10$ , and

$$T_{\text{Length of Bits}=10*10} = T_{10} (1 + 2 + 4 + 8 + 16 + 32 + 64 + 128 + 256 + 512 + 1024)$$

$$= T_{10} * 2047$$

$$= 0.00000000177 * 2047 = 0.00000362319\text{sec}$$

If the length of the bits is 512:

$$= T_{10} * 9,007,199,254,740,991 = 15,942,742.68089155407\text{sec} = 6.5 \text{ months}$$

Lately, (Skrevet, 2013) is suggesting to use minimal 1024 bits but Windows 7 and 8 are using 512 bits. ZRSA with 3 prime numbers and 3 random numbers is amplifying  $n$  and subsequently  $\phi$ . In other words, it is reaching the bigger numbers and, in some cases, reaching the maximum value of  $n$  with ease. Moreover, as it was discussed earlier, the computing performance is continuing to increase and will be so for the next decades, and with the inherited feature of ZRSA, it would be still easy to use by then.

However, the following discussion shows that the three prime numbers that are required for the ZRSA algorithm could improve the aim of the security algorithm by generating three different d's. Since, ZRSA uses three prime numbers, generating three d's is possible if prime p generates  $d_p$ , prime q generates  $d_q$  and prime r generates  $d_r$ . With three d's, ZRSA is called Modified ZRSA. The full details of the modified ZRSA algorithm are presented in section 3.2.5.5.

### 3.2.5.5 Working Principle of Modified ZRSA

Again the modified ZRSA is based on three basic steps: Key Generation, Encryption and Decryption. All of the three steps are described below.

#### 3.2.5.5.1 Key Generation

The values of n's and  $\phi$ 's are worked out as follows:

Set of n's equations:

$$n_p = q * z$$

$$n_q = p * z$$

$$n_z = p * q$$

Note that  $n_p$  is related to the combination of prime's q & z,  $n_q$  is related to the combination of prime's p & r and  $n_r$  is related to the combination of prime's p & q. other combinations, such as p & q, q & z and z & p or even p & q & z, will lead to similar results.

Set of  $\phi$ 's equations:

$$\phi_p = (q-1)(z-1)$$

$$\phi_q = (p-1)(z-1)$$

$$\phi_r = (p-1)(q-1)$$

ZRSA generates two types of keys. They are classified into public and private keys. The public key can be disclosed to anyone but the private key is confidential and never disclosed as it is primarily very important in decrypting the data encoded using the RSA system. The keys are generated from the below factors.

Initially, the user for ZRSA opts for three distinct prime numbers p, q and z. For higher security purposes, the integers, p, q and z, should be chosen at random and the same bit-length. Prime integers can be efficiently found out using a primality test. The user for ZRSA opts for two

distinct random numbers, namely a & b. The third random number labelled as z expressed as the multiplication of the first two random numbers, a & b. As the random numbers will increase the value of n, within this section, they have been eliminated from the following example. However, they have been implemented within the NetSimSec simulator.

The next stage involves computing the value of "n<sub>p</sub> , n<sub>q</sub> & n<sub>z</sub>".

The third iteration process is using moduli for the asymmetric cryptographic process.

The next step is computing the values of "phi's" which are "φ<sub>p</sub> , φ<sub>q</sub> & φ<sub>z</sub>".

The fifth step involves choosing an integer E such that 1 < E < φ, GCD(E, φ(n)) = 1

The sixth step is to find the value of d = E<sup>-1</sup> Mod φ(n). From the formula, d is the multiplicative inverse of E Mod φ. d is kept as the private exponent encryption secret.

To find the modular inverse with respect to 'φ', d, the following set of equations has to be used:

$$d_p * E_p = 1 \text{ Mod } \phi_p$$

$$d_q * E_q = 1 \text{ Mod } \phi_q$$

$$d_z * E_z = 1 \text{ Mod } \phi_z$$

The extended Euclidian algorithm has to be used to calculate the value of 'd', as follows:

$$\text{GCD}(\phi, E) = \phi x + E y, \text{ where } \text{GCD}(\phi, E) = 1 \text{ in ZRSA, and}$$

y = d if d < φ and d ≠ negative integer.

How to find the inverse of φ mod E for p, q & z primes?

If GCD = E\*x + φ\*y, E = n and φ = m, GCD is equal 1.

Hence:

$$1 = E*x + \phi*y$$

If E = r<sub>i-1</sub> and φ = r<sub>i-2</sub>, use the following iterative equations:

$$r_{i-2} = q_i r_{i-1} + r_i \quad q_i \text{ is quotient and } r_i \text{ is the remainder}$$

$$d = x_{i-2} - (x_{i-1} * q_{i-2}) * \text{Mod } \phi, \text{ and}$$

$$r_{i-2} = q_i r_{i-1} + r_i$$

The above set of equations has produced n, E and d. Hence, the public key and the private keys for primes p, q and z are:

p

Public Key =  $(n_p, E_p)$  and

Private Key =  $(n_p, d_p)$

q

Public Key =  $(n_q, E_q)$  and

Private Key =  $(n_q, d_q)$

r

Public Key =  $(n_z, E_z)$  and

Private Key =  $(n_z, d_z)$

ZRSA public and private keys can be summarised as two set of keys:

**Public key**  $(E_p, E_q, E_z, n_p, n_q, n_z)$

**Private key**  $(d_p, d_q, d_z, n_p, n_q, n_z)$

p, q & z keys are used for successive letters repeatedly.

### 3.2.5.5.2 Encryption and Decryption

The encryption and decryption keys for:

p

To be applied to the first letter of the encrypted/decrypted message:

Encryption  $C = M^{E_p} \pmod{n_p}$

Decryption  $M = C^{d_p} \pmod{n_p}$

q

To be applied to the second letter of the encrypted/decrypted message:

Encryption  $C = M^{E_q} \pmod{n_q}$

Decryption  $M = C^{d_q} \pmod{n_q}$

z

To be applied to the third letter of the encrypted/decrypted message:

Encryption  $C = M^{Ez} \pmod{n_z}$

Decryption  $M = C^{dz} \pmod{n_z}$

### 3.2.5.5.3 Example

The following example has been applied to the “HELLO” message implementing the modified ZRSA process:

#### 3.2.5.5.3.1 Prime p Keys

First, prime p keys can be calculated as follows:

**p = 11, q = 23, z = 53**

$n_p = q * z = 23 * 53 = 1219$

$\phi(n_p) = (q-1) * (z-1) = (23-1) * (53-1) = 1144$

$E_p = 157$

Using the Extended Euclidian algorithm to calculate  $d_p$ :

The GCD formula is

**$ax + by = \text{GCD}(a, b)$**

If applied on prime p, the coefficients a & b should be replaced with  $\phi_p$  &  $E_p$  consecutively and y with d as follows:

**$\phi_p x + E_p d = \text{GCD}(\phi_p, E_p)$**

or

**$\phi_p x + E_p d = 1$**

The Euclidean Algorithm is applied to prime p to generate the public and private keys. The listed functions were programmed within Excel using  $E = 157$  &  $\phi_p = 1144$ . As a result, Excel has produced the value  $d = 1093$ , as shown in Figure 3.2.5.5.3.1.

	e	Phi	$r_{i-2} = q_i * r_{i-1} + r_i$									
Step i	a	b	$r_{i-2}$	$q_i$	$r_{i-1}$	$r_i$ (If $r_i = 0$ , stop Step i+1)	$x_{i-2}$	$x_{i-1}$	$q_{i-2}$	$x_{\text{Modi}} = x_{i-2} - (x_{i-1} * q_{i-2})$	$x_i = x_{\text{Modi}} * \text{mod } b$	
0	157	1144	1144	7	157	45	0	0	0	0	0	
1		1144	157	3	45	22	0	0	0	1	1	
2		1144	45	2	22	1	0	1	7	-7	1137	
3		1144	22	22	1	0	1	1137	3	-3410	22	
4		1144					1137	22	2	1093	1093	

**Figure 3.2.5.5.3.1:** Euclidean Algorithm is Applied to Calculate Prime p Keys

Therefore, the encryption key is [1144, 157] and the decryption key is [1144, 1093].

### 3.2.5.5.3.2 Prime q Keys

Next, prime q keys can be calculated as follows:

$$p = 11, q = 23, z = 53$$

$$n_q = p * z = 11 * 53 = 583$$

$$\phi(n_q) = (p-1) * (z-1) = (11-1) * (53-1) = 520$$

$$E_q = 23$$

The Excel Extended Euclidian algorithm calculated d. It is:

$$d_q = 407$$

Therefore, the encryption key is [583, 23] and the decryption key is [583, 407].

### 3.2.5.5.3.3 Prime z Keys

Finally, prime z keys can be calculated as follows:

$$p = 11, q = 23, z = 53$$

$$n_z = p * q = 11 * 23 = 253$$

$$\phi(n_z) = (p-1) * (q-1) = (11-1) * (23-1) = 220$$

$$E_z = 139$$

The Excel Extended Euclidian algorithm calculated d. It is:

$$d_z = 209$$

Therefore, the encryption key is [220, 139] and the decryption key is [220, 209].

### Another Example with Prime and random numbers

Prime 1 = 8719; prime 2 = 239; prime 3 = 503;

Random 1 (A) = 5214; Random 2 (B) = 46000; Random 3 (W) = 1534;

E = 13;

### 3.2.5.5.3.4 Encrypting The Message

The following steps show an example of encrypting and decrypting the message "HELLO" message using Modified RSA algorithm using the example in section 3.2.5.5.3

#### Encrypting the first character

##### Letter H

$$C = M^{EP} \text{ Mod } n_p$$

$$C = 8^{157} \text{ Mod } 1219 = 167$$

#### **Decrypting the first character**

$$M = C^{dp} \text{ Mod } n_p$$

$$M = 167^{1093} \text{ Mod } 1219 = 8 = H$$

#### **Encrypting the second character**

##### **Letter E**

$$C = M^{Eq} \text{ Mod } n_q$$

$$C = 5^{23} \text{ Mod } 520 = 125$$

#### **Decrypting the second character**

$$M = C^{dq} \text{ Mod } n_q$$

$$M = 125^{407} \text{ Mod } 520 = 5 = E$$

#### **Encrypting the third character**

##### **Letter L**

$$C = M^{Er} \text{ Mod } n_z$$

$$C = 12^{139} \text{ Mod } 242 = 78$$

#### **Decrypting the third character**

$$M = C^{dz} \text{ Mod } n_z$$

$$M = 78^{195} \text{ Mod } 242 = 12 = L$$

#### **Encrypting the fourth character**

##### **Letter L**

$$C = M^{Ep} \text{ Mod } n_p$$

$$C = 12^{157} \text{ Mod } 1219 = 118$$

#### **Decrypting the fourth character**

$$M = C^{dp} \text{ Mod } n_p$$

$$M = 118^{1093} \text{ Mod } 1219 = 12 = L$$

#### **Encrypting the fifth character**

##### **Letter O**

$$C = M^{Eq} \text{ Mod } n_q$$

$$C = 15^{23} \text{ Mod } 520 = 215$$

#### **Decrypting the fifth character**

$$M = C^{dq} \text{ Mod } n_q$$

$$M = 215^{407} \text{ Mod } 520 = 15 = O$$

### **3.2.5.6 Comparison Between Brute Force Attacks on RSA and Modified ZRSA**

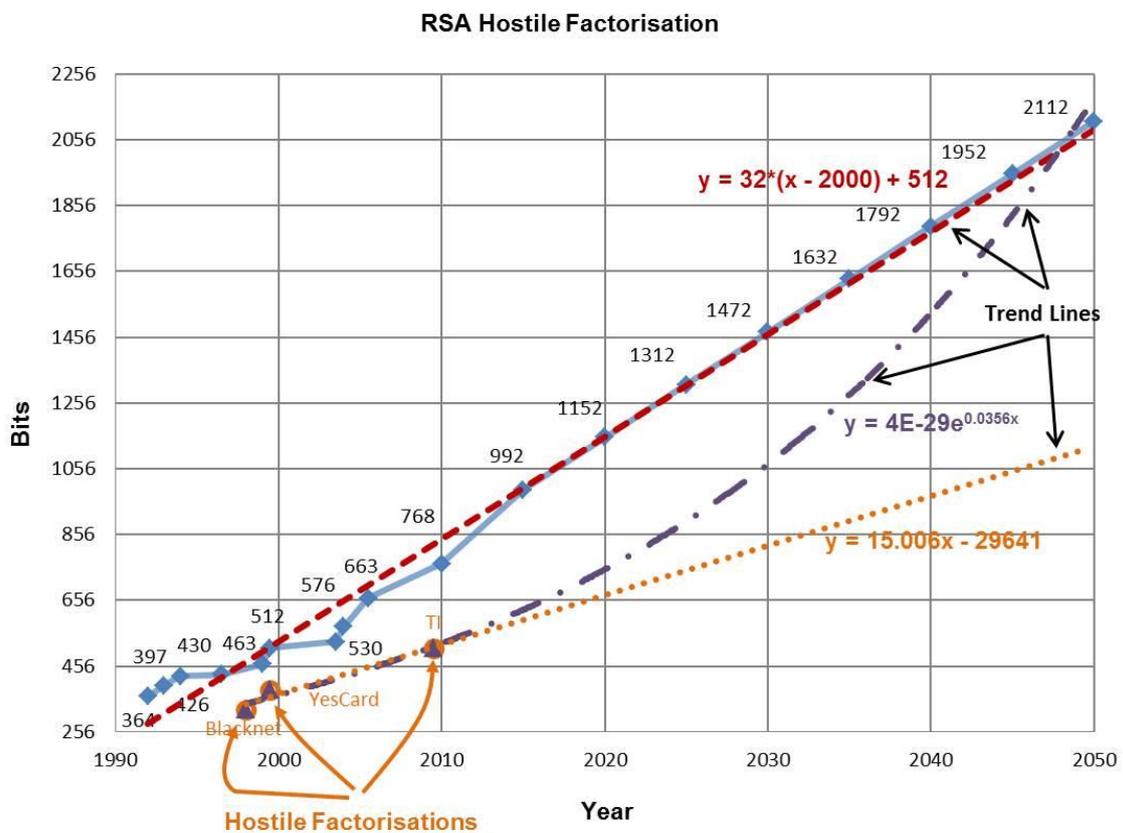
Brute Force attacks (BFAs) or hostile factorisations have been tried on RSA with success.

Three key attacks are recorded below. They represented milestones of factorising RSA since its introduction.

The process of BFA on RSA uses the main factorisation algorithm and set of efficient techniques that remove redundant steps which would speed up the factorisation rate in an interesting way. The reported BFA attacks on RSA have taken between 6 months and 2.5 year for a group of scientist mathematicians.

Due to the required length of time and resources needed to apply a successful BFA on ZRSA, a linear formula has been derived from the RSA BFA and applied on ZRSA.

Figure 3.2.5.6.1 shows RSA key length (in bits) and hostile factorisations with trendlines. The sample data of figure 3.2.5.6.1 are listed in Appendix A.



**Figure 3.2.5.6.1: RSA Key Length, Brute Force Factorisation and Trendlines**

The samples of the key length that are shown in the blue curve in figure 3.2.5.6.1 have been recommended by NSA (Giry, 2015) and (Arjen K. Lenstra, 2001) from 1990 to date. The red dashed straight trend line extends the prediction to 2050.

There are three listed RSA hostile attacks that took place between 1995 and 2009. In figure 23.2.5.6.1, they are labelled as Blacknet (1995), Yescard (1998) and TI (2009):

1995: With two Sun machines with a peak computing power of 1300 MIPS, it took three months to crack RSA with a 384 bits bit key length (Leyland, 1995). This is called Blacknet PGP Key.

1998: Serge Humpich was able to factor the 321 bit key for the French YesCard (Pelé, 1999). The hostile factorization called YesCard Key.

2009: Benjamin Moody listed the factorization of the 512-bit RSA modulus (Michael, 2009). There are no details published in open literature about Moody's brute force method or computing power needed.

These three hostile attacks have been shown in figure 3.2.5.6.1 with an orange dotted linear line represented in Eq 3.2.5.6.1:

$$y = 15.006x - 29641 \text{-----} \text{ (Eq 3.2.5.6.1)}$$

Eq3.2.5.6.1 is the trendline for BFA RSA, assuming that the trend of the advances in technology would still be the same as today with binary logic. So, for example, in 2050, the factorised-bits of RSA would be 1980.94.

However, new promising technologies are emerging that can deliver speed and performance to computing which is unparalleled to the established binary silicon technology. Quantum Computing (QC) and Quantum Biology (QB) are rapidly maturing to deliver real time performance to biophysical and metabolic processes within living systems, as well as factorising secure encryption algorithms in the cryptanalysis field.

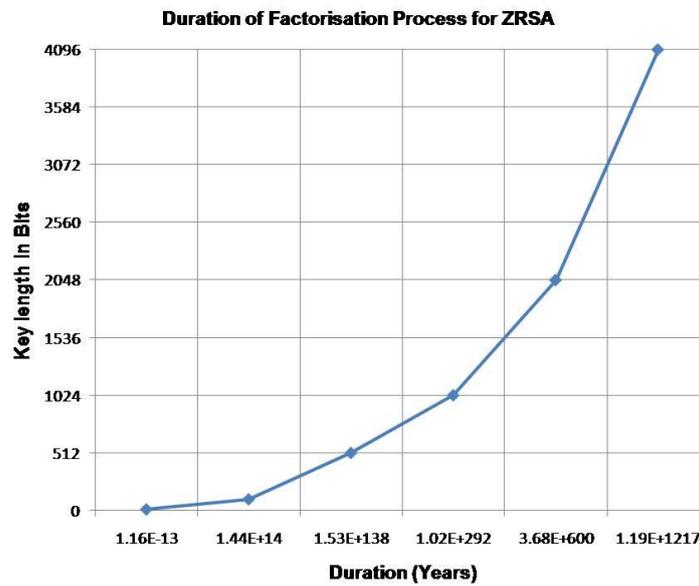
Instead of binary, they use Qubits (Quantum Bits) logic (Dudziak, 2014) to achieve nearer-than-expected results. Though, QC and QB are not yet implemented, full scale investment in research has been utilised for this purpose and within a few years they will be commercialised.

However, to take into account the expected advances in delivering speed and performance in computing, the trendline linear formula shown in Eq 3.2.5.6.1 has to be changed. The changes in the formula have to be in line with exponential growth. Hence, if the formula is changed to predict the advances of the technology with an exponential growth, as shown in figure 3.2.5.6.1, the new formula will be:

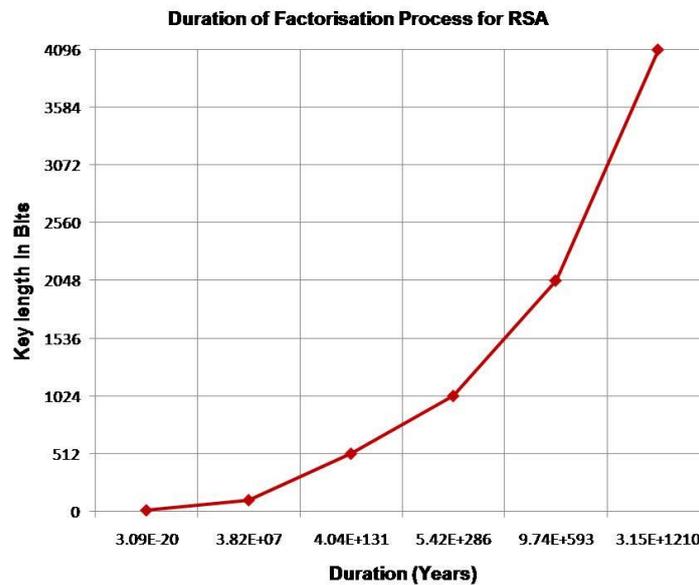
$$y = 8E-234x^{71.397} \text{-----} \text{ (Eq 3.2.5.6.2)}$$

Eq 3.2.5.6.2 is the purple dashed-dotted trend curve to BF RSA, assuming that the trend of the advances of the technology would be based on Qubit logic. So, for example, in 2050, the factorised-bits of RSA would be 2251.03.

To compare the factorisation process between RSA and ZRSA, figure 3.2.5.6.2 (a) and (b) have been drawn.



(a)

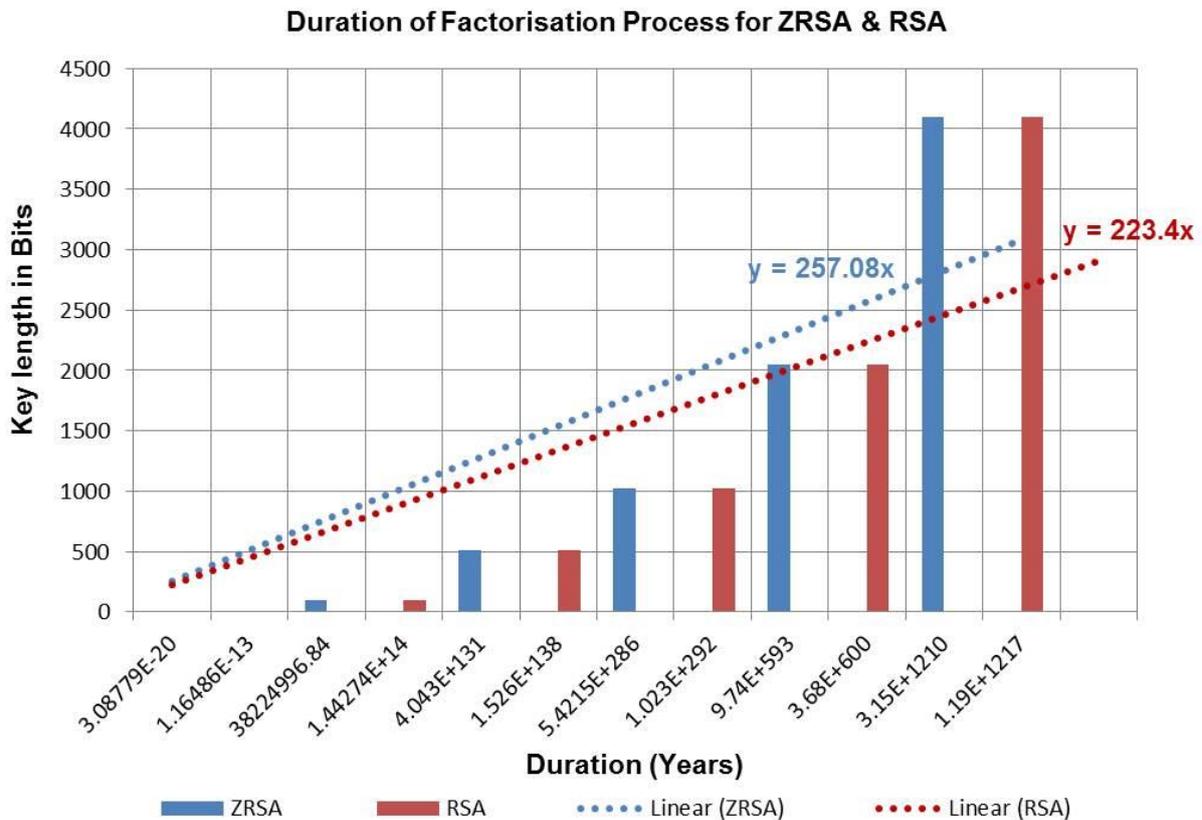


(b)

**Figure 3.2.5.6.2:** Duration of Factorisation Process for (a) RSA Key Length and (b) ZRSA Key Length

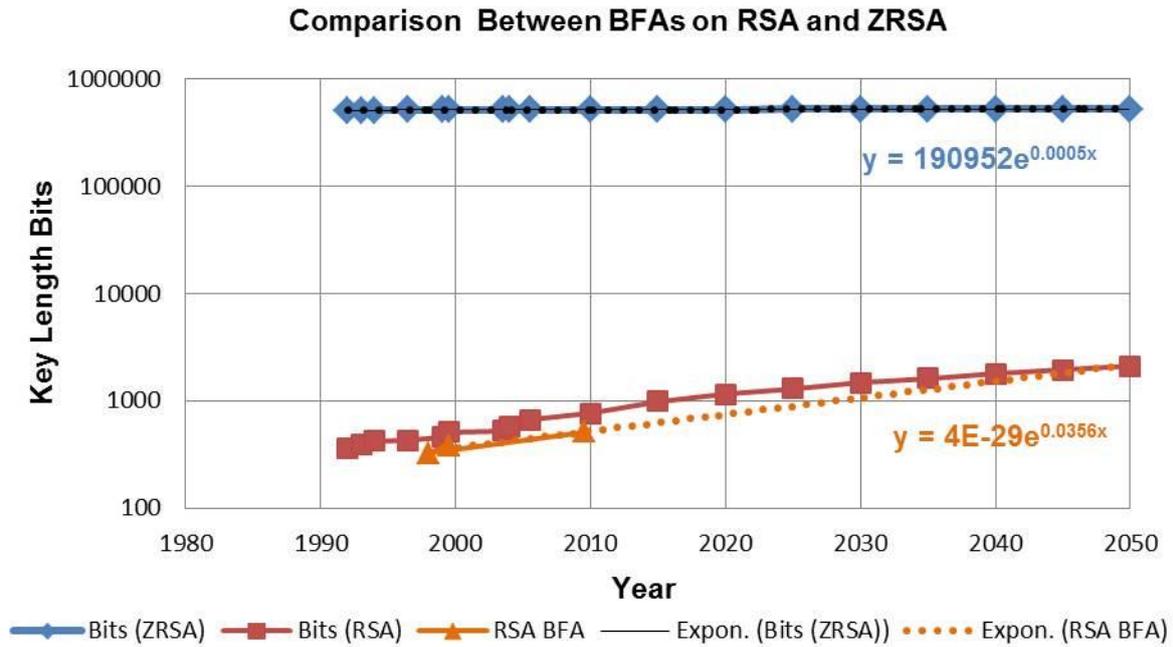
The trendline could not be derived from the individual graphs for RSA and ZRSA shown above as there were discrepancies between the duration ranges. Therefore, the most appropriate graph to represent both datasets was found to be a bar chart. This enabled an exponential

trendline to be set based on the data series with their equations displayed on the chart. From this, they can be used to represent the advances in technology of computer performance, which is shown in figure 3.2.5.6.3. The sample data of figure 3.2.5.6.3 are listed in Appendix B.



**Figure 3.2.5.6.3: RSA Key Length, Brute Force Factorisation and Trendlines**

The linear trends for RSA and ZRSA are represented by the dotted lines in figure 3.2.5.6.3. The trend shows that for the same duration, the number of bits required to factorise the key is less for ZRSA than for RSA. This means it is more difficult for hackers to factorise ZRSA. To solidify this achievement, the ZRSA trendline equation,  $y = 257.08x$ , has been used to determine the bit length required to brute force attack ZRSA, as shown in figure 3.2.5.6.4.



**Figure 3.2.5.6.4:** BFAs on RSA and ZRSA with Their Exponential trends

Appendix C presents the dataset used to plot figure 3.2.5.6.4. To determine the intersection point between the exponential trendline for RSA BFA and ZRSA the following calculations were carried out on  $y = 190952e^{0.0005x}$  and  $y = 4E-29e^{0.0356x}$ :

$$190952e^{0.0005x} = 4E - 29e^{0.0356x}$$

$$x = \frac{\ln 4.8 (E33)}{0.0351}$$

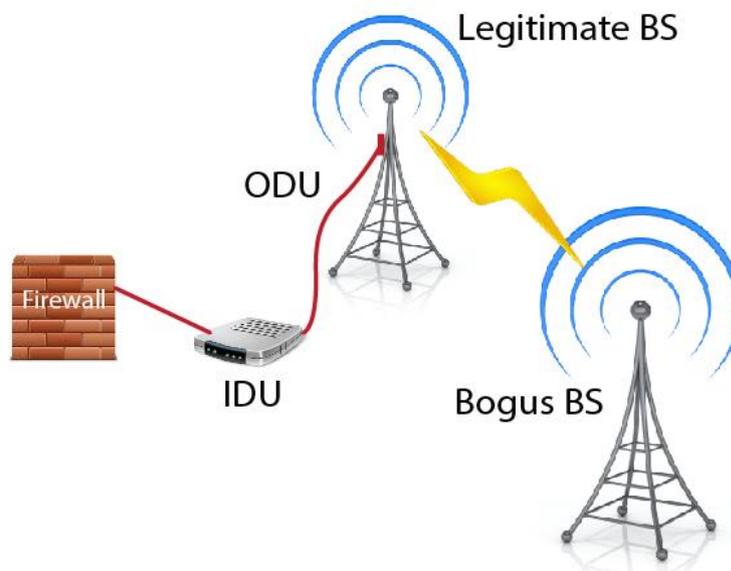
$$x = 3188$$

The bit length is so large for ZRSA that it will take hackers until the year 3188 to factorise the key. However, if RSA is going to be used hackers will achieve factorisation by 2045. This assumption is based on the QC era.

### 3.2.6 Firewall at Client's Premises

Firewalls are used to block unwanted programs and websites through filtering of network traffic. Firewalls behave as both firewalls and routers depending on the initial installation configuration to a particular firewall software. A firewall stops viruses and worms by blocking ports used by them (Li, et al., 2010). A firewall can be configured with different policies which enable them to filter access to websites and firewall resources for different users within the network. Based on their router capabilities firewalls can be used to configure virtual private networks (VPNs), which can be used to create a secure channel between networks, thereby guaranteeing confidentiality of information that is exchanged between these networks. Firewalls normally operate in the Network layer and Application layer; hence the connection has to be established before the firewall can come into the picture.

The assumption taken is that the Rogue base station is broadcasting bogus downlink map messages at a higher RSSi (Received Signal Strength Indicator) than the legitimate base station. During the initial setup of customer premises equipment of OutDoorUnit which is the antenna and the InDoorUnit which provides RJ45 ports for users to have access to the fixed network and for nomadic users, a firewall is installed with rules configured for allowing only access by legitimate base stations, based on their IP addresses. This is depicted in the network diagram shown in Figure 3.2.6.1.



**Figure 3.2.6.1:** Firewall at Client's Premises

The rogue base station will flood the client with multiple bogus messages, transmitting at a higher RSSi than the legitimate base station. Once these packets get to the firewall, the source will be identified based on the stored list of legitimate BS IP addresses and will be identified as invalid and a new scan is started for the legitimate BS. This is an idea to prevent basic DoS

attacks, for more complex attacks it might be tackled or prevented with the author's suggestion of implementing EBOC (discussed above in section 3.2.1). The reason that Firewalls are not capable of detecting all types of DoS attacks is because of the layer in which they operate, but they still provide good protection for any other attacks that are made from behind the firewall. The installation of firewalls at all clients' premises will mean more cost and hence the service will be more expensive for the client.

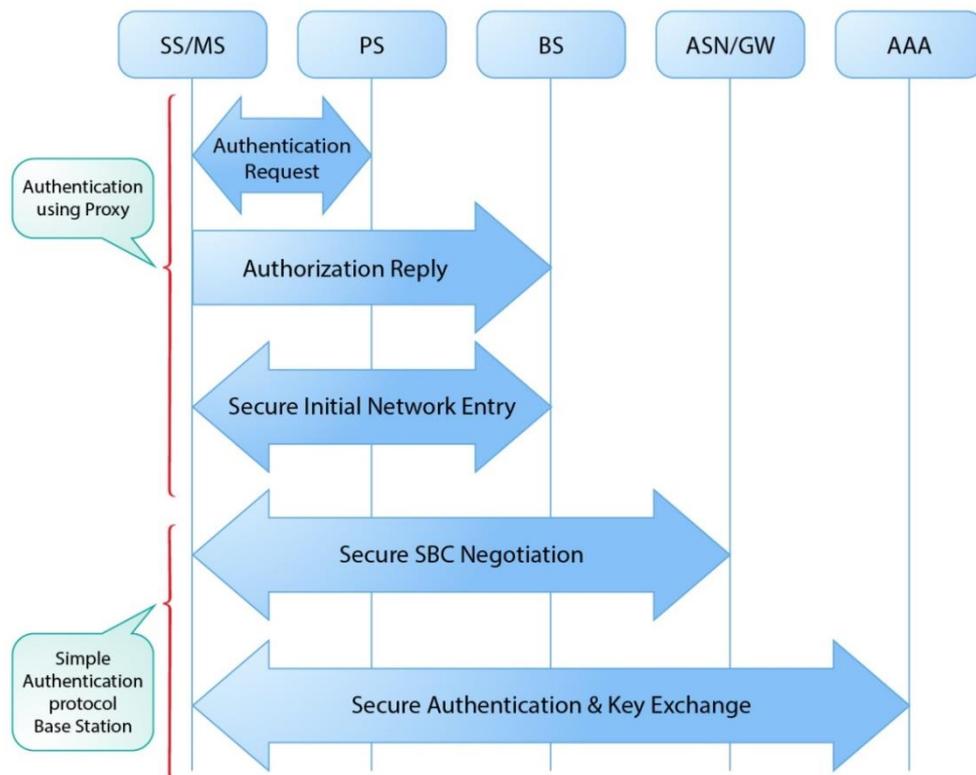
This scenario is applicable to fixed and nomadic users but not for mobile users.

### 3.3 Zaabi Security Algorithm based on Message Authentication Code for WiMAX (Z Algorithm)

Z Algorithm (ZA) is a suite of improved algorithms and procedures that are incorporated together to provide firm security management system for WiMAX networks. Formula (3.3.1) shows the main components that form the Z Algorithm.

$$ZA = HMAC + \textit{Authentication using Proxy Base Station} + \textit{Simple Authentication Protocol} + \textit{Firewall at Client's Premises} \quad \text{----- (3.3.1)}$$

Figure 3.3.1 represents Proxy Base Station, secure initialisation set up, secure SBC negotiation and secure Authentication and Key Exchange processes.



**Figure 3.3.1:** Firm Security Management System for WiMAX Network Under ZA

After PS verifies the authenticity of SS/MS, PS hands over the authorization process to the BS. This is taken care of within the Authentication using Proxy Base Station process.

During the Simple Authentication Protocol Base Station process, the Message Authentication Code security is confirmed as the MS/SS sends a service request to the BS and the BS responds with a UL-MAP message which includes the BS's X.509 certificate. The MS/SS verifies the authenticity of the BS and confirms that BS supports Simple Authentication Protocol and calculates the MAC and sends a message containing TSSI, PSK and nonce, which are encrypted with the BS's Public Key. Upon receiving the message the BS derives PEK & PAK, calculates the Message Authentication Code (MAC) and sends it to the MS/SS for confirmation.

Within the SS Basic Capability (SSBC) negotiation step, SS Requests Security Parameters from PKM authentication and registration process from BS. In return Context-Requested and Context-Report are returned from the ASN/GW (GateWay) to hand off the SBC Response to the SS to open the authentication procedure.

The Authentication and Key Exchange stage involves a secure communication to exchange the two certificates (one for the SS device & the other one is the factory certificate) and then completes the authentication process by implementing the enhanced RSA algorithm to carry out the encryption.

Therefore, ZA has prevented DoS attacks and provided a high degree of authentication and encryption to the initialisation and data transmission stages at the transport level. In addition, ZA is adding a Firewall capability at the application level to reach a good level of security to the WiMAX network.

### 3.4 Summary

The research work carried out by researchers reviewed above in this chapter, looks at the solution to Denial of Service (DoS) attacks at the BS and Securing of Management Messages where the Simple Authentication Protocol is used. The next chapter looks at the Universal Solutions to WiMAX security threats such as DoS, Securing of Management messages and masquerading as well as standard security requirements.

An enhanced version of the Message Authentication Code (MAC) algorithm has been proposed here to provide message integrity in WiMAX 802.16 standard. It incorporates the MAC function in the authentication protocol and during the sending of data. The latter part is in the form of HMAC.

RSA has been discussed and described with examples. Weaknesses of RSA have been identified. A new encryption method has been suggested, called ZRSA, mathematically discussed and supported with an example. Brute Force attacks on both RSA and ZRSA have been introduced. A comparison conclusion showed that ZRSA is superior to RSA in terms of the required factorisation time.

Next Chapter presents the details of the Certificate Authority tools of the new algorithm, Z Algorithm.

## Chapter 4: The Certificate Authority Tools of the Proposed Algorithm: Zaabi Algorithm

### 4.1 Overview

The details of the Certificate Authority tools of the proposed algorithm, Z Algorithm are presented here. As Z Algorithm uses X.509 certificate, the Certificate Authority and the Certificate Authority Control have been elaborated within this chapter. It is followed by the details of Universal Certificate Exchange in WiMAX. Versions A and B of Z Algorithm simulator flowchart are revealed in section 4.2.4 and 4.2.5. The chapter concludes with a brief summary.

### 4.2 Certificate Authority

This section examines the scenario of 'Certificate Authority' involved in the exchange of digital certificates securely and adapting the scenario to the WiMAX network's initial ranging in order to secure the management messages with novel protocols for network entry and authentication.

#### 4.2.1 Certificate Authority Control

Certificate authorities provide control over the distribution of keys and certificates in order to provide greater security overall as it will be more difficult for any unauthorized party to compromise a dedicated authority of key/certificate distribution than for the unauthorized party to compromise the security information stored with an individual party. This normally involves individuals or organizations registering with the certificate authority and requesting for the required information from the certificate authority for communicating with a particular party. The necessary certificates can be retrieved from the certificate authority and installed on the client systems or WiMAX modems during the initial setup.

#### 4.2.2 X.509

The X.509 certificate is an ITU-T recommendation which defines a directory service. The directory is usually a database of information about clients stored and made available to users who can authenticate themselves to the certificate authority in order to get the relevant information stored with the certificate authority so as to communicate with the intended party.

The X.509 standard is widely used particularly in WiMAX 802.16 for authenticating BS by MS and vice versa.

Figure 4.2.2.1 shows the X.509 certificate format.

Version
Certificate Serial Number
Algorithm
Parameters
Issue name
Not Before
Not After
Subject name
Algorithms
Parameters
Key
Issuer unique Identifier
Subject unique Identifier
Extensions
Algorithms
Parameters
Encrypted Hash

**Figure 4.2.2.1: X.509 Certificate**

The fields for X.509 are defined as follows:

- Version: describes the version of the certificate based on the fields available, version 1 is default. Version 2 has 'Issuer unique Identifier' or 'subject unique identifier' present. Version 3 has extension(s) present.
- Serial number: This is an integer value that is uniquely associated with the certificate by the certificate authority to identify by the serial integer value the certificate number.
- Signature Algorithm Identifier: This field is used to identify the algorithm used to sign the certificate and associated parameters, these details are repeated in the signature field of the certificate.

- Issuer Name: This field contains the name of the certificate authority that creates and signs the certificate.
- Period of Validity: This field indicates the validity start date and end date of the certificate.
- Subject Name: This field displays the name of the client to whom the certificate is issued and whose public key appears on the certificate as means of binding the two.
- Subject's Public Key information: This field shows the clients public key, the algorithm supported for the key and associated parameters.
- Issuers' unique identifier: It is an optional bit string field used to identify the issuing certificate authority in case the 'Issuer Name field' has been used for different entities.
- Subject unique identifier: It is an optional bit string field used to identify the client in case the 'Subject Name field' has been used for different entities.
- Extensions: These fields are available in version 3 instead of having fixed fields to ignore or consider values that can be set in the extension fields.
- Signature: This field contains the hash code of the other fields encrypted with the CA's private key; it also includes the signature algorithm identifier.

### 4.2.3 Universal Certificate Exchange in WiMAX

There is a standard format for the X.509 certificate stated by William Stallings 2011 (William, 2011) is defined as follows:

$$CA\langle\langle A \rangle\rangle = CA\{V, SN, AI, CA, UCA, A, UA, Ap, T^A\}$$

Where  $Y\langle\langle X \rangle\rangle$  = the certificate of the user X issued by certification authority Y

$Y\{I\}$  = the signing of I by Y. It consists of I with an encrypted hash code appended

V = version of the certificate

SN = serial number of the certificate

AI = identifier of the algorithm used to sign the certificate

CA = name of certificate authority

UCA = optional unique identifier of the CA

A = name of user A

UA = optional unique identifier of the user A

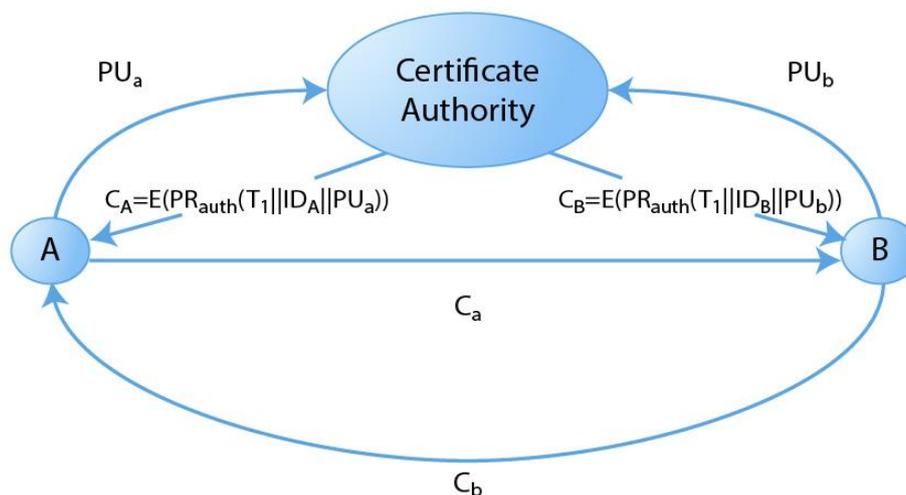
$Ap$  = public key of user A

$T^A$  = period of validity of the certificate

Basically client certificates have the following characteristics:

- A certificate issued by a certificate authority can be verified by any client in possession of certificate authorities' public key.
- Certificates issued by the certificate authority cannot be altered by anyone without it being detected.

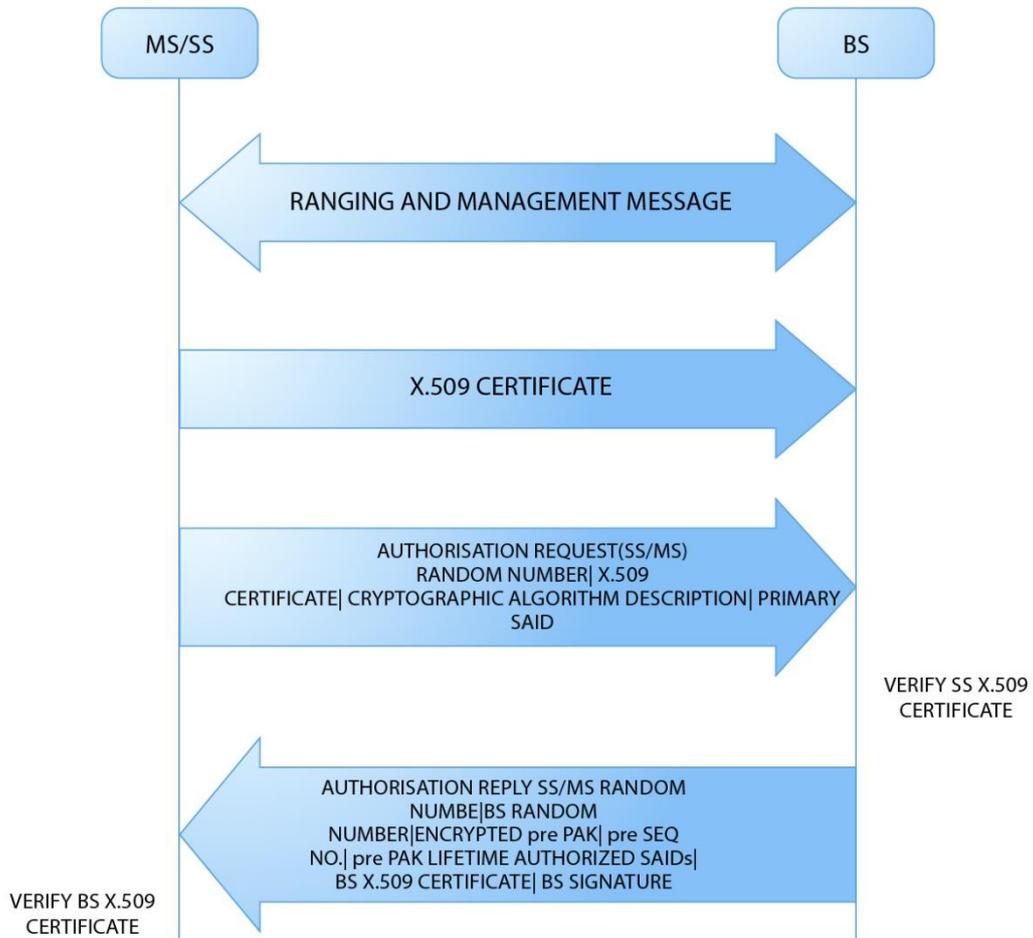
The default certificate exchange of certificates with certificate authority is shown in Figure 4.2.3.1



**Figure 4.2.3.1: Public Key Exchange**

Parties A and B exchange their public key with the certificate authority. The certificate authority generates the certificates and sends A's certificate to A and B's certificate to B. A and B exchange each other's certificate and start communication. In the case of WiMAX A is the MS and the certificate authority is BS.

Digital certificates are basically downloaded and installed with an internet connection. In the case of WiMAX the client X.509 certificate is preinstalled in the device or contains a function to installed the certificate on initial power-up. Figure 4.2.3.2 shows the exchange of the X.509 certificate during the authentication process.

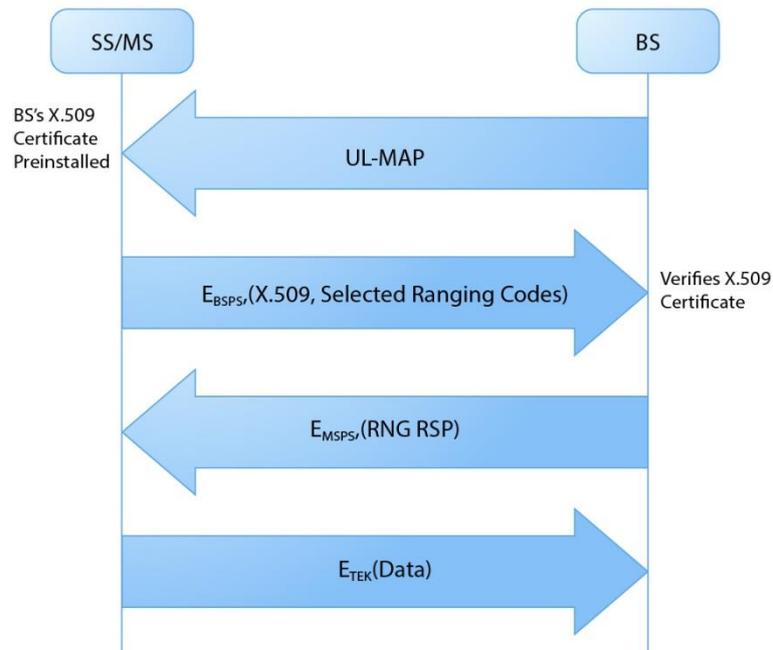


**Figure 4.2.3.2: Public Key Exchange**

Based on the authentication process shown in figure 4.2.3.2, the initial ranging messages are exchanged without any encryption and it is from this that a Denial of Service attack can be mounted by an attacker.

#### 4.2.4 Z Algorithm Version A

The X.509 certificate for the required BS in the region can be installed in the MS on commissioning. The MS can send the selected ranging codes and its X.509 certificate based on the public key of the communicating BS retrieved from the BS's X.509 certificate. The BS can send subsequent messages encrypted with the MS's public key retrieved from the X.509 certificate it receives. The protocol is shown in the ranging message flow in figure 4.2.4.1.



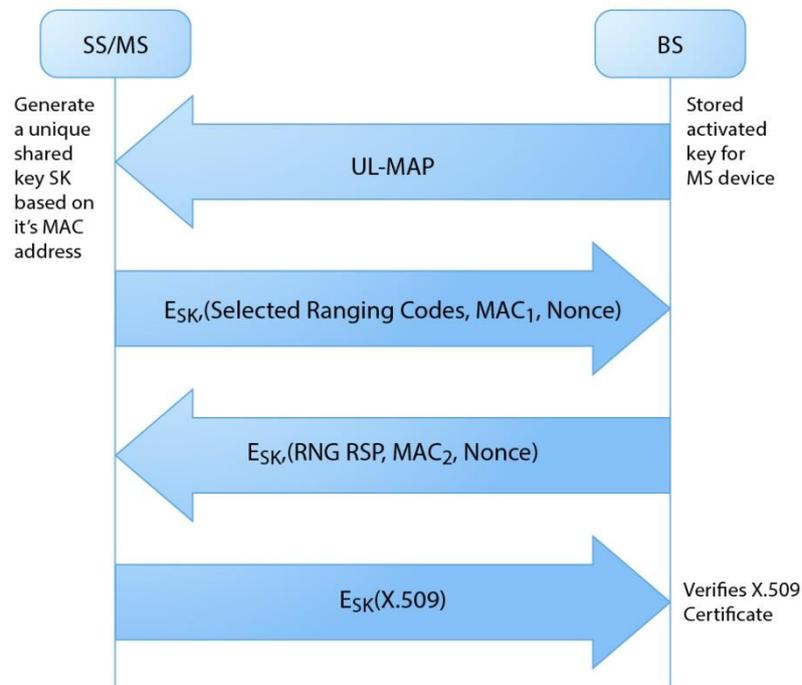
**Figure 4.2.4.1:** Ranging with X.509 Certificate

Considering the mobility of the MS the X.509 certificate can be updated automatically if it moves from one region to another by means of soft handover. It identifies the base station via GPS since the communication is happening through a secured channel. The above protocol can be adopted with Authorization key and Key Generation Key and Traffic Encryption Key being incorporated right from the management messages.

#### 4.2.5 Z Algorithm Version B

The initial encryption key could be generated by a function using the unique MAC address of the device and encrypt subsequent ranging messages with the key generated, the MS sends a RGN-REQ message a MAC function output and a nonce can be included to provide integrity and freshness. The BS can decrypt the message from a database of matching key which is identified by a function which gives identical encryption as the one received by an alleged valid device and is activated on a sever before commissioning of the client end device, the BS responds with a RNG-RSP message which includes a MAC and a nonce after verifying the

integrity and freshness of the message sent by the MS. The protocol is described in the ranging message flow in Figure 4.2.5.1.



**Figure 4.2.5.1:** Ranging messages exchange with shared key protocol

On securely exchanging the management message, the X.509 certificate is exchanged with encryption by the shared key once the X.509 certificate is received by the BS, it encrypts the authorization reply with the MS's public key. The MS upon receiving the reply continues the protocol as specified by the IEEE802.16 standard for exchange of authorization key AK, key encryption key KEK and traffic encryption keys TEKs. Z Algorithm version B uses the MAC (Message Authentication Code) cryptographic algorithm for encryption and decryption.

### 4.3 Summary

This chapter looks at required details of the Certificate Authorities that support the security management messages. The certificate authority method of securing the management messages entails a little extra financial investment and for making sure the BS certificate is pre-installed in the MS device. The shared key alternative might involve an additional database server for shared key verification involving slight extra cost, which might be a factor for determining the cost of the service. The next chapter addresses the implementation issue of the Proposed Architecture and lists & discusses the likely contribution to knowledge.

## Chapter 5: Network Security Simulator (NetSecSim) Design

This chapter discusses designing and developing the wireless network security simulator, which is used to accomplish the security requirements. In this section, a general idea about network simulators is addressed as well as the early planning of the simulator storyboard. The subsequent section shows the three main parts of the simulator, which are initialisation signal, WiMAX DoS attack and solution to overcome it. Each section covers all the details in depth.

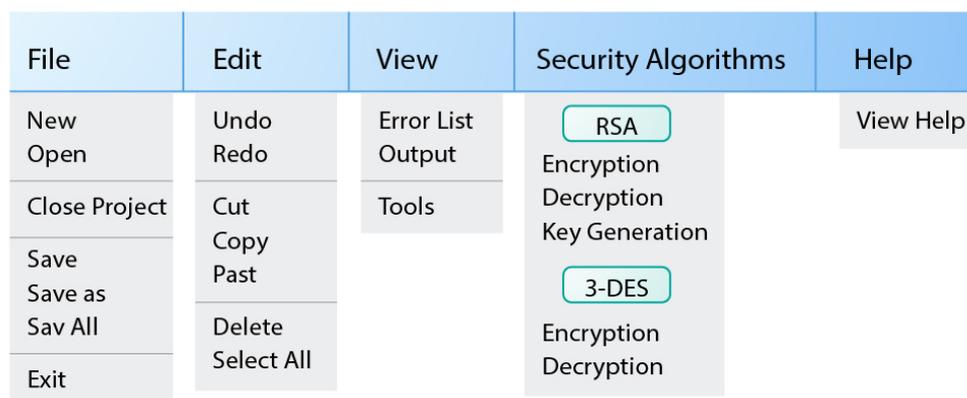
### 5.1 Simulator Overview

The development of the simulator has gone through a set of stages, starting from concept through to implementation. The concept and design is laid down in subsection 5.1.1 and the flowchart of the implementation is explained in subsection 5.1.2.

#### 5.1.1 Simulator Concept and Design

The wireless network simulator is designed and developed using C# language in Microsoft Visual Studio. Typically, simulators have very good performance for specific scenarios without the need for the hardware to be developed. One aim in mind is that, network simulators are used to check system behaviour.

NetSecSim simulator is designed to simulate the security issues of WiMAX 802.16e. However, it only covers one type of attack; that is the DoS attack and offers a new cryptography algorithm, ZRSA, which is based on RSA. Some features of using network simulators are cost-effectiveness and short time for simulation development. This simulator is focused on the point-to-multipoint operation. Figure 5.1.1.1 shows the storyboard of the simulator, which was designed at the early stage of the project.



File	Edit	View	Security Algorithms	Help
New Open	Undo Redo	Error List Output	<input type="checkbox"/> RSA Encryption Decryption Key Generation	<input type="checkbox"/> View Help
Close Project	Cut Copy Paste	Tools	<input type="checkbox"/> 3-DES Encryption Decryption	
Save Save as Sav All	Delete Select All			
Exit				

WiMAX Simulator Storyboard

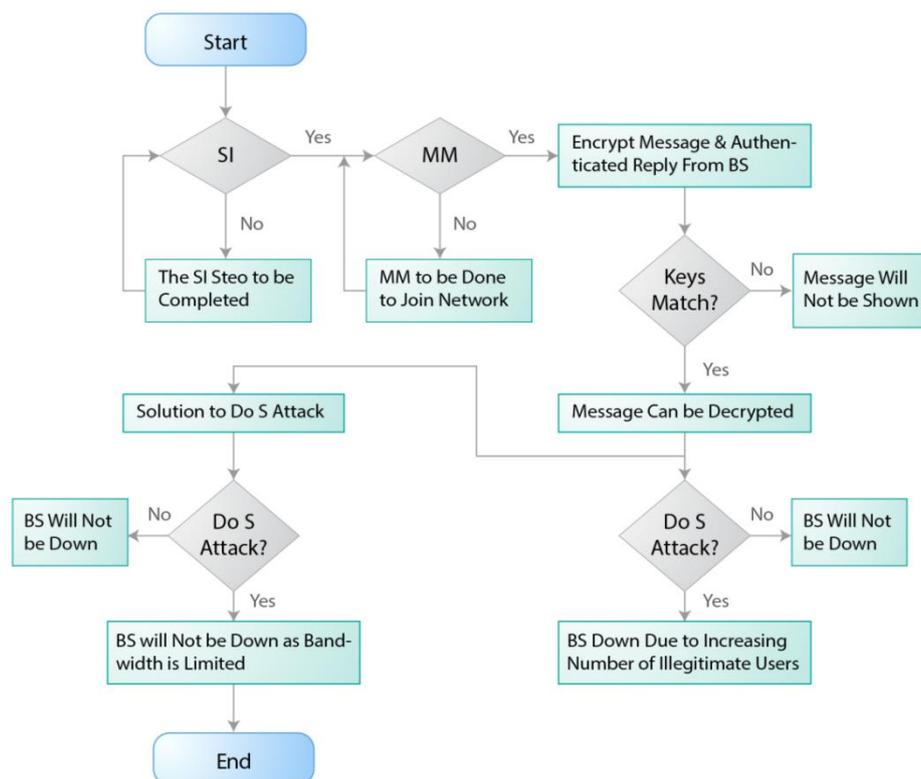
**Figure 5.1.1.1:** NetSecSim Simulator Storyboard

During the project period, the simulator was improved to address various functions, as follows:

- Management message procedure at the first step of the simulation, which is the initialisation process.
- To provide more security to the simulator, RSA, ZRSA, modified ZRSA and 3DES algorithms were used for scrambling the messages and adding authentication to the message, respectively.
- Finally, to show how an attack could happen and how to overcome it.

### 5.1.2 Simulator Flowchart

The coding implementation in C# of the simulator followed a few processes that depict the possible requirements for a network user who would like to simulate some of the security parameters associated with wireless networks. The wireless network that has been chosen, as discussed in chapter 1 and 2, is WiMAX based on the IEEE 802.16e standard. To follow the development of the code with ease, use of the simulator and to track the next sections of this chapter, a flow chart is presented in figure 5.1.2.1.

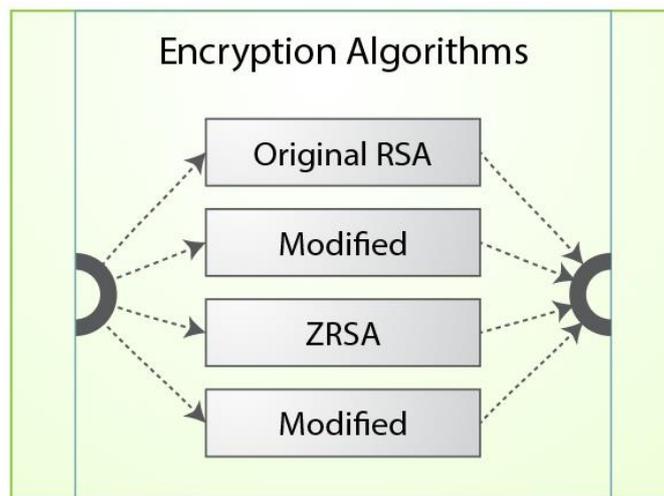


**Figure 5.1.2.1: NetSecSim Flowchart**

The following description on NetSecSim flowchart is based on five decision units, starting with the first one which is called SI (Signal Initialisation). This is an essential step where messages

are exchanged between SS and BS to establish the connection between both parties. The simulator has implemented the 11 SI steps that IEEE 802.16e protocol has dictated to establish a healthy connection. So, SI establishes a communication between BS and SS if the 11 messages are complete. The SI decision unit has been coded to question BS and SS accordingly at each message, if the answer is satisfactory, the loop moves on 11 steps up until it pronounces the completion of the initialisation.

The next step is the MM (Management Messages) decision unit. Its purpose is to allow SS to join the network. If SI step is completed, similarly, the MM follows the IEEE 802.16e protocol and exchange further messages until this process is completed before the simulator moves on to the next one, the Encryption and Authentication step. All crypto analysis algorithms are C# coded within the Encryption and Authentication step. Figure 5.1.2.2 lists the 4 implemented encryption algorithms.



**Figure 5.1.2.2:** Implemented Encryption Algorithms in NetSecSim

The full description of RSA, modified RSA, ZRSA and modified ZRSA is given in Chapter 3. The first three algorithms, original RSA, modified RSA and ZRSA, generate one public key and one private key, while the fourth one, Modified ZRSA generates 3 public keys and 3 private keys. NetSecSim has been designed to display the generated keys. One of the encryption algorithms must be selected. The encryption algorithms in NetSecSim are explained expanded and supported with simulator results in section 5.4.

The third decision unit is the Key Match. There are sub-processes in this step/unit as the SS waits for the authentication key (AK) and authenticated reply from BS.

Once SS receives the authenticated reply the message can be decrypted unless the keys do not match at both ends. That is determined by the fourth decision unit, the “Dos Attack?” Currently, the Network is down if a DoS attack takes place.

The new solution of the author to tackle DoS attack is demonstrated with the fifth and last decision unit. If the new approach is applied, BS will not accept more users as the number of illegitimate users added by the attacker is limited with the allowed bandwidth. Hence, NetSecSim is allowing legitimate users to join without denial of service.

There are three main sections in this chapter of the report, initialisation process, WiMAX DoS attack and the solution to overcome it.

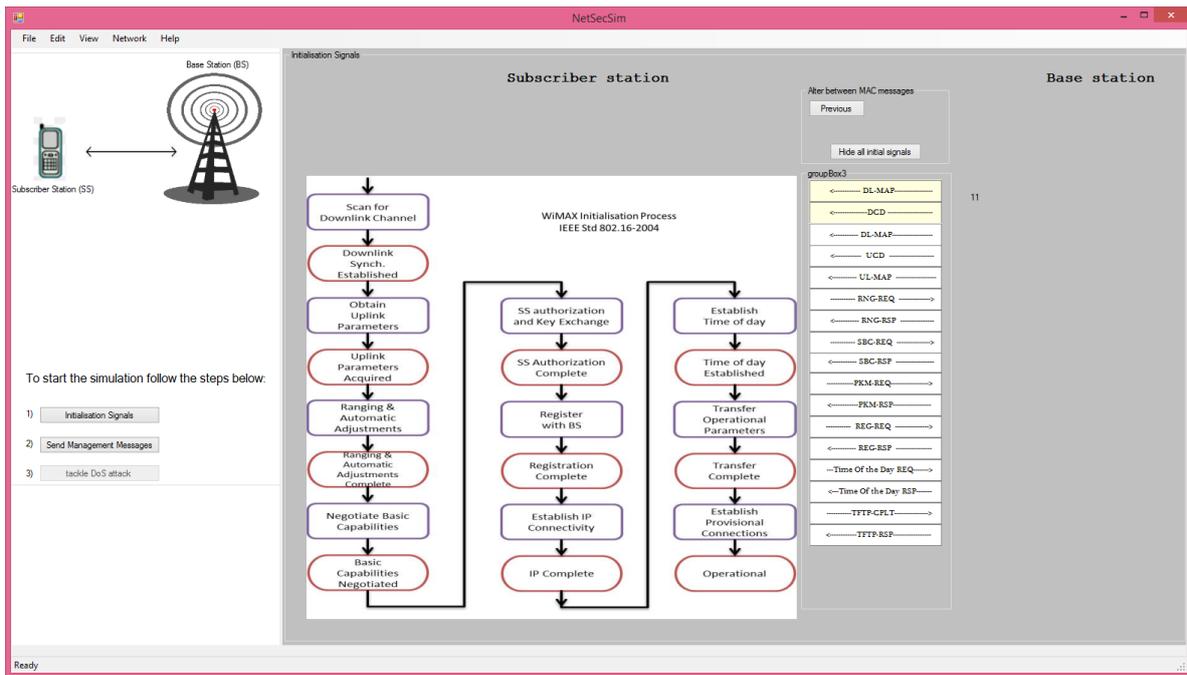
## 5.2 Initialisation Process

Signal initialisation is an important stage in the wireless network protocols such as WiMAX and LTE. The initialisation processes perform a number of important functions similar to terminals in WiMAX that exchange security capabilities and authorization information.

There are three different level of management connections; basic, primary and secondary connections. The first connection, basic management, is used by the BS and SS MACs to exchange short, time urgent MAC management messages. For longer, more delay-tolerant MAC management messages, the primary connection is used. If the primary connection is successful, BS and SS use the secondary management connection. This connection is used to allow standards-based messages. These standards are DHCP, TFTP, SNMP etc. The secondary management messages are not MAC management messages but they are used only by the managed SSs (Nuaymi, 2007).

As the basic and the secondary connections are not directly related to the security weaknesses in a wireless network, NetSecSim is implementing only the primary connection, the management messages connection.

Figure 5.2.1 shows the window of the signal initialisation during simulation. These messages contain the important details for establishing the connection between SS and BS.



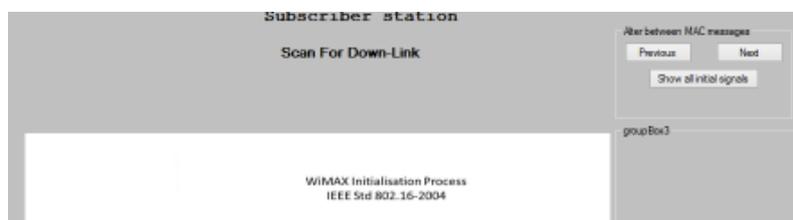
**Figure 5.2.1: Signal Initialisation Messages**

NetSecSim's initialisation processes conform to one of the wireless networks protocols; the chosen one for NetSecSim is WiMAX.

As can be seen in figure 5.2.1, there are a number of signals. The following sub sections discuss each of the signals shown above.

### 5.2.1 Scanning and Synchronisation

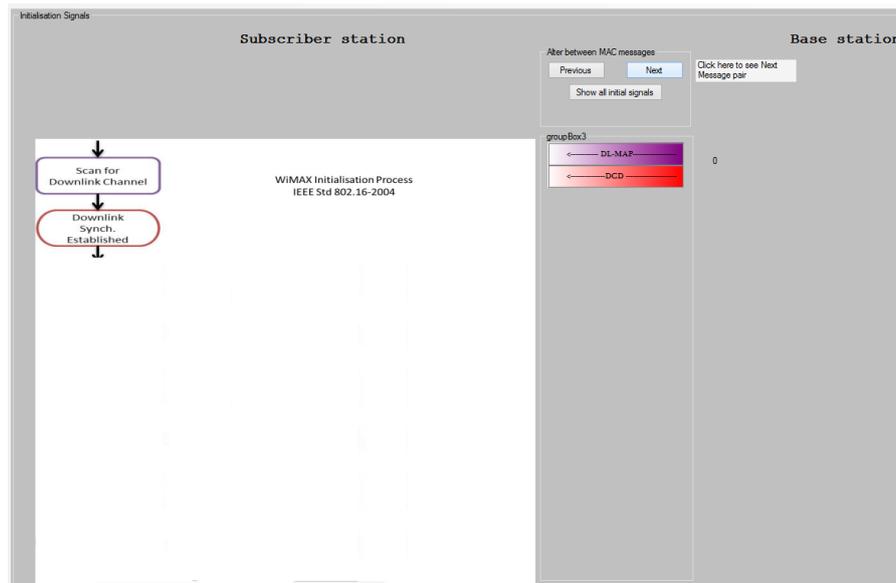
SS attempts to achieve a downlink channel. In case of signal loss or signal initialisation, there is a non-volatile memory equipped with SS in which the last downlink channel parameters are saved. SS attempts to recover the saved parameters of the downlink channel. However, if SS cannot recover them, it scans for any downlink channel available. Figure 5.2.1.1 shows the scanning process in the simulator.



**Figure 5.2.1.1: Scanning Process**

## 5.2.2 Achieving Downlink Channel

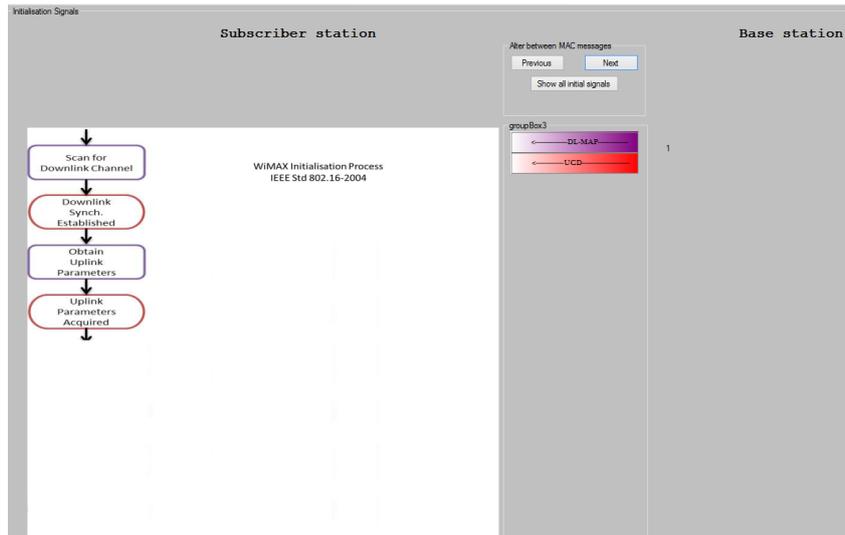
Scanning for downlink map (DL-MAP) MAC management messages is done through the MAC. MAC synchronisation is obtained once SS obtains at least one DL-MAP message. If SS keeps obtaining DL-MAP and a downlink channel descriptor, DCD, the MAC will continue being synchronised. However, if MAC can no longer receive suitable DL-MAP and DCD, SS has to redo the synchronisation process. Figure 5.2.2.1 shows SS obtaining the DL-MAP and DCD during WiMAX simulation.



**Figure 5.2.2.1:** DL-MAP and DCD Messages

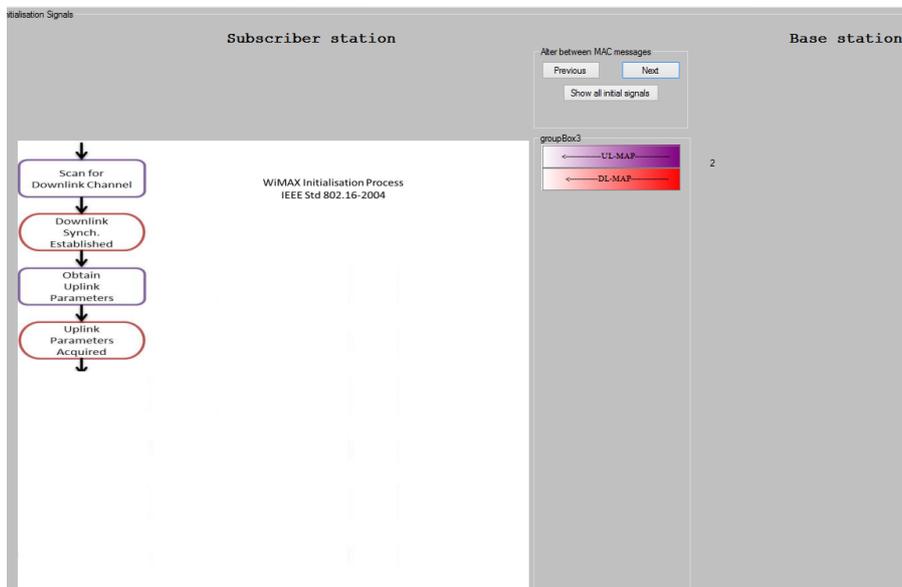
## 5.2.3 Achieving Uplink Channel

After the synchronisation process, the SS receives an uplink channel descriptor, UCD, as shown in figure 5.2.3.1. The BS sends the messages at regular intervals to the accessible uplink channels. These messages are addressed to the MAC broadcast address. There is a certain time limit for searching for an uplink channel, and if one is not found then SS again scans for a downlink channel. The SS verifies whether the uplink channel is usable or not from the description parameters of the channel. If the channel is not usable, SS has to scan for a new downlink channel. But, if the channel is usable, UCD parameters can be extracted by the SS.



**Figure 5.2.3.1: DL-MAP and UCD Messages**

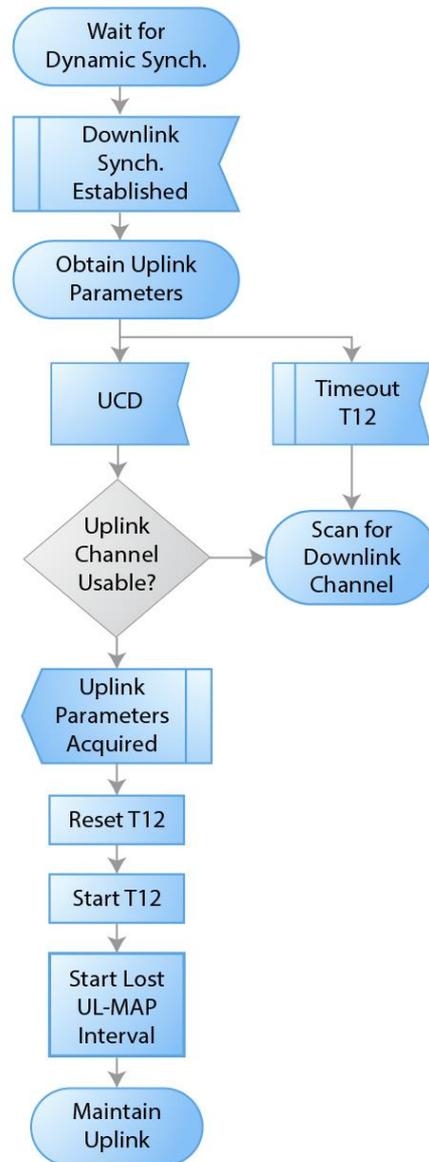
After that, in the next pair of messages the SS receives an uplink map UL-MAP and DL-MAP. In other words, SS waits for a bandwidth allocation for that particular channel. According to the mechanism of both bandwidth allocation and MAC operation, it might begin transmitting uplink. Figure 5.2.3.2 shows the UL-MAP along with the DL-MAP to extract the time synchronisation.



**Figure 5.2.3.2: UL-MAP and DL-MAP Messages**

During the process of obtaining the uplink channel parameters, an initial ranging process takes place to check the initial ranging period. In order for the SS to check the period of the initial

ranging, it scans the UL-MAP. The flowchart shown in figure 5.2.3.3 demonstrates how SS achieves the uplink channel parameters.



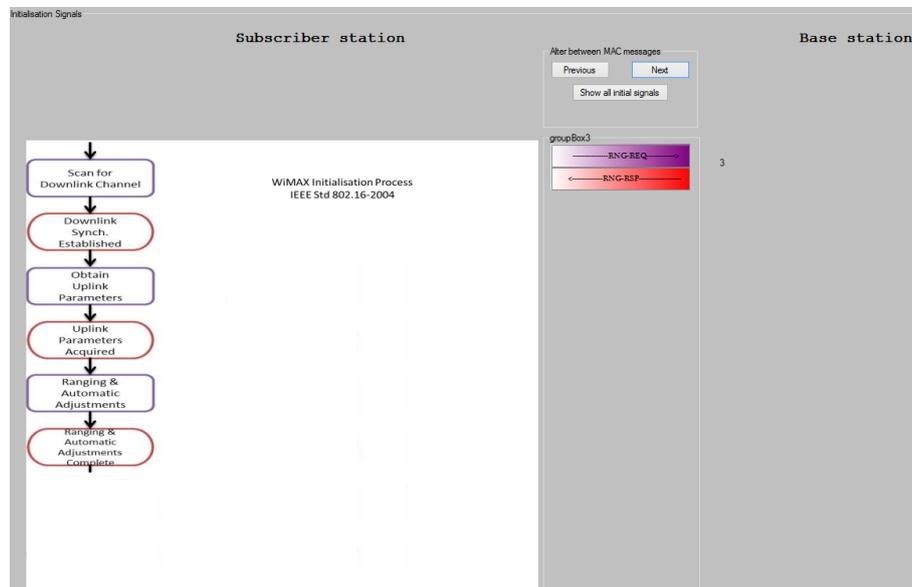
**Figure 5.2.3.3:** Achieving uplink channel parameters

Timer T2 is used at SS, it waits for broadcast ranging timeout up to maximum duration “5\* ranging interval. Similarly, timer T12 at SS, waits for UCD descriptor up to maximum duration of “5\* UCD interval. During this period, uplink parameters are acquired.

### 5.2.4 Initial Ranging

The purpose of this process is to achieve both power adjustment and timing offset. The SS transmits the RNG-REQ message in an initial ranging period.

As shown in figure 5.2.4.1, at this stage a ranging request (RNG-REQ) and ranging response (RNG-RSP) are sent and received between both SS and BS. The identity of the tool is identified by the 48-bit MAC address.



**Figure 5.2.4.1: RNG-REQ and RNG-RSP messages**

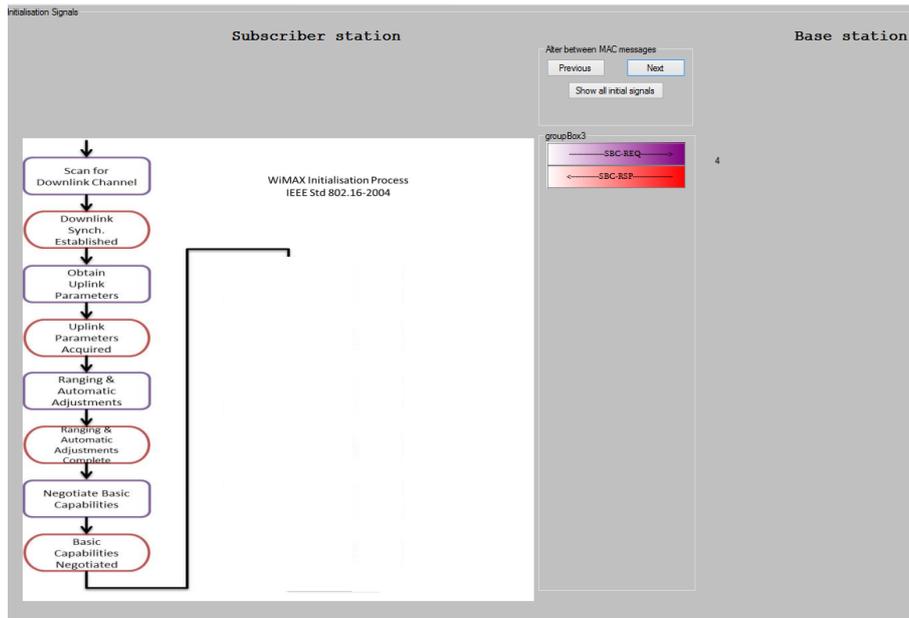
When transmitting the message, the CID value should be set to zero as for non-initialised SS. The ranging helps in correcting the timing offset for SS specified to demonstrate to the BS that it has been collocated (located). In order for the SS to be collocated next to the BS, it has to adjust the timing offset to match the internal delay.

The maximum strength of the transmitted signal should be computed by the SS. Also, RNG-REQ message should be sent using the minimum transmit power level specified by the BS. If the SS does not obtain the ranging response RNG-RSP from BS, then SS transmits another RNG-REQ using the maximum transmit power level specified by the BS. If the RNG-REQ reaches the BS then it responds with RNG-RSP using the CID value. The RNG-RSP message contains both CID values, basic and primary management, which are allocated to the corresponding SS.

### 5.2.5 Basic Capabilities

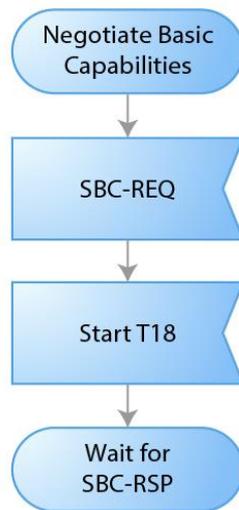
Within this process, SS informs BS of its basic capabilities. SBC-REQ (SS Basic Capability Request) is transmitted by SS and received by BS. In turn, BS responds with SBC-RSP REQ (SS Basic Capability Response) message.

Figure 5.2.5.1 shows the SBC-REQ and SBC-RSP messages during the simulation.



**Figure 5.2.5.1: SBC-REQ and SBC-RSP messages**

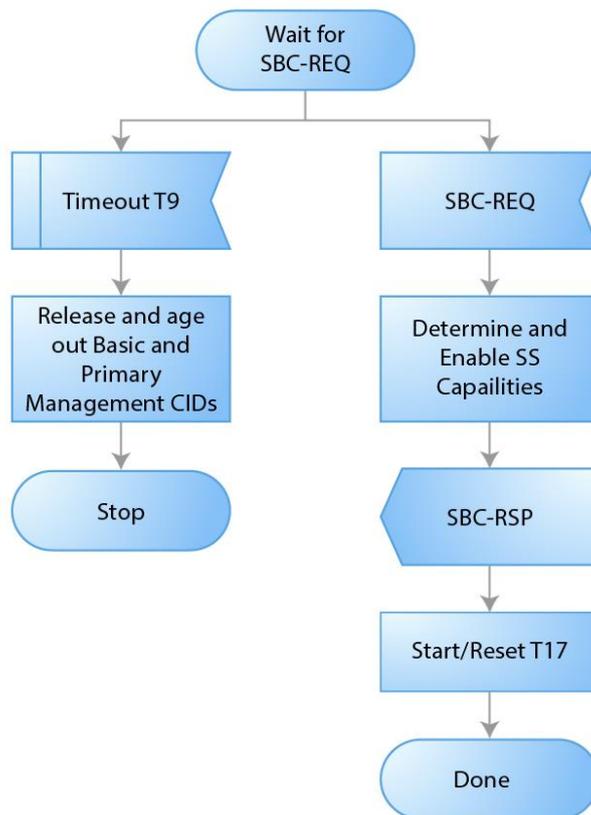
Instantly after completing the initial ranging process, SS sends an SBC-REQ message to notify BS of its basic capabilities. SS capabilities in the message are shown in figure 5.2.5.2.



**Figure 5.2.5.2: Basic Capabilities (SS)**

Timer T18 is used at SS, waits for SBC-RSP timeout for default 50ms and maximum is less than the registration timeout, T9.

BS transmits a response message to SS, as shown in figure 5.2.5.3, which is SBC-RSP containing the intersection of both SS and BS capabilities.



**Figure 5.2.5.3: Basic Capabilities (BS)**

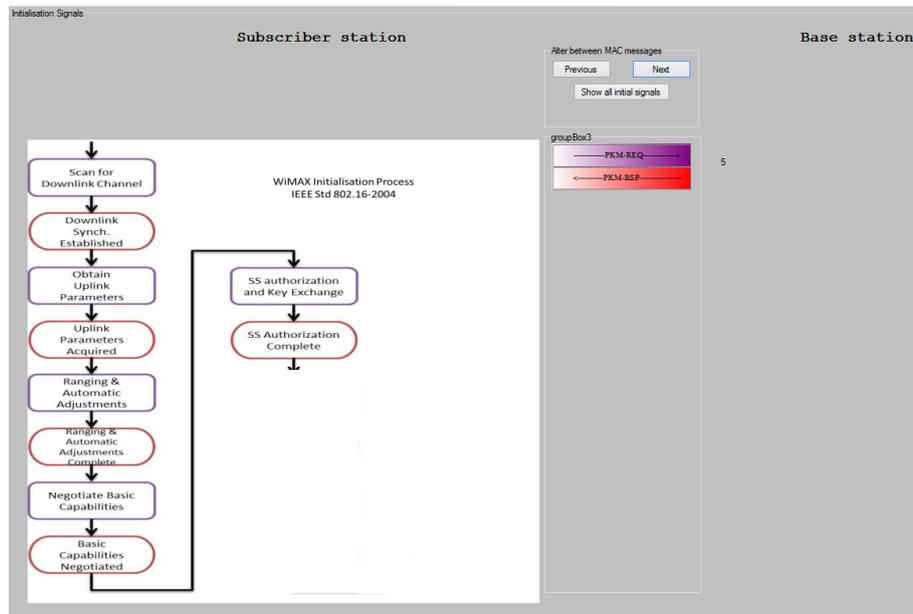
Timer T17 is located at BS to allow for SS to complete authorization and key exchange for minimum 5 min where the default value is 5 minimum. Timer T9 is again used at BS and is called Registration Timeout. It is the time allowed between BS sending a RNG RSP (success) to an SS, and receiving a SBC-REQ from the same SS within minimum and default value of 300ms.

### 5.2.6 SS Authorisation and Key Exchange

In order to verify the authentication of the MAC digest messages received from BS at the early stage of the initialisation process, an authentication key (AK) has to be requested by sending a privacy key management request (PKM-REQ) message to the correspondent BS. After sending the request message, the BS sends a response message to the SS, which is the PKM-RSP message that contains the AK. In NetSecSim C# code, a special class has been developed to simulate authorisation and key exchange as presented within NetSecSim flowchart in previous subsection within Chapter 5.

The SS has to keep the AK secured by storing it in the non-volatile memory. For the BS to send the AK to SS securely, it encrypts the key using the aforementioned 4 encryption algorithms, namely, RSA, modified RSA, ZRSA and modified ZRSA. As mentioned in the

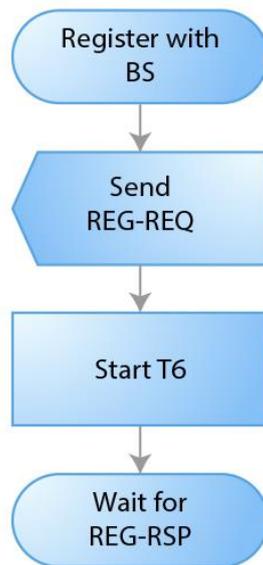
literature review, it is required that any of the four encryption algorithms uses two keys; therefore BS encrypts the AK with SS's public key (in case of modified ZRSA, three keys). The BS then obtains the public key of SS by checking the digital certificate for SS that was sent to BS at the start of the initialisation process. Section 5.3.2 detailed the exchange of the keys between BS and SS. Figure 5.2.6.1 below illustrates NetSecSim simulator view of this stage.



**Figure 5.2.6.1: PKM-REQ and PKM-RSP messages**

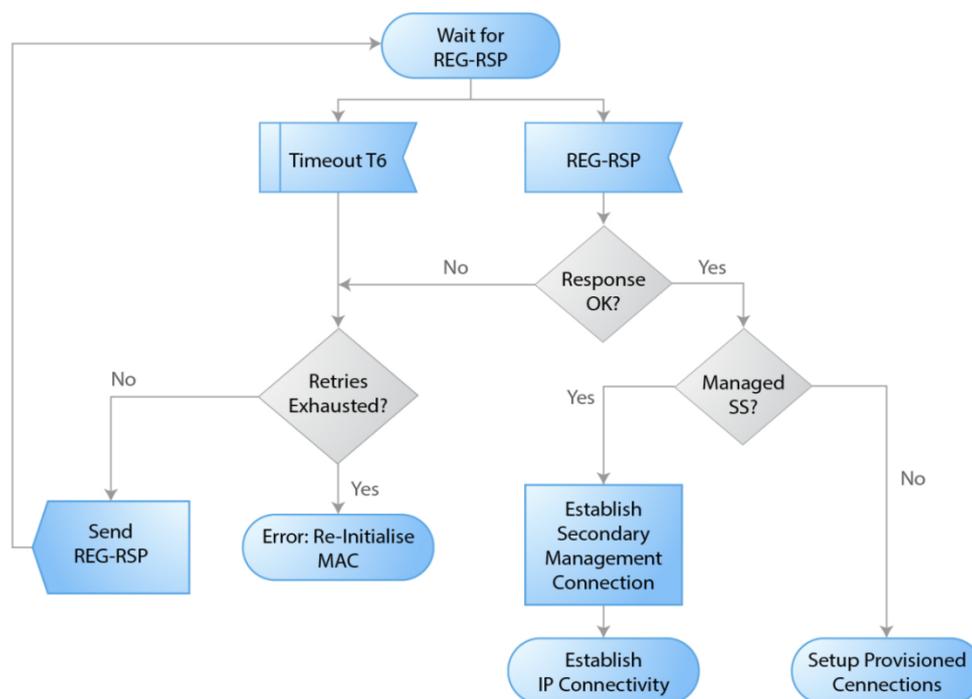
### 5.2.7 Registration

Registration is the procedure of a registration request (REG-REQ) message being sent by SS to the BS in order to be authorised to join the network. A manageable SS receives the secondary management CID if the message being sent by the SS shows that SS is being managed. BS then responds with an REG-RSP message, which contains the secondary management CID. The process of registration is depicted in figure 5.2.7.1.



**Figure 5.2.7.1: Registration (SS)**

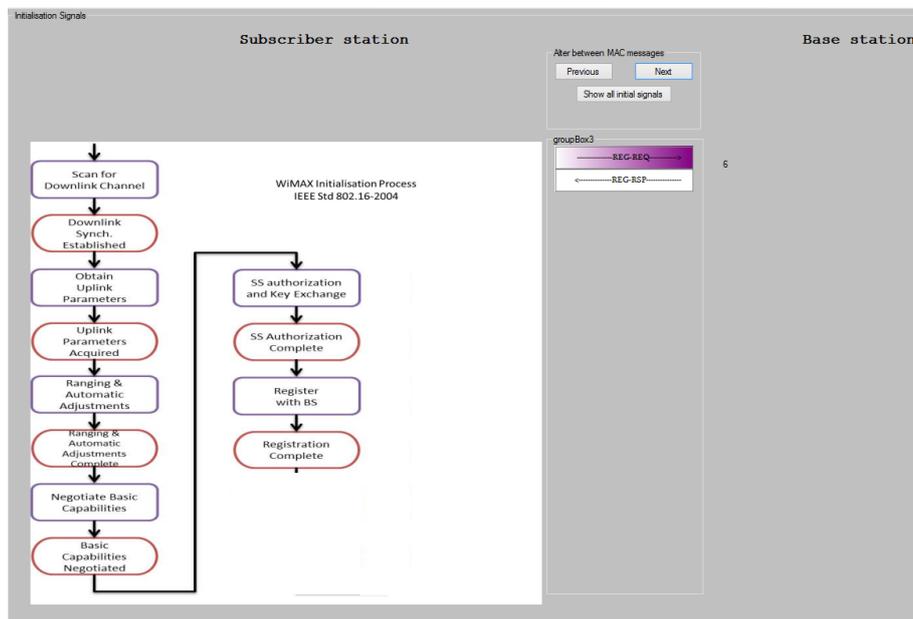
When SS transmits the request message to BS, it waits until BS responds. Figure 5.2.7.2 shows the waiting for registration response. T6 timer is used at SS within the REG-RSP process. It waits for registration response for maximum duration of 3sec.



**Figure 5.2.7.2: Wait for REG-RSP**

While sending the registration request message, SS can specify which IP version it supports on the secondary management connection. When SS specifies the IP on REG-REQ, BS will

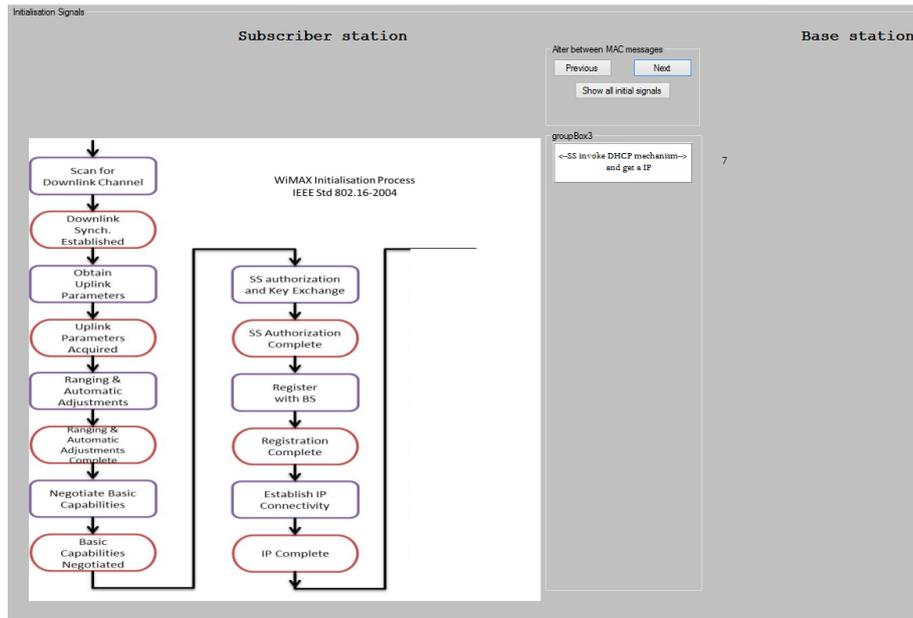
include the parameters for the same version of IP along with the REG-RSP for the secondary management connection. Figure 5.2.7.3 shows both REG-REQ and REG-RSP messages while simulating.



**Figure 5.2.7.3: REG-REQ and REG-RSP messages**

### 5.2.8 Initiate IP Connectivity

In this part of the initialisation process, the SS invokes dynamic host configuration protocol (DHCP) mechanisms. The reason for this is to achieve an IP address and any parameters needed to initiate IP connectivity. A configuration parameters file is included in DHCP response if the SS has a configuration file. Initiation of IP connectivity is done on the secondary management connectivity of SS. Figure 5.2.8.1 illustrates the message of DHCP while simulating.

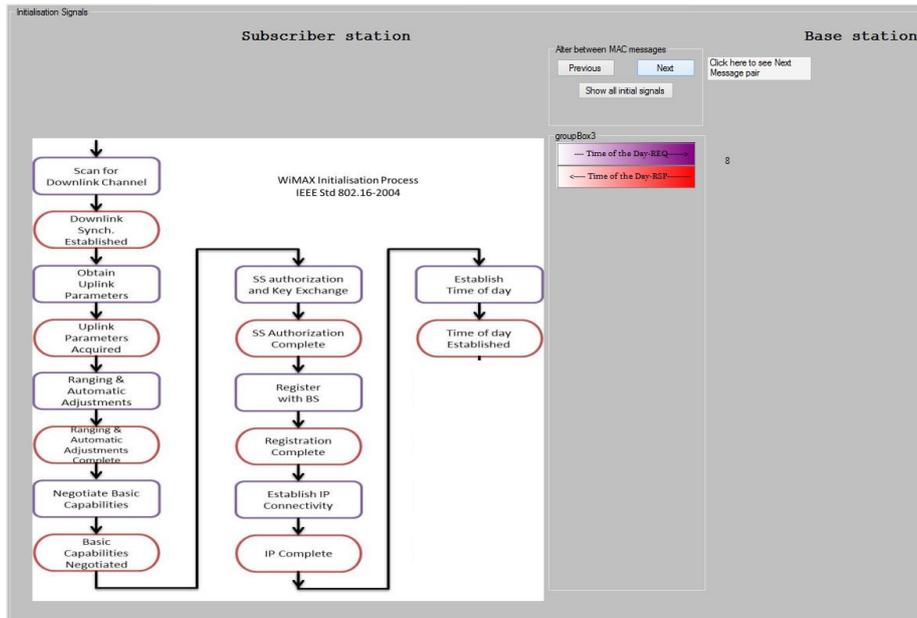


**Figure 5.2.8.1: DHCP message**

### 5.2.8.1 Initiate Time of Day

Figure 5.2.8.1.1, shows the time of day of the request and response messages and the following paragraph explains the process. The management system requires both BS and SS to have the current time of day for recovery of time-stamping logged events.

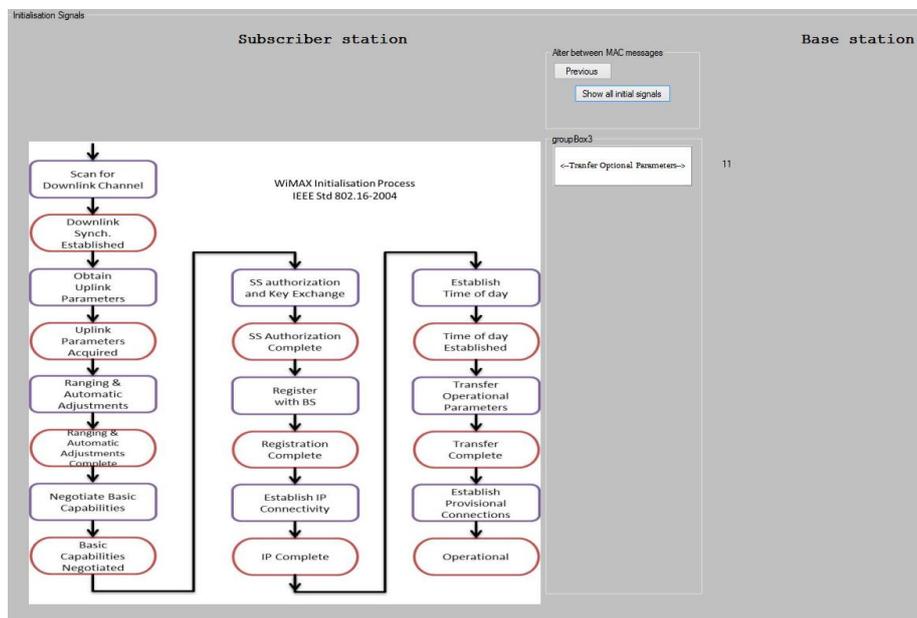
Time of day must be precise to the nearest second, but does not need to be authenticated. User datagram protocol UDP is utilised when sending the time of day request message and receiving the time of day response, combining the recovery of the time from the server (coordinated universal time UTC) with the time offset obtained by the DHCP to generate the present time. This step of the initialisation process is done based on the secondary management connection of the SS. The response of the time of day message is not based on a successful registration process. However, it is important for the uncompleted operation. SS should not send more than three requests in a period of five minutes, as the time of day request is an implementation dependent.



**Figure 5.2.8.1.1: Time of day request and response**

## 5.2.10 Transfer Operational Parameters

In figure 5.2.10.1, the last two pair of messages of this part of the simulator are shown.



**Figure 5.2.10.1: TFTP-CPLT and TFTP-RSP**

In this step, the configuration file is downloaded by the SS's secondary management connection using TFTP after successfully completing the DHCP process. The SIADDR (Server Ip ADDRESS) field is the DHCP response state of the TFTP configuration file. A "binary exponential backoff" is needed for the SS to utilise an adaptive timeout. After successfully

downloading the configuration file, a TFTP-CPLT message is sent by SS to inform BS of the SS's primary management connection. SS keeps sending the message until BS responds with a TFTP-RSP which contains "OK", or SS stops retransmission because of exhaustion.

### 5.3 MAC Management Messages in WiMAX

This is the main window of the NetSecSim simulator, which follows on from the signal initialisation window. The WiMAX/802.16 standards manage, maintain and control the communication between both SS and BS through a defined 46 MAC management messages. The following figure, figure 5.3.1, shows the simulation view.

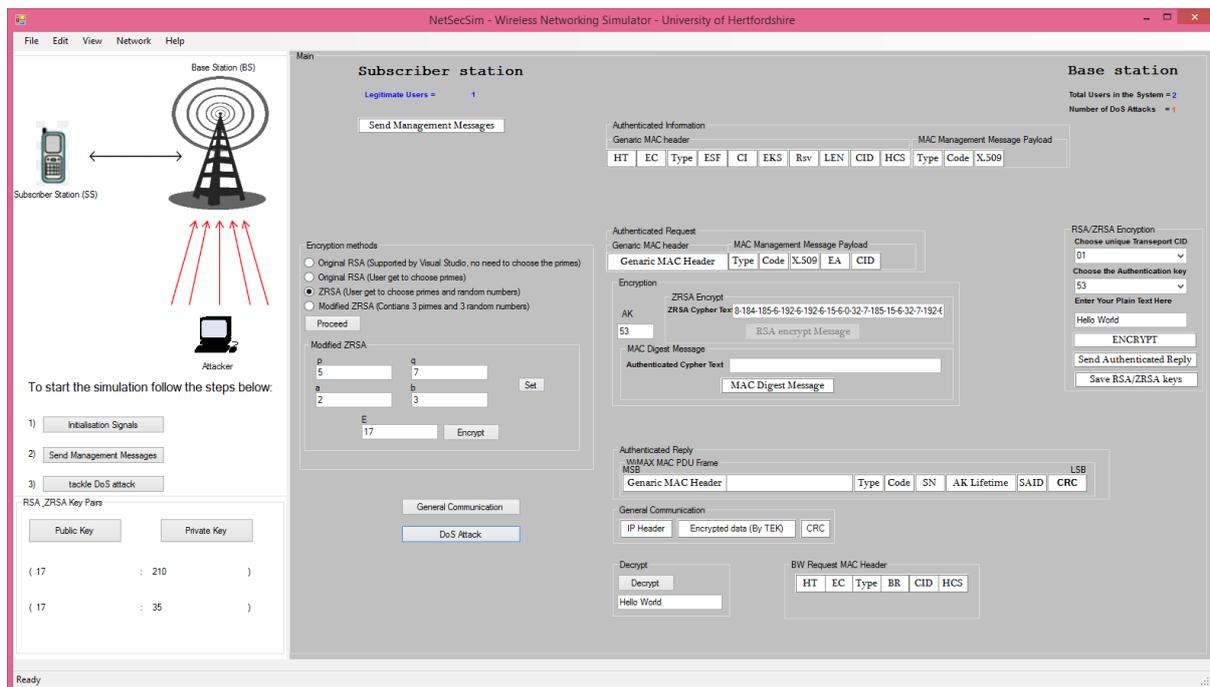


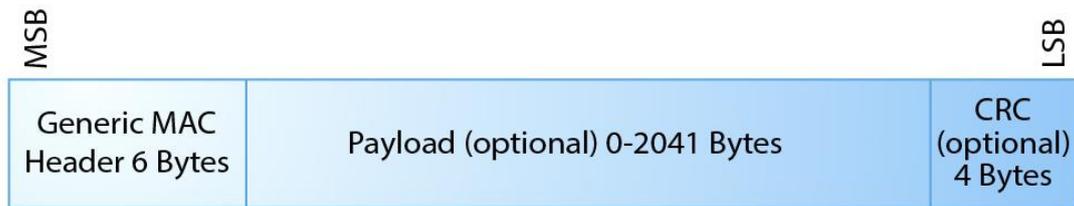
Figure 5.3.1: MAC Management Messages

As can be seen from figure 5.21, there are two main parts of MAC management messages; i) Authenticated Information and ii) Authenticated Request. In each one, there are two parts; i) Generic MAC Header and ii) MAC Management Message Payload. In both parts, there are a number of fields. All fields will be explained in this section of the thesis.

#### 5.3.1 MAC PDU Formats

Each MAC PDU (Protocol Data Unit) starts with a fixed length generic MAC header. A MAC PDU payload, if it is used, follows the header. If so, zero or more subheaders and MAC SDU (Service Data Units) are included in the payload. The information of payload might differ in length. As a result, a variable number of bytes might be shown by the MAC PDU. This results

in the MAC channelling a higher layer of different traffic types as it does not know the bit prototypes or formats of the messages. Figure 5.3.1.1 depicts the form of MAC PDU.



**Figure 5.3.1.1: MAC PDU Form**

### 5.3.2 MAC Header Formats

There are two different types of MAC header formats. A Generic MAC Header (GMH) located at the beginning of every MAC PDU, including convergence sublayer SC data or MAC management messages. The other type is the Bandwidth Request Header (BRH), which is utilised to demand more bandwidth. The first field is called HT, header type, determine the header type while the second field called EC, encryption control, is it is set, encrypt the payload.

Table 5.3.2.1 shows the description of each field in the generic MAC header.

Name	Length (bits)	Description
HT	1-bit	Header Type is a single bit field used to differentiate between both formats' bandwidth, request header and generic MAC header For Generic MAC header = 0 For Bandwidth request header = 1
EC	1-bit	Encryption Control When payload is not encrypted = 0 When payload is encrypted = 1
Type	6-bits	The type field is used to show the unique types of payload and subheaders in the current message payload
ESF	1-bit	Extended subheader field is present when ESF = 1, and not present when ESF = 0. It always appears immediately after the generic MAC header and before other subheaders, it is valid in DL and UL, and every extended subheader is not encrypted

CI	1-bit	CRC indicator, if it = 1, then CRC is contained in the PDU by adding it after the encryption to the PDU payload; if it = 0, then CRC is not contained in the payload
EKS	2-bits	Encryption key sequence is the traffic encryption key (TEK) indicator. In addition to the initialisation vector utilised for payload encryption, the EKS field is important when the EC field is set to 1
Rsv	1-bit	In the meantime, this field is reserved by the 802.16 standards for future improvements
LEN	11-bits	This field represents the MAC PDU as well as the MAC header and CRC length in bytes
CID	16-bits	This is the connection identifier
HCS	8-bits	Errors in the header are spotted by the use of the header check sequence field. The first five bytes value of the cell header of the HCS field are calculated by the transmitter, then the result is placed in the last byte of the MAC header in the HCS. This is the remainder of the division of modulo2 by the generator polynomial

**Table 5.3.2.1: Generic MAC header fields**

After the generic MAC header comes the MAC management messages payload, which consists of three different fields, as shown in table 5.3.2.2 with full explanation.

Name	Length (bits)	Description
Type	8-bits	The field type is used to show the management message type, the table shows 256 different types of this field
Code	8-bits	The code field is used to show the privacy key management type, and when message is received with unacceptable it will be discarded
X.509	--	The x.509 is the SS certificate which is issued by the manufacturer. This certificate is the SS public key which combines the public key of RSA to the recognising information of SS in a certifiable approach.

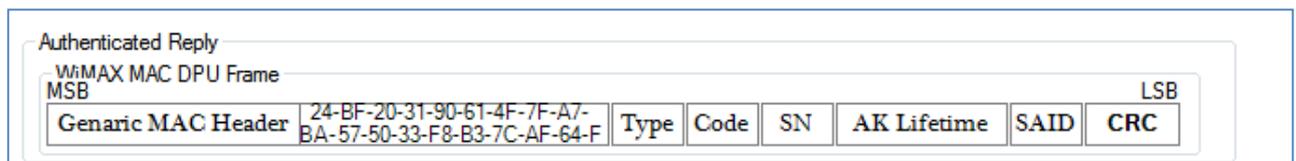
**Table 5.3.2.2: MAC Management Message Payload**

It can be seen from the MAC management message figure that, when SS sends an authenticated request, the MAC management message payload (MMMP) contains five fields

instead of three. The first of the two new added fields is EA, which stands for encryption algorithm. This field is used to inform the BS which algorithm technique the SS is using, in order to encrypt SS's payload using the same encryption algorithm.

The second new field is CID, which is used to show the connection identifier that is distinct for SS transport data.

Figure 5.3.2.1 shows the authenticated reply from the BS in the same simulator window as that of the MAC management message.



**Figure 5.3.2.1:** Authenticated Reply from BS

As shown in figure 5.3.2.1 above, the most significant bit, (MSB), is placed at the left hand side of the WiMAX MAC PDU frame, and the least significant bit, (LSB), is placed at the right hand side of the WiMAX MAC PDU frame. In the MAC PDU frame there are eight different fields, some of which were defined in the table of MAC management message payload. It can be seen that the MMMP is the only part which is encrypted, whereas the rest are not. Table 5.3.2.3 shows the definitions of SN, AK Lifetime, SAID and CRC.

Name	Description
SN	This is the sequence number field used to show the number of MAC PDU to the receiver. It also helps the receiver, which is, in this case, the SS, to know which messages are spoofed by a third party and thereby helps in preventing reply attack from happening to the WiMAX system
AK-Lifetime	Authentication key is used to provide freshness in the messages transmitted between the SS and BS; AK-lifetime is used to inform the receiver (SS) about how long the key is valid. The information provided is saved in the dedicated memory
SAID	This field is the security association identifier and is 16-bits in length and utilised by the privacy protocol to recognise the security association (SA) of the system
CRC	This is the cyclic redundancy check field, which is used to detect errors in the MAC PDU

**Table 5.3.2.3:** MAC PDU Frame Fields

The important MAC management messages are transmitted during the initialisation process, and some of these are continuously transferred, such as UCD, UL-MAP and DCD. These messages are transmitted through one of the following management connections, basic management connection, primary management connection and secondary management connection. These management connections are also utilised to transmit short time critical MAC management. The three 16-bits CID are used to recognise the three management connections.

## 5.4 NetSecSim Simulator Security

The main aim of NetSecSim is to provide a security simulation environment to wireless networks. To that end this section presents the two major achievements.

The parts of signal initialisation and the management messages that have opened the opportunity to attackers to attack wireless network, e.g. WiMAX, have been presented here. Hence, new methods to tackle security vulnerabilities such as Enhanced encryption & decryption algorithms and the Denial of Service attacks are revealed in section 5.4.1 and 5.4.2 consecutively.

### 5.4.1 Encryption and Decryption Algorithms

This section details the means of simulating security within the NetSecSim simulator, based on the security sublayer of the MAC layer, as mentioned in 2.2.2.3 of Chapter 2. The security sublayer is dedicated to providing confidentiality, privacy and authentication to the WiMAX system. Encryption and authentication are the two main principles used to obtain security in the NetSecSim simulator design.

Authentication is utilised to only allow access to the network for an authorised entity and prevent unauthorised entities from having access to the network. There are two main ways of achieving authentication, open system authentication (OSA) and shared key authentication (SKA).

The OSA is approached by SS sending an authentication request, linking its MAC address to the BS. The BS then responds to the request by either accepting or denying.

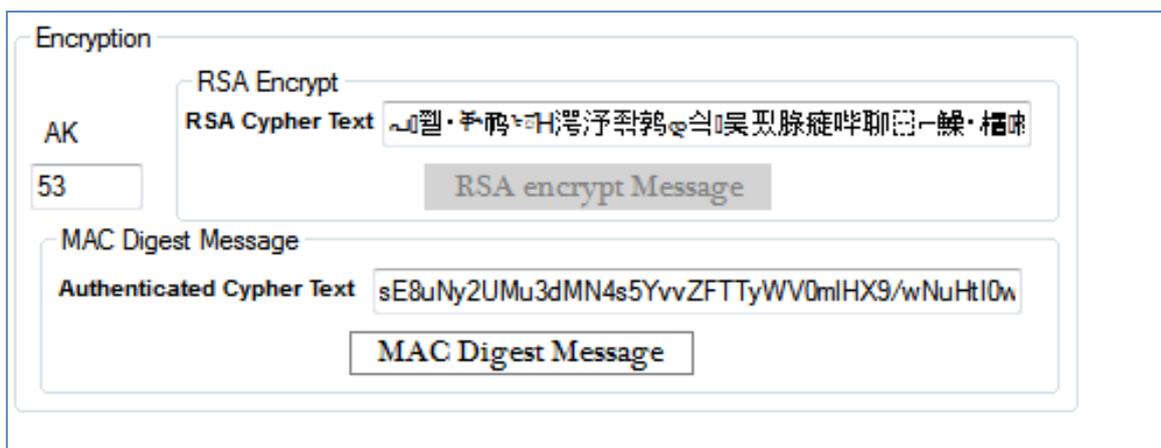
The other approach is SKA in which shared keys are used to obtain authentication. Both parties must know the keys for encryption and decryption. To successfully exchange keys between both parties, SS sends a PKM-REQ message to BS containing the X.509 digital certificate of SS. Then BS responds to the SS request after checking the certificate and encrypts the key using the same algorithm encryption as SS. The SKA process is referred to in the SS Authorisation and Key Exchange section 5.2.6.

The second security principle used in WiMAX is Encryption. When SS obtains authentication from BS, BS encrypts the message which contains the data, to add confidentiality and integrity to the message. SS then sends a message to BS requesting the encryption keys, which are known as traffic encryption keys (TEKs). The message which includes the TEKs is encrypted and the encryption keys are known by both parties. While WiMAX uses three different algorithms, this project provides four different encryption algorithms which can be used to perform the process, RSA, ZRSA, 3DES and AES.

As stated earlier, for this simulator only one type of attack is considered, namely BFA attack, and to prevent it from being successful, RSA and ZRSA algorithms were used to encrypt and decrypt the exchanged messages. It is recommended that RSA and ZRSA uses bit length of  $2^{512}$  but NetSecSim is limited to Visual Studio bit size of  $2^{64}$ . In section 3.2.5.6, it was proved that ZRSA is more formidable against factorisation as it would require three times the duration that is required to factorise RSA.

As hackers are able to factorise RSA with bit length  $2^{512}$ , (Skrevet, 2013 ) is recommending RSA to use bit length  $2^{2048}$ . However, due to the fact that, ZRSA requires three times the duration of factorising RSA, the researcher/author of this project strongly believes that with the current available computing power, it would suffice to use bit length  $2^{512}$  with ZRSA.

Authentication was achieved by the use of 3DES algorithm with MAC Digest (MD5). Figure 5.4.1.1 shows ZRSA and MAC Digest ciphertexts after the encryption process in the NetSecSim simulator.

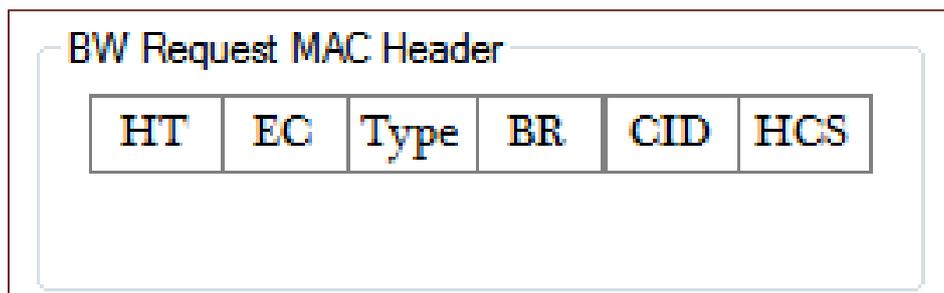


**Figure 5.4.1.1:** ZRSA and MAC Digest Ciphertexts

### 5.4.2 NetSecSim Simulator with DoS Attack

Up to this point, despite all the steps taken to make NetSecSim simulator MAC layer more secure, Denial of Services (DoS) attack can still take place. This is due to the difficulty in making all the messages transmitted between the SS and BS authenticated using just one authentication key.

As explained previously in the thesis, the WiMAX PHY layer is based on OFDMA and supports MIMO. Since it is based on OFDMA technology, it has very high data rates. In the case of a user having large data to be transmitted, an extra bandwidth can be requested by BW-REQ message. Figure 5.4.2.1 shows the 6 fields of the BW Request MAC header.



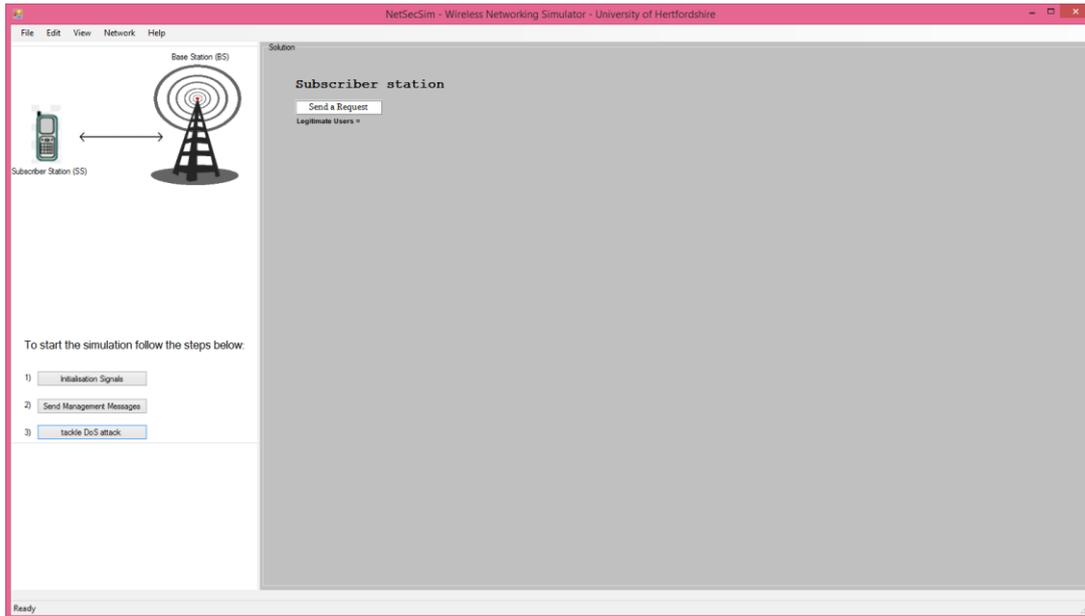
**Figure 5.4.2.1:** BW Request MAC Header

From the above BW request MAC header figure, five of these fields were fully explained in section 5.3.2 of this chapter, chapter 5. However, there is one new field, BR which stands for bandwidth request. It is 19-bits in length and is used to show the number of bytes needed by SS for the Uplink BW. The BW-REQ does not include physical overhead, as it contains the primary management connection CID value. So, the WiMAX 802.16 technology has the advantage of the user being able to request bandwidth with no limitations.

However, it is also a disadvantage as an attacker can use it to make the server busy for other users and overload the BS by using all of the network's free bandwidth. The attacker can harm the network by joining as a legitimate user. The effects of the attack to the system are as follow:

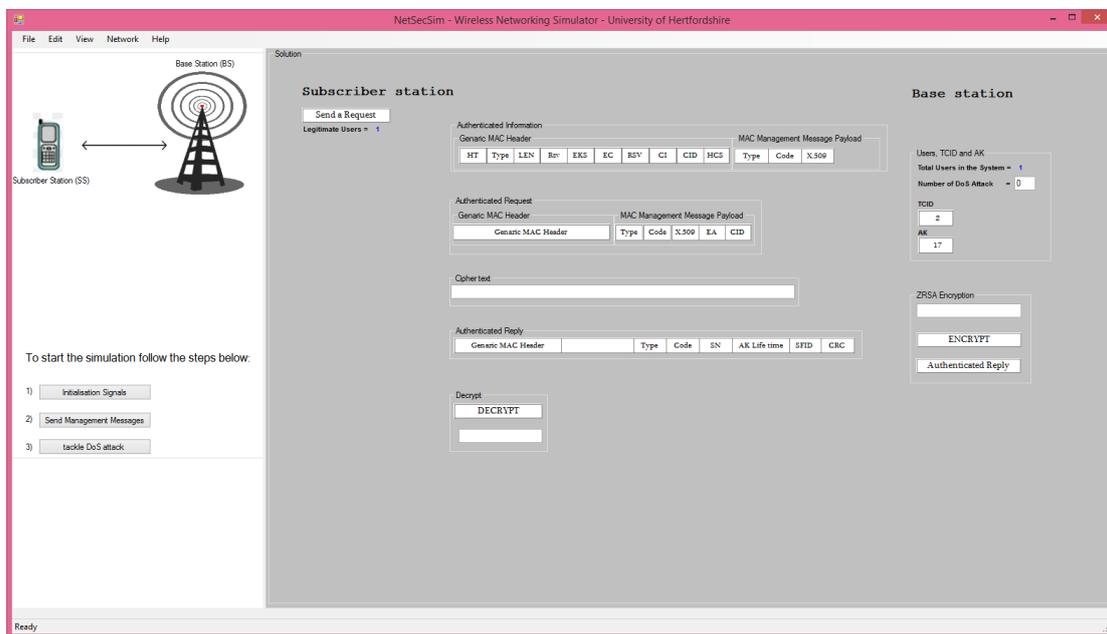
- Not allowing current users of the system to gain a good service quality
- Denying new users connection with the base station
- A busy network from the base station may appear to current users of the network as new users try to communicate with the base station

NetSecSim facilitates DoS attack simulation as shown in figure 5.4.2.2.



**Figure 5.4.2.2:** Selecting DoS attack Option with NetSecSim

Figure 5.4.2.3 shows a screenshot of the solution developed here to overcome the DoS attack in the NetSecSim simulator.



**Figure 5.4.2.3:** Solution to DoS Attack

It was mentioned before that some of the transmitted MAC management messages between the SS and BS are not encrypted and unauthenticated. A message being authenticated by the sender does not mean it is encrypted. It is hard to use an authentication key to encrypt the payload in messages transmitted using broadcast channels.

As a result, an attacker can attack the network using DoS attack because of the MAC management messages encryption and authentication loophole. The attacker also has the ability of spoofing MAC management messages and obtaining both sender and receiver details. In order to solve this problem, another way of encrypting MAC management messages in the simulator was introduced.

The MAC management messages are encrypted using the RSA and ZRSA encryption algorithm, which is the same way of encryption used by the WiMAX standard. Then, by using the AK and DES encryption algorithm, the digital signature is added to the encrypted payload. The final stage is to transmit the signed and encrypted payload using the MD5 algorithm. By performing the simulation through the steps mentioned, three security services can be obtained, confidentiality, authentication and integrity.

Confidentiality is obtained by encrypting the messages transmitted between both parties using the encryption algorithms mentioned above. The authentication security service is achieved by adding the digital signature to the encrypted messages to prove the identity of the sender to receiver. The integrity security service is obtained by using the MD5 technique, which is adding a hash function to the message.

The main aim of the simulator is to verify that the messages received by the receiver are authenticated. In the case of message authentication, the user is able to decrypt the ciphertext received. However, in the case of authentication failure, the received ciphertext cannot be decrypted, but is discarded by the system. In this simulator, spoofing messages is hard, for the following reasons:

- The management message received by the receiver does not contain all or any of the plaintext
- BS signed the message before sending it to SS using the AK

A solution was introduced and developed in the NetSecSim simulator for the overloading process caused by DoS attackers. This attack is carried out by the attacker requiring more bandwidth using the BW-REQ MAC management message. As was previously stated, the WiMAX standards offer bandwidth as users request by sending a BW-REQ message to BS; however, as a solution in this simulator, each user can obtain a limited bandwidth. By applying a limited bandwidth request for users of the system, it includes the legitimate users as well. This solution prevents overloading the BS by a user, but will not provide service from relevant BS for other users.

Using the access control security service, a further solution was introduced to overcome the problem mentioned above. By using the access control, the system can identify the real legitimate users. To ensure that the users of the system are legitimate users, a database contains all the X.509 certificates of users who are already preregistered in the system. For a new user to gain access to the network, the X.509 certificate of the new SS is compared with the certificates which are kept in the system database. If the certificate of the new user matches one of the certificates of the relevant service provider, then the user continues the communication with BS. However, if the certificate of the new user does not match any certificate on the database, then BS denies service for the user and communication no longer takes place. This is explained in section 4.2.3.

## 5.5 Summary

In this chapter, a simulator that is called NetSecSim is designed and developed using the C# Visual Studio language. It is meant to address all security issues with 4G and 5G wireless network protocols such as WiMAX and LTE. As WiMAX is supported with a standard published by IEEE under the 802.16e protocol, it has been decided that the focus of NetSecSim to be on WiMAX. However, the code is open for future additions to include LTE and other protocols.

The simulation environment has been addressed via three main sections; i) Initialisation Process, ii) MAC Management Messages in WiMAX and iii) NetSecSim Simulator with DoS Attack. The three major areas were discussed and supported with explanation and simulation results within this chapter.

The simulator, NetSecSim, provides RSA and ZRSA (developed by the author), as well as modified versions of them. So, the user would have four options to select for encryption and decryption of the payload.

In addition a solution was introduced and developed in the NetSecSim simulator for the overloading process caused by DoS attackers. This attack is carried out by the attacker requiring more bandwidth using the BW-REQ MAC management message. As was previously stated, the WiMAX standards offer bandwidth as users request by sending a BW-REQ message to BS; however, as a solution in this simulator, each user can obtain a limited bandwidth. By applying a limited bandwidth request for users of the system, it includes the legitimate users as well. This solution prevents overloading the BS by a user, but will not provide service from relevant BS for other users.

## Chapter 6: Conclusion and Future Work

### 6.1 Conclusion

A chart of the protocols of wireless classification has been presented in Chapter 2. It is concluded that features of the security setup for these protocols do have common ground and share the same problems faced by users with hackers.

The project has investigated the WiMAX architecture and carried out a thorough survey of its security management. There is enough evidence, despite the security enhancement, that security vulnerability is still an issue with WiMAX.

This project suggested a Universal Solution to WiMAX security threats, such as DoS, securing management messages and masquerading, together with well-known standard security requirements.

The suggested solution has been condensed in an algorithm called Z Algorithm. Two approaches or phases have been laid down to suggest how to implement the solution with the next stage of this research program.

An enhanced version of the Message Authentication Code (MAC) algorithm has been proposed here to provide message integrity in the WiMAX 802.16 standard. It incorporates the MAC function in the authentication protocol and during the sending of data. The latter part is in the form of HMAC.

Mathematical analysis was applied on the WiMAX encryption/decryption algorithm, RSA. Weaknesses have been identified in RSA and, as a result, a new encryption/decryption algorithm, called ZRSA, has been suggested. Brute Force attacks on both RSA and ZRSA have been introduced. A comparison conclusion showed that ZRSA is superior to RSA in terms of the required factorisation time.

A platform to simulate security algorithms does not exist in the open market. A novel simulator that is called NetSecSim was designed and developed using the C# Visual Studio language. Its purpose was to address all security issues with 4G and 5G wireless network protocols such as WiMAX and LTE. As WiMAX is supported with a standard published by IEEE under the 802.16e protocol, it was decided that the focus of NetSecSim should be on WiMAX. However, the code is open for future additions to include LTE and other protocols.

The simulation environment was addressed through three main sections; i) Initialisation Process, ii) MAC Management Messages in WiMAX and iii) NetSecSim Simulator with DoS

Attack. The three major areas were discussed and supported with explanation and simulation results within this chapter.

The simulator, NetSecSim, provides simulation for RSA and ZRSA (developed by the author), as well as modified versions of them. So, the user would have four options to select for encryption and decryption of the payload.

In addition, a solution was introduced and developed in the NetSecSim simulator for the overloading process caused by DoS attackers. This attack is carried out by the attacker requiring more bandwidth using the BW-REQ MAC management messages, than is available. As was previously stated, the WiMAX standards offer bandwidth as users request by sending a BW-REQ message to the BS; however, as a solution in this simulator, each user can obtain a limited bandwidth. By applying a limited bandwidth request for users of the system, it includes the legitimate users as well. This solution prevents overloading the BS by a user, but will not provide service from the relevant BS for other users.

## 6.2 Future Work

The author is recommending further research project to focus on NetSecSim simulator to cover LTE. As mentioned in chapter 3, the trend is to merge WiMAX and LTE; this simulator, NetSecSim would be a truly effective tool to researchers in cryptanalysis.

Since QC is promising, in future, to deliver unparalleled computing performance, ZRSA algorithm could be simulated on one of the existing architectures of QC and further scrutiny could be placed on ZRSA which may lead to a new protocol such as 5G to be self-impregnable.

## References

- Al-Hamami, A. H. & A. I. A., 2012. Enhanced Method for RSA Cryptosystem Algorithm. *2012 International Conference on Advanced Computer Science Applications and Technologies*, pp. 402-408.
- Altaf, A., Iqbal, F. & Javed, Y., 2008. *A Secure, Seamless and Soft Handover between WiMAX and 3G Networks*. s.l., International conference on Convergence and Hybrid Information Technology.
- Alzaabi, M., 2013. *Efficient Authentication Security Algorithm for WiMAX*, Hatifled: University of Hertfordshire.
- Alzaabi, M., Ranjeeth, D., Alukaidey, T. & Salman, K., 2013. Survey on Security Algorithms for WiMAX. *International Journal of Computer Networks & Communications (IJCNC)*.
- Anon., n.d. *RFC 2104*, s.l.: s.n.
- Arjen K. Lenstra, E. R. V., 2001. Selecting Cryptographic Key Sizes. *Cryptology*, Volume 14, pp. 225-293.
- Barbeau, M., 2005. *WiMAX Threat Analysis*. Montreal, Quebec, Canada, Proceedings of the ACM.
- Barker, E. & R. A., January, 2011. *Transitions: Recommendation for Transitioning the Use of Cryptographic Algorithms and Key Lengths*, USA: National Institute of Standards and Technology - NIST.
- Boavida, F., 2007. *5th International Conference on Wired/Wireless Internet Communications, WIC 2007*. Portugal, Springer.
- Bogdanoski, M., 2008. *IEEE802.16 Security Issues A Survey*. s.l., TELFOR.
- Chang, J., Abichar, Z. & Hsu, C.-Y., 2010. WiMAX or LTE: Who will Lead the Broadband Mobile Internet?. *IT Professional*, Issue 3, pp. 26-32.
- Chen, K.-C. & De Marca, J. R. B., 2008. *Mobile WiMAX*. West Sussex: Wiley.
- Clawson, C. C., 1996. *Mathematical Mysteries*. Oxford: Plenum Press.
- Curtis, S., 2014. How quantum computers will undermine cryptography. *The Telegraph*, Wednesday November.

Delfs, H. & K. H., 2007. *Introduction to cryptography: principles and applications*. Canada: Springer. .

Diffie, W. & Hellman, M., 1976. New Directions in Cryptography. *IEEE TRANSACTIONS ON INFORMATION THEORY*, VOL. IT-22(6), pp. 644-654.

Dudziak, M. J., 2014. *Quantum Computing BioMedicine Program (QCMP)*. [Online] Available at: <https://www.linkedin.com/pulse/qubits-super-distributed-grids-implementing-refining-quantum-dudziak?trkSplashRedir=true&forceNoSplash=true> [Accessed 3 Nov 2015].

Elleithy, A., Abuzagheh, A. & Abuzneid, A., 2008. *A NEW MECHANISM TO SOLVE IEEE 802.16 AUTHENTICATION VULNERABILITIE*. Bridgeport, CT, Computer Science and Engineering Department, University of Bridgeport.

Exchange, S., 2012. *Cryptography*. [Online] Available at: <http://crypto.stackexchange.com/questions/1978/how-big-an-rsa-key-is-considered-secure-today> [Accessed 2 September 2015].

Ferguson, N. & Schneier, B., 2003. *Introduction to cryptography: principles and applications*. NY: Wiley.

Galbraith, J. & Saarenmaa, O., 2005. *X.509 authentication in SSH2*, NM 87111, USA: IETF .

Gardner, M., 1989. *Penrose Tiles to Trapdoor Ciphers*. s.l.:W.H. Freeman & Co..

Giry, D., 2015. Cryptographic Key Length Recommendation. *BlueKryt*, Volume 29.2.

Green Packet, 2015. *DUO*. [Online] Available at: <http://www.greenpacket.com/solution/duo/> [Accessed 2nd September 2015].

Grieu, F., 2000. *A Chosen Messages Attack on the ISO/IEC 9796-1 Signature Scheme*. Berlin, Germany, Theory and Application of Cryptographic Techniques - EUROCRYPT , pp. 70-80.

Grieu, F., 21/01/2015. *Cryptography Beta*, Paris, France: Stack Exchange: <http://crypto.stackexchange.com/questions/1978/how-big-an-rsa-key-is-considered-secure-today>.

Hong J., A. M. G. B., 2011. Simulating Denial of Service Attack Using WiMAX Experimental Setup. *International Journal of Network and Mobile Technologies*, 2(1), pp. 30-34.

- IEEE, 2005. *Patent*, New York: Patent No. US 8005179 B2.
- Kent, S. & Seo, K., 2005. Security Architecture for the Internet Protocol. *IETF RFC 4301*, December.
- Kumar, A., Liu, Y., Sengupta, J. & Divya, 2010. Evolution of Mobile Wireless Communication Networks: 1G to 4G. *International Journal of Electronics & Communication Technology*, 1(1), pp. 68-72.
- Kumarjit Banerjee, K., Nath, S. & Kumar, S., 2013. *Improved Trial Division Technique for Primality Checking in RSA Algorithm*. India, I. J. Computer Network and Information Security.
- Leyland, P., 1995. *Hostile Attack on BlackNet's PGP Key*. [Online] Available at: <http://sattlers.org/mickey/tech/privacy/topics/pgp/misc/blacknet-key-attack.html> [Accessed 3 Nov 2015].
- Li, Y., Z, Y. & Nui, W., 2010. *A Method of Privacy Preserving in Mobile Wireless Environments*. s.l., 7th International Conference on In Ubiquitous Intelligence & Computing and 7th International Conference on Autonomic & Trusted Computing (UIC/ATC), IEEE.
- Li, Z. & T. A., 1998. *RSA Public Key Cryptosystem*, s.l.: s.n.
- M. Bellare, M., Canetti, R. & Krawczyk, H., 1996. *Keying Hash Functions for Message Authentication*. s.l., CRYPTO '96 - Springer-Verlag.
- Michael, 2009. *TI-83 Plus OS Signing Key Cracked*. [Online] Available at: <http://www.ticalc.org/archives/news/articles/14/145/145154.html> [Accessed 2 Nov 2015].
- Molisch, A. F., 2010. *Wireless Communications*. 2nd ed. s.l.:Wiley.
- Nguyen, T., 2009. *A survey of WiMAX security threats*, s.l.: Washington University.
- NIIST, 2003. *NIST IPsec and IKE Simulation Tool*. [Online] Available at: <http://www.antd.nist.gov/niist/> [Accessed January 2014].
- Nuaymi, L., 2007. *WiMAX Technology for Broadband Wireless Access*. West Sussex: Wiley.
- Patidar, R. a. B. R., 2013. Modified RSA Cryptosystem Based on Offline Storage and Prime Number. *2013 IEEE International Conference on Computational Intelligence and Computing Research*.

Pau, V., 2005. Development of Secure IPsec Tunnelling in a Mobile IP Architecture. 10 November.

Pelé, (., 1999. *French banking smartcard cracked : the story!*. [Online] Available at: <http://web.archive.org/web/20051016203246/http://www.parodie.com/english/smartcard.htm> [Accessed 3 Nov 2015].

Pugila, D., Chitralla, H. & Lunawat, S. & V. P. R. V., 2013. An Efficient Encryption Algorithm Based on Public Key Cryptography. *International Journal of Engineering and Technology*, 5(3), pp. 3064-3067.

Sanjay P. Ahuja, N. C., 2010. An Assessment of WiMax Security. *Communications and Network*, Volume 2, pp. 134-137.

Shikha, V. K., 2013. A Review of WiMax / 802.16 Security Threats. *International Journal of Advanced Research in Computer and Communication Engineering*, 2(11), pp. 4443-4448.

Shukla, S., Khare, V., Garg, S. & Sharma, P., 2013. Comparative Study of 1G, 2G, 3G and 4G. *Journal of Engineering, Computers & Applied Sciences*, 2(4), pp. 55-63.

Skrevet, 2013 . *Choosing a GnuPG RSA key size*. [Online] Available at: <http://oletange.blogspot.dk/2013/09/choosing-gnupg-rsa-key-size.html> [Accessed 3rd March 2015].

Tshering, F., Deshpande, P., Sharma, S. & Sardana, A., 2013. *Proxy Base Station based Authentication Protocol for Broadband Wireless Network*. s.l., International Journal of Computer Applications.

Ulvan, A., Andrlik, V. & Bestak, R., 2009. *The Overhead and Efficiency Analysis on WiMAX's MAC Management Message*. Indonesia, Internetworking Indonesia, 3.

William, S., 2011. *Cryptography and Network Security Principles and Practices*, s.l.: Pearson.

WiMAX, Q., 2015. *What is Wibro?*, New York: <http://quantumwimax.com/index.php?page=What-is-Wibro>, Cited on 05/09/2015.

## Appendices

### Appendix A

Figure 3.2.5.6.1 samples:

Year	Bits
1992	364
1993	397
1994	426
1996.5	430
1999	463
1999.5	512
2003.5	530
2004	576
2005.5	663
2010	768
2015	992
2020	1152
2025	1312
2030	1472
2035	1632
2040	1792
2045	1952
2050	2112
<b>BF Attacks on RSA</b>	
Year	Factorised Bits
1999.5	384
1998	325
2009.5	512

## Appendix B

Figure 3.2.5.6.2 samples:

Bits		Duration range for both RSA & ZRSA
ZRSA	RSA	
10		3.08779E-20
	10	1.16486E-13
100		38224996.84
	100	1.44274E+14
512		4.043E+131
	512	1.526E+138
1024		5.4215E+286
	1024	1.023E+292
2048		9.74E+593
	2048	3.68E+600
4096		3.15E+1210
	4096	1.19E+1217

## Appendix C

Year	Bits (RSA)	Bits (ZRSA)
1992	364	512103
1993	397	512360
1994	426	512618
1996.5	430	513260
1999	463	513903
1999.5	512	514031
2003.5	530	515060
2004	576	515188
2005.5	663	515574
2010	768	516731
2015	992	518016
2020	1152	519302
2025	1312	520587
2030	1472	521872
2035	1632	523158
2040	1792	524443
2045	1952	525729
2050	2112	527014
Year	RSA BFA	
1999.5	384	
1998	325	
2009.5	512	