DESIGN OF USER AUTHENTICATION MODEL FOR BLUETOOTH PICONET

By

ZABLON O. BIRUNDU

REG NO: KCA/11/00888

A DISSERTATION SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF DEGREE OF MASTER OF SCIENCE IN DATA COMMUNICATION IN THE SCHOOL OF COMPUTING AND INFORMATION MANAGEMENT AT KCA UNIVERSITY

NOVEMBER, 2012

ABSTRACT

The main objective of this research was to design a user authentication model for the Bluetooth piconets. Bluetooth is a new and young technology of data communication, since its birth it has been facing a lot of challenges. Security is one of the challenges and this research was conducted in an effort to enhance its security by creation of user authentication model to curb unauthorized usage. Apparently Bluetooth is open to any user of the device be it authorized or unauthorized. The login model designed in this thesis will allow only authorized users to use the Bluetooth piconet. The model was designed and created using J2ME program and was tested and found to be working well. The model was able to protect the Bluetooth piconet from unauthorized users.

DECLARATION

I hereby declare that this dissertation is my original work and has not been previously published or submitted elsewhere for award of a degree. I also declare that this contains no material written and published by other people except where due reference is made and author duly acknowledged.

Student Name: _____ **Reg. No.** _____

Sign: _____ **Date:** _____

I do herby confirm that I have examined the master's dissertation of _____

_____

And have certified that all revisions that the dissertation panel and examiners recommended have been adequately addressed.

Sign:_____ **Date:** _____

Supervisor's Name: Dr. Patrick Kanyi.

DEDICATION

First, I would like to thank God Almighty for everything I am today. Secondly, I would like to dedicate this thesis to the following.

My parents for their endless love, support and encouragement.

My supervisor Dr. Kanyi and his colleagues for helping me to undertake this dissertation from start to completion.

My lovely wife Emily and son Allerius who supported me and always kept a calm attitude during hard situations.

My Brothers, sisters classmates and friends for their support and encouragement all through.

May God Bless you all.

Table of Contents

List of Figures

ACCRONYMS AND ABBREVIATIONS

BD_ADDR-Bluetooth device address

CLDC-Connected Limited Device Configuration

DoS-Denial-of-Service

FHSS-Frequency Hopping Spread Spectrum

GAP-Generic Access Profile

GHZ-Gigahertz

ISM-Industrial Scientific Medical

J2ME-Java 2 Micro Edition

JABWT-Java APIs for Bluetooth Wireless Technology

L2CAP-Logical Link Control and Adaption Protocol

LMP- Link Manager Protocol

MAC-Media Control Address

MHZ-Megahertz

MID-MIDlets Mobile Information Device

MIDP-Mobile Information Device Profile

PAN-Personal Area Network

PC- Personal Computer

PDA-Personal Digital Assistants

RFCOMM- Logical Link Control and Adaptation Protocol

SDDB-Service Discovery DataBase

SDP-Service Discovery Protocol

SIG-Special Interest Group

SPP-Serial Port Profile

WAN-Wireless Local Area Network

WAP-Wireless Application Protocol

WTKs-Wireless ToolKits

CHAPTER ONE: INTRODUCTION

1.1 Background of the Problem

Piconet is a network formed when two Bluetooth enabled devices connect to each other. It is an ad hoc network formed without any underlying infrastructure, implementing security in such a network is complex since devices enter and leave the network as they wish. Many applications are being developed to use Bluetooth piconets, and currently there is no user authentication in Bluetooth piconets, if anybody gets hold on a Bluetooth device then it becomes easily usable by the holder and thus any data communication can be done easily with or without authorization. Now Bluetooth devices have become more sophisticated and powerful in terms of the data that they hold and their processing power, thus calling for the control of their access by incorporating user authentication. This thesis therefore proposes to design a user authentication model for Bluetooth piconet, in order to improve its security by ensuring that authentication will be required for both the user and the device. The user will be authenticated to the device first then the device can normally authenticate itself to the other devices that it wants to communicate with.

1.2 Problem Statement

Recently Bluetooth has become popular with many portable personal devices such as laptops, PDAs, earphones, mobile phones, watches and many other devices. Portability means great convenience and the power of the device affords much productivity and use, however on the other hand it also increases the potential for loss and theft. Since these portable devices can perform more functions and store more data, they become great instruments for attackers to target whether they are with the owner or attackers have acquired them by some means from the owner. Bluetooth operates on the ISM license free band; it uses low power hence making it suitable for the power limited devices. The Bluetooth protocol was intended to replace wires connecting the portable devices to allow for wireless interconnection, however with its popularity and capability it has been adopted in many handheld devices for communication

purposes. Bluetooth technology enables two Bluetooth enabled devices to connect and communicate, sending voice and data to each other. When two Bluetooth enabled devices connect to each other they form a wireless personal area network also known as a piconet. Bluetooth piconet just like any other wireless network is prone to security risks and according to (Colleen, 2006) Bluetooth piconet employs security procedures such as authorization, authentication and optional encryption. Authentication is proving the identification of a computer or its user, or in Bluetooth's case, involves identification of one piconet member to another member. Authorization involves granting or denying access to network resources to both the user and the device. Encryption is the process of translating data into secret code used between Bluetooth enabled devices during communication so that eavesdroppers cannot read its contents. Despite all of these defense mechanisms being in place, Bluetooth has shown to have some security risks. The same is also indicated by (Lewis, 2005) where he indicated that Bluetooth devices communicating in the wild, form various sizes of ad hoc networks where varying number of devices can be entering and leaving the piconets and scatternets created by the devices at any time hence making it difficult for Bluetooth piconet to maintain good security. Therefore the ad hoc nature of the piconet makes it difficult for security to be fully implemented as devices join and leave the piconets at their own pleasure. Joining and leaving willingly makes the network vulnerable since some devices can impersonate others that have already left, in a piconet once a device has been paired, it is put in the paired devices list showing that connection has been established, this is a security risk because if the owner happens to lose the device, not only will the data contained in the lost device at risk if they fell into the wrong hands, but also any other devices that these Bluetooth enabled devices had paired with are also vulnerable. Bluetooth devices that previously had created a trusted relationship (i.e. had paired with another Bluetooth device) will store the trust relationship by keeping the respective keys in non-volatile memory unless set in way to delete the keys after some determined period of time. This becomes a great risk to the other Bluetooth enabled devices owned by the victim who lost the Bluetooth device or others who had paired with the lost or stolen device. Usually all the keys, which includes link keys and unit keys, of the other Bluetooth enabled devices that had been paired with the lost or stolen device are stored on the database of that device and it is possible for the attacker or culprit to access the link keys from the victims device and be able to:

i. Eavesdrop on communications between the victims device and those it had paired with previously,
ii. Establish a communication with the unsuspecting devices that it had paired with previously and access unauthorized data.
iii. To achieve more sophisticated attacks by using the unit keys obtained from the lost/stolen device to program another more powerful Bluetooth enabled device in order to impersonate or spoof that device.
iv. Obtain or determine personal identification numbers of the other device or devices that it had previously paired with, or
v. Be able to know some relationships that had been established previously between the victim's device and other devices.

Considering the above risks that the lost/stolen device might pose to the victims other devices, a need arises to implement user authentication in the application layer. By implementing user authentication into the Bluetooth device, will ensure that anybody in possession of the Bluetooth device will be required to be authenticated in order to use the piconet. Session timeouts will also be included in order to guard against the problem of forgetting to terminate a session. In any wireless network security is a concern; this is because it is hard to detect if your connection is being tapped by outsiders and for Bluetooth piconets case, devices are always enquiring for any active devices anytime, anywhere in order to try to establish a connection with each other. This might seem ok but in reality it poses security risk and according to (Aroackiasamy, & Latha, 2010) a Bluetooth piconet can be at risk, if one or more devices in the network which is used as bridge to other connected networks is compromised, this could expose the devices or their attached networks. Attacks can be carried out easily and they may not be detected easily since the network is ad hoc, with no access point, and no centralized security administration as there is with a WLAN where MAC address filtering and other security mechanisms are employed to provide protection against rogue access. This proposed thesis will focus on identifying the security weaknesses in the Bluetooth piconet topology, propose the best mechanisms in which they can be mitigated and the design of a user authentication model to be used at the application level for user authentication.

1.3 Purpose and Scope

The purpose of this thesis is to provide information about the Bluetooth piconet security vulnerabilities, propose mechanism to mitigate them and the development of a user authentication model for the Bluetooth piconet users to be implemented at the application level.

1.4 Objectives

The main objective of this research is to design and implement a user authentication application for Bluetooth piconets that can be used to address the issue of the device being a security risk when the device owner losses it.

i) To highlight on architecture of the Bluetooth so as to understand the core protocols and how they work.
ii) To identify security risks and vulnerabilities associated with Bluetooth piconets and their mitigation mechanisms.
iii) To design a user authentication model for Bluetooth piconets users.
iv) To implement user authentication at the application level.

1.5 Research Questions

The research seeks to answer the following questions.

i) What are Bluetooth piconet protocols?
ii) What are Bluetooth piconet security vulnerabilities and risks?
iii) What are the solutions to those weaknesses?
iv) How can Bluetooth piconets security be improved by incorporating user authentication?

1.6 Justification

Bluetooth wireless is constantly growing in popularity because of its convenience in exchanging of information between mobile devices. As Bluetooth usage rises, so do the security risks associated with the technology. A person can own more than two of these devices and it's most likely that they would want to interconnect them through the Bluetooth protocol for purposes of data communication, forming Bluetooth piconet. Most of these devices are personal and portable and they will most likely find their way into organizations where the owners are working, and considering the fact that most wireless technologies are prone to many attacks, their usage in the workplace exposes organizations to security risks. Most of these organizations do not have a security policy for these kinds of networks (Colleen, 2006) hence organizations can be exposed easily without knowing. Some of the Bluetooth wireless attacks include eavesdropping and impersonation, person in the middle attack, piconet/service mapping, denial of service attack and theft etc.

There has been a reluctant approach to piconet security, yet there is a great scope for experimenting with the design of security models for this kind of network and due to its mobility adequate security mechanisms are needed to make it secure. However, there has been an attitude that it is not beneficial to be burdened with these requirements at this early stage. (Bennett, Clarke, Evans, & Andy, 2010). This clearly shows that researchers are reluctant in researching in this area of security, yet it's a network of our modern days. Bluetooth piconets don't have user authentication mechanism, where any user of the network is authenticated first before accessing it; normally it is the device that is authenticated and not the user, so what if it is not the owner who possessing the device? If by any chance a device gets lost or stolen then the device will always authenticate itself to any device it had paired with previously, establishing a legitimate communication link, thus unauthorized third parties will normally be able to use it almost immediately (Ajay, 2008). This thesis is aimed at addressing this problem by designing a user authentication model for the Bluetooth piconets, which will enable the device to differentiate between a legitimate user and unauthorized users, to guard against device embezzlement and other unauthorized parties from gaining instant access to the piconet.

CHAPTER TWO: LITERATURE REVIEW

2.1 Introduction

This chapter focuses on presenting the literature review of the research, which focuses on presenting information about Bluetooth protocols, its security and knowledge from related work. These will serve as the foundation for the Bluetooth application to be created and will also be a source of information to the reader.

2.2 Bluetooth Background

Bluetooth is a short-range radio link capable of transmitting voice as well as data. Bluetooth was intended to eliminate cables that connect personal electronic devices, such as PDAs, laptops, wireless headsets and printers. The main advantages of Bluetooth are its robustness, low complexity, low cost, low power consumption and universality. Its communication distance ranges from 10m to 100m depending on its power. Bluetooth is a global standard (i.e. IEEE 802.15, PAN); one that allows people to get connected through their devices virtually anywhere and without any infrastructure (Gelzayd, 2002). Bluetooth protocol was first created in order to replace the cables interconnecting the portable devices but due to its ad hoc nature more research is being done in improving the protocol for its use in ad hoc networking. Security in ad hoc networking comes with more complexities; this is due to its dynamism and infrastructure less in nature.

Bluetooth gets its name from a Danish Viking king named Blataand, who was king of Denmark around the late 900s. Blataand was responsible for Christianizing Denmark and uniting it with part of Norway. His name only signifies the importance of countries in this region of the world in the wireless industry. In 1998, Bluetooth consortium named Bluetooth Special Interest Group (SIG) was formed and comprises of more than 1000 members, comprising of companies like Toshiba, Intel, Nokia and other many more companies (Bluetooth, 2006).

Bluetooth piconet radio wave operate in the unlicensed ISM band with 2.45 GHz frequency, and it uses a spread-spectrum frequency-hopping technique which takes a narrowband signal and spreads it over a broader portion of the available radio frequency band. In frequency

hopping spread spectrum (FHSS), the available frequency is divided into multiple channels of smaller bandwidth (Preetha, 2010). The hopping rate is 1600 hops per second. The operating frequency band is divided into 1MHz spaced channel, each supporting data rate of 1MHz. Frequency hopping technique is used to reduce the effect of interference from other users in the same band.

2.3 Bluetooth Protocols

Bluetooth is a low power protocol suitable for low power devices (cell phones, PDAs, Laptops, computers) to communicate with each other over a small range. Beside cable replacement, where Bluetooth replaces a number of cables, such as those that were traditionally used for peripheral devices (e.g., mouse and keyboard connections), printers, and wireless headsets and earphones that connect with personal computers (PC) or mobile telephones, it has made sharing of files easier where a Bluetooth-enabled device can form a wireless network known as a piconet to support file sharing capabilities with other Bluetooth-enabled devices, such as laptops. Bluetooth provides automatic wireless synchronization between Bluetooth-enabled devices. For example, Bluetooth allows synchronization of contact information stored in an electronic address books and calendars. A Bluetooth device that has Internet connectivity can share that connectivity with other Bluetooth-enabled devices. For example, a laptop can use a mobile phone to establish a dial-up connection, so that the laptop can use the phone as a modem to access the Internet, which is also known as tethering. Bluetooth protocols can be categorized into four categories (Dahlberg, 2002) i.e. core protocols, cable replacement protocol, (include RFCOMM), telephony control protocols (include TCS Binary and AT-commands) and adopted protocols (include PPP, UDP/TCP/IP, WAP, WAE). The core protocols including Baseband, LMP, L2CAP, and SDP, are required by most devices when the other protocols are more dependent on the applications that they support. The following diagram represents the Bluetooth protocol stack. (Patheja, Akhilesh, & Nagwanshi, 2012)

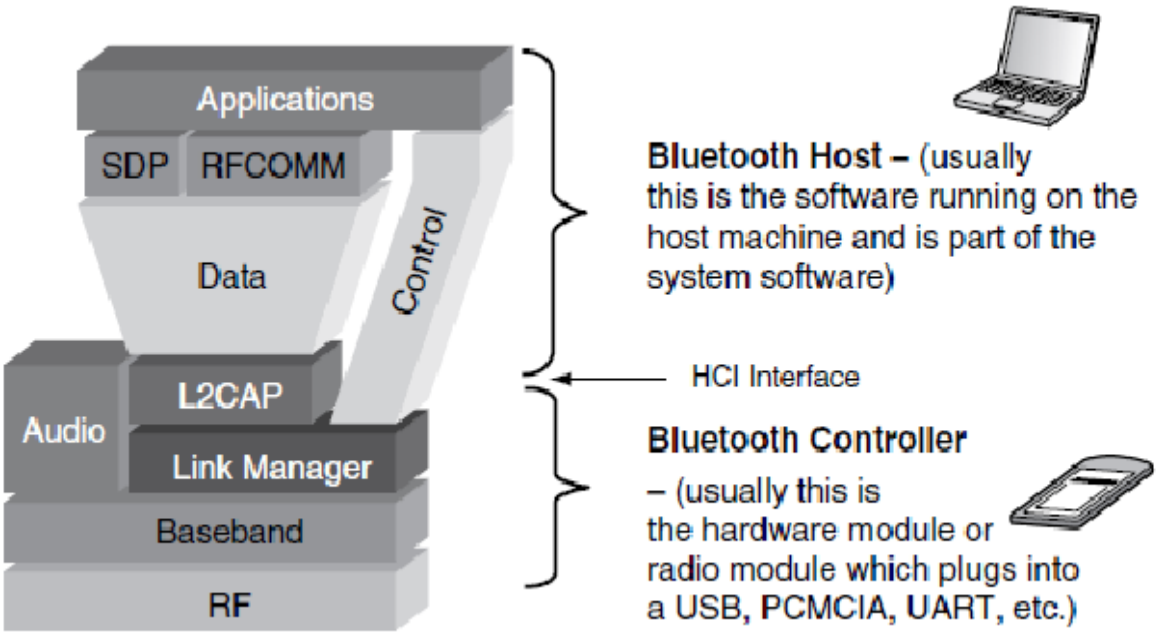

Figure 1 Bluetooth Protocol Stack (Thompson, Kline, & Bala, 2008)

2.3.0 Radio layer

Radio layer is the layer responsible for establishing the physical wireless connection. It uses fast frequency hopping modulation to avoid interference with other devices communicating in the same ISM band. There are 79 channels each with 1 MHz apart which is divided from the 2.4 GHz frequency band of Bluetooth (from 2.402 to 2.480 GHz). Bluetooth uses spread spectrum to hop from one another with up to 1600 times per second (Haataja, 2009).

2.3.1 Baseband layer

The work of the Base band layer is to control and send data packets over the radio link. It provides two channels for transmission, i.e. synchronous connection oriented (SCO) for voice and asynchronous connectionless oriented (ACO) for data. It can establish piconets between Bluetooth units and decide the roles of master and slave in the piconet. It also performs other functions such as correcting errors, transmission of audio, link management and control (Patheja et al.2012).

2.3.2 Link manager protocol (LMP)

Link Manager Protocol (LMP) is responsible for link set-up between Bluetooth-enabled devices. Its functions include security service, the control and negotiation of Baseband packet sizes. LMP controls the power modes and duty cycles of the Bluetooth radio device and manages the states of a Bluetooth unit in a piconet. It is also responsible for both the pairing process and for the handling of the challenge/response procedure for device authentication purposes (Niem, 2003).

2.3.3 Logical link control and adaptation protocol (L2CAP)

The main function of this protocol is the provision of connection-oriented and connectionless data services to the upper layers. It receives data and changes it to conform to the Bluetooth format. The Quality of Service parameter is also exchanged at this layer (Haataja, & Keijo, 2006).

2.3.4 Host controller interface (HCI)

This protocol acts as an interface between the hardware and software. It provides a uniform command interface to facilitate access to capabilities of hardware, e.g. Baseband controller, link manager, control and event registers. (Holmburg, Marcus, & Eric, 2004)

2.3.5 Service discovery protocol (SDP)

Using SDP to discover services is an important part of the Bluetooth framework and provides the basis for all the usage models. SDP queries device information, services information, and the characteristics of the services, that enable a suitable connection between two or more Bluetooth devices to be established (Haataja, 2009).

2.3.6 Application layer

Application layer is responsible for the management of the communications between host applications and provide guidelines to developers on how applications should use the protocol stack. This is the layer where the implementation of the user authentication model will be implemented.

Telephony Control, Cable Replacement and Adopted Protocols are application-oriented protocols that allows for applications to run on the Bluetooth core protocols. It is not ll applications that use all the protocols shown in Figure 1, but instead, some applications use one or more vertical slices of Bluetooth protocol stack, hence applications may use different protocol stacks but use common Bluetooth data link and physical layer. Additional protocols such as FTP and HTTP etc. can also be used in Bluetooth due to its openness in nature.

2.4 Piconet Topology

A Bluetooth network also known as piconet is formed by simply having a minimum of two or a maximum of eight Bluetooth devices establish a communication link between each other. Bluetooth has two types of networks namely piconet and scatternet, where a Piconet is a small network that can be formed by up to a maximum of eight active nodes, among them one a master and the rest active slaves it can also scale to include up to 255 inactive slaves. Communication between a master and a slave can be either one-to-one or one-to-many. A piconet has a maximum of seven slaves while Scatternet is formed by a combination of two or more piconets connected together. (Welke, 2006). A slave node in one piconet can be a master in another piconet. This slave node can receive messages from both the piconets and act as master/slave at the same time and it communicates by means of multi hopping to avoid interference from nodes and any other network which uses same frequency hopping spread spectrum method such as Wifi etc., it hops 1600 hop per second. The baseband layer works as the MAC layer in LANs and in order for it to access given layer it uses the TDMA. All the master and slave communicate using time slot (625 μs) (Ajay, & Komal, 2010). The master device works as the moderator to all the other devices in the piconet including determining the networks frequency hopping scheme. It manages all the communication between itself and all the slave devices as well as between the slave devices. Pairing must be established between the devices in a piconet where pairing is a trusted relationship formed by the exchange of secret codes among the piconet devices. The codes must be exchanged in secret manner for security purposes. Every device is given a Bluetooth address used not only for its identification, but also for synchronizing the frequency hopping between devices and generation link and encryption keys in the Bluetooth security procedures.

The diagrams below represent the piconet topology and the master and slave concept in a scatternet respectively.

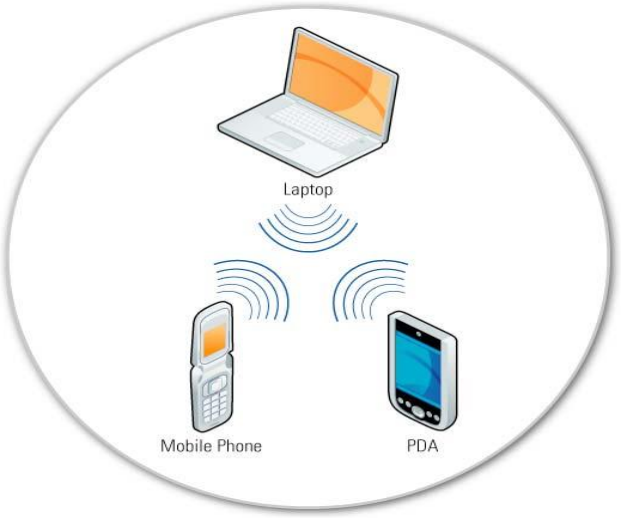

Figure 2: Bluetooth piconet topology (Karen, & John, 2008)

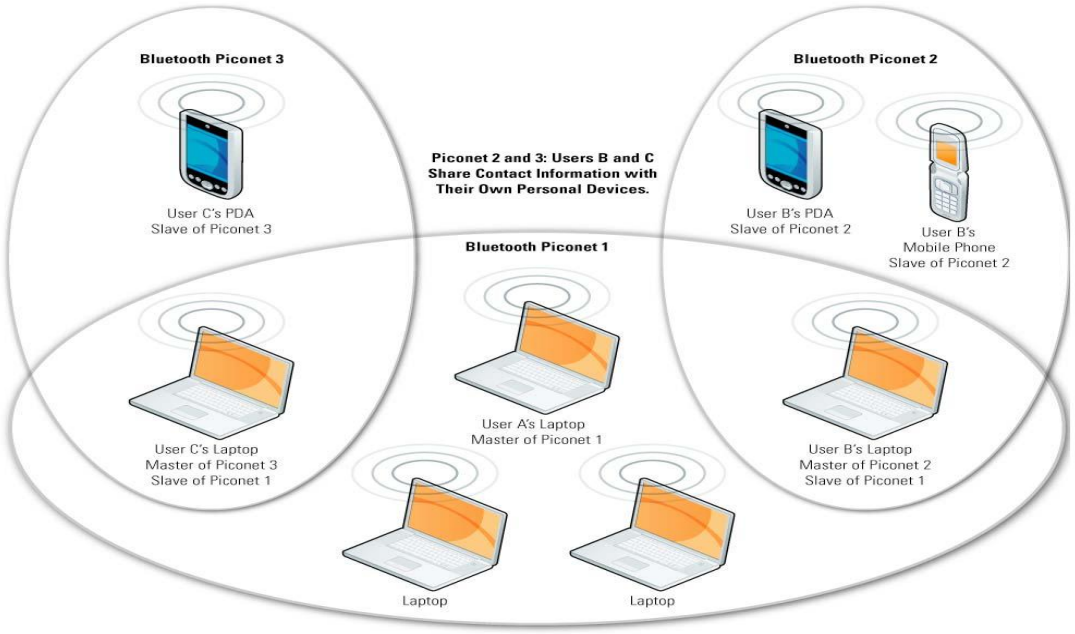

Figure 3: A Scatternet formed by piconet A, B and C represented by cycles (Karen, & John, 2008)

Scatternet is formed when two or more piconets join together, it allows for several Bluetooth devices to form a network over an extended distance in a dynamic topology that can self re-organize at any given session. As a device comes towards or moves away from the master device in a piconet, their relationships in the immediate piconet change at any given time.

2.4.0 Piconet establishment

Each Bluetooth device has to form a piconet in order for it to communicate with another Bluetooth device. In the establishment of a piconet it has to go through some steps of state changes and exchange information with other devices for connections. Initially, all devices will be in Standby state. Then when a device (potential master) wishes to connect with other devices (potential slaves), it will send its ID packet in an Inquiry state. Any device in Inquiry Scan state picks up the ID packet and replies with a FHS packet, which will contain information of the hopping sequence and then goes back to Standby (Persson & Manivannan, 2003).

On receiving the FHS packet from a potential slave, the potential master reverts to Standby state. During this time, the potential master's baseband notifies the upper layer for a connection establishment. Its upper layer then instructs the baseband to change to Page state and start paging the potential slave through sending its ID packets. The slave to be device, after a certain period of time, changes its state to Page Scan state, in order to scan the ID packets sent by a potential master. After a page timeout time ends, the potential master then transitions to Master Response state to listen for the response from the potential slave. When a potential slave, receives the ID packets in the Page Scan state, it responds by sending an ID packet to the potential master and then switch to the state of Slave Response in order to listen for the potential master's FHS packet. Finally, a connection between them is setup. During the course of connection, the master sends POLL packets to its slave and the slave replies it with a NULL packet. (Yelena, 2002)

In a piconet, the master device will assign a 3-bit address to each of its slaves. Each slave receives its assigned address in the piconet, and attaches it to all packets it sends to its master.

Figure below shows the state transition diagram for setting up a connection between devices

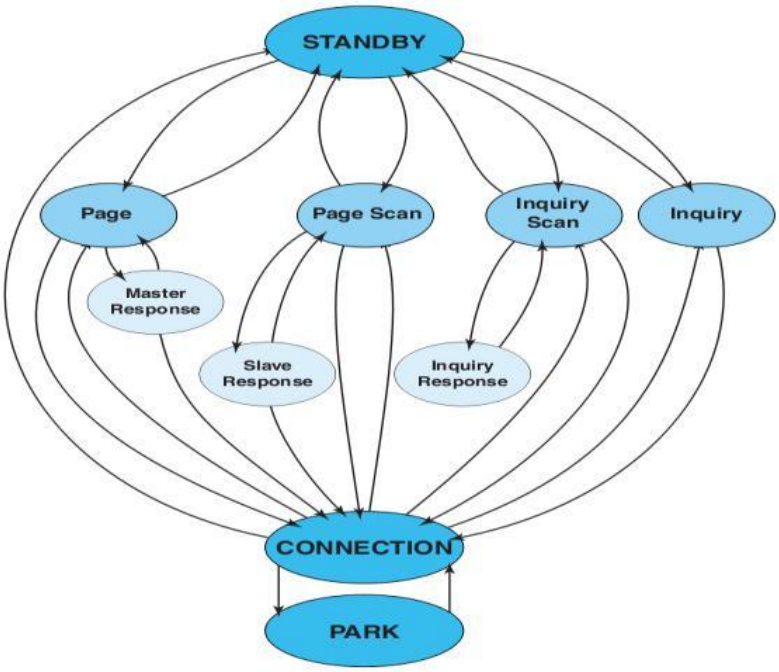

Figure 4: Shows the state transition for piconet devices (Ahmed, 2009)

In the above state diagram, in the proposed model a device should be able to log off the user if the timeout value set by the user has expired. In order to achieve value of security it is recommended for the user to set a low value such that the device can be able to logout quickly before any harm is done. Setting a high timeout value will keep the piconet open for a long time, allowing for harm to be done. However short value will make the device to be logging off more often and hence for a genuine user it will be annoying and discouraging. The states are discussed below.

2.4.1 Standby state

In a standby state the device is in low power mode and only the native clock (CLKN) is running, the Bluetooth device is in its default state.

2.4.2 Inquiry state

It is not clear of how often a unit should leave its standby or connection states to perform inquiry but it might be periodic or can be triggered by a user request. The task of deciding on this is left the implementers. If a unit wishes to discover other Bluetooth units in range it enters an

inquiry state and continues transmitting an inquiry message at using different hop frequencies. In the inquiry state the transmitting and receiving frequencies follows the inquiry hopping sequence and inquiry response hopping sequence that are determined by the General Inquiry Access Code (GIAC) and the native clock of the discovering device. Hopping sequence contains two groups of frequencies: train A and train B (each16 frequencies long). In between inquiry transmissions the unit listens for responses (which is an FHS packet). If a response is received it is not acknowledged and the unit continues with the inquiry transmissions. The inquiring unit will only leave the inquiring state after it has either received a preset number of responses or when the Inquiry time ends.

2.4.3 Inquiry scan

As mentioned in enquiry state, Bluetooth units leave Standby or Connection states to scan the channel for inquiries periodically. The period of inquiry scan can be 0s (continuous scan) – R0, 1.28s – R1 mode or 2.56s – R2 mode. A scanning unit listens for an IAC for 10ms and in this duration the receiver of the scanning device will listen on a single frequency determined by the inquiry scan hopping sequence and the current value of the device's clock. The scanning device always changes its listening frequency, in regard to inquiry hopping sequence, every 1.28s. The figure below shows the inquiry process.

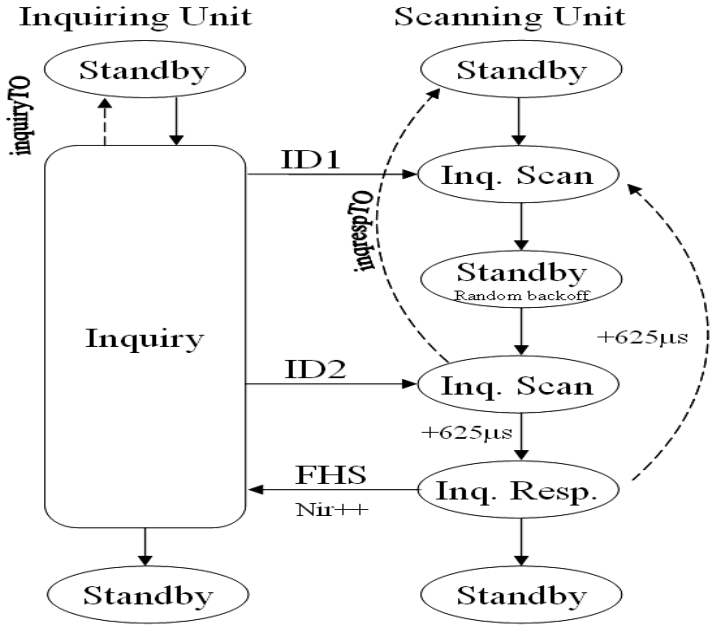

Figure 5: The Bluetooth Inquiry Process (Isaksson, 2004)

2.4.4 Page scan

Page scanning similar to the inquiry scan, but in page scan a unit will listen for its own unique DAC in order to respond. In page scan there are 32 paging frequencies, which comprises of a page hopping sequence, which is determined by the paged unit's BD_ADDR. In every 1.28s a different listening frequency is selected and during a scan window the unit listens only on one frequency.

2.4.5 Paging

When a Bluetooth unit wants to make a connection to another unit it pages with that unit. Paging is the sending of an ID packet with certain DAC in it several times until a response is received. The Master does not know exactly when the slave wakes up and on which hop frequency, hence it transmits a train of identical DACs at different hop frequencies at the same time listening in between for responses. The master uses the slave's BD_ADDR and an estimate of the slave's clock to determine the page hopping sequence. To cater for the uncertainty in the knowledge of a slave's clock, the master sends its page message during a short time interval on a number of wake-up frequencies. During each transmission slot the Master will sequentially transmit on 2 different hopping frequencies. The page hopping sequence consists of 32 frequencies which are divided into two trains of 16 frequencies each. If a slave responds to the first out of two paging messages in a slot then the response packet will be the same as the paging packet and is sent on the same frequency. The master will respond with an FHS packet on the next frequency in the page hopping sequence. The slave acknowledges the master's FHS packet and after that both units are in Connection state. The master takes responsibility of controlling all transmissions. For example it polls the slave to see if it has data to send, if it has it will respond with a data packet if not it will respond with a NULL packet. The slave will keep listening as long as the FHS packet is not received until pageregpTO is reached and in every 1.25s it will change the hop frequency according to page hop sequence. When nothing is received the slave returns to page scan for 1 scan period and if no pages are received during this interval, it will continue scanning and then return to the state it was in before, but if no poll packet is received by the Slave or response is not received by the Master within the period of newconnectionTO number of slots, they will return to page/page scan states.

2.4.6 Connection state

If the connection state channel hopping sequence is used it will be derived from the master's BD_ADDR. A Bluetooth device in connection state can be in one of the following four modes described below (Dahlberg, 2002).

- **Active mode**-In the active mode the Master schedules the transmissions based on traffic

  demands to and from different slaves. Active slaves listen in master-to-slave slots.
- **Sniff mode**-In order to save battery power a sniff mode can be used. In this mode the duty cycle of slave's listen activity is reduced. To enter sniff mode, the master shall issue a sniff command through the Link Manager (LM) protocol.
- **Hold mode** -the ACL link to a slave will not be supported temporarily but the slave unit keeps its active member address (AM_ADDR) during hold mode. During the process of entering the hold mode both Master and Slave agree on the holdTO value, when the value expires the hold mode ends.
- **Parked mode**-A Slave is in this mode when it does not need to participate in the piconet channel, but remain synchronized to the channel. In this state the Slave gives up its AM_ADDR, and gets two new addresses i.e. the Parked Member.Address (PM_ADDR, 8 bits) and the Access Request Address (AR_ADDR, 8 bits). They are used for master-initiated unpark and slave-initiated unpark respectively. A Parked slave wakes up at regular intervals to both listen to the channel and to re-synchronize, to check for any broadcast messages.

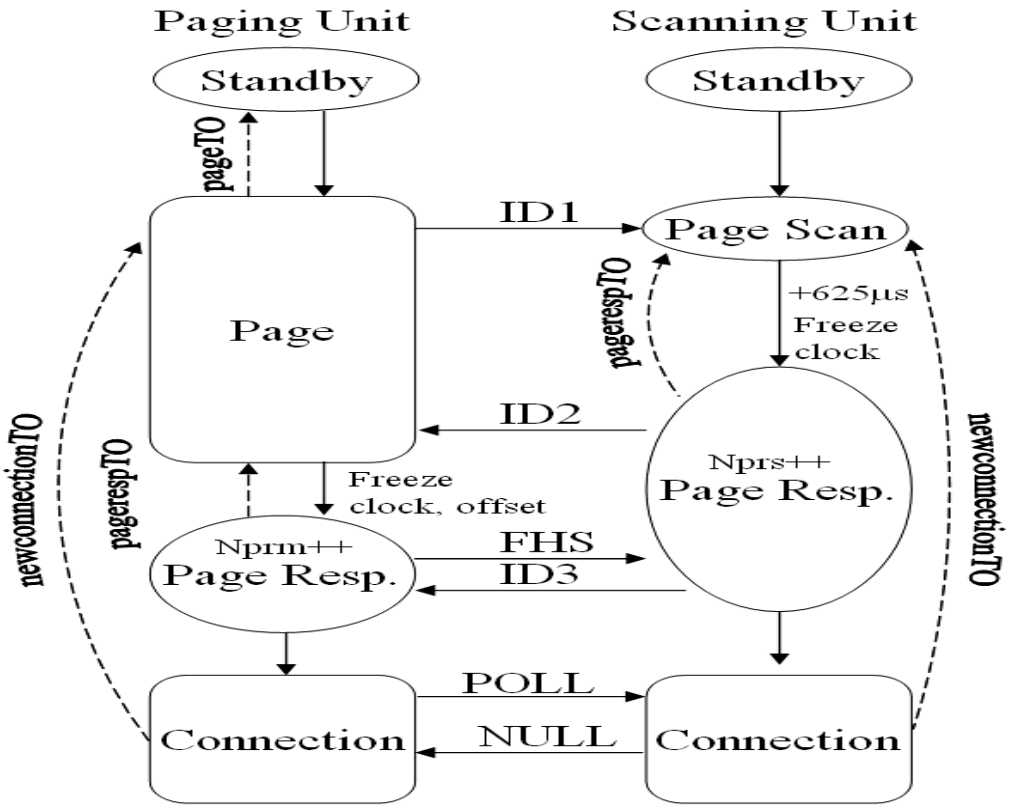

Figure 6: Shows the Paging and the connection procedure (Isaksson, 2004)

2.5 Bluetooth Profiles

Bluetooth Special Interest Group (SIG) with over 1500 member companies is concerned with developing Bluetooth features and capabilities and has achieved alot by developing more than sixty profiles. Bluetooth profiles are specific characteristics or uses for the standard. These profiles are of small differing protocol stack to facilitate handling of the type of necessary communication for each particular application. For example, the serial profile is used to emulate RS-232 communications over radio frequency (RF), other profiles are dialup networking (DUN), Service Discovery (SDP), local area network (LAN), headphones, and synchronization; just to name but a few. Just recent innovations, is the use of Bluetooth in automobiles opening up to many new features such as hands-free operation of a cellular phone. Other many more features being discussed are the automation of climate control, sound systems, seating, and ignition. Car manufacturer giants such as Daimler-Chrysler, Ford, and BMW have manufactured modules/options for their brand of vehicles which are available in the market today. Another area of application is the integration of Bluetooth to the industry to replace easily damaged cables in

systems control, inventory, troubleshooting and automation, which will result in improved flexibility and productivity. (Niem, 2003)

2.6 Analysis of Bluetooth Security

Below is a Bluetooth security architecture that can help us understand and visualize how Bluetooth security works.

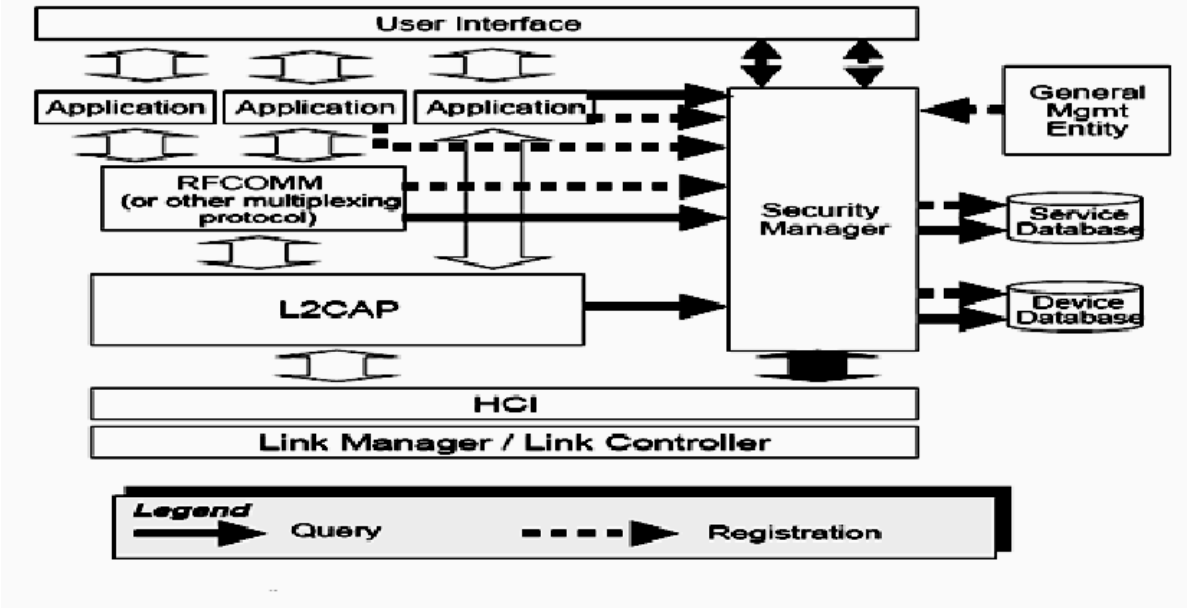

Figure 7: Security Architecture of Bluetooth (Ajay, 2008)

2.6.1 Security architecture.

In the security architecture above the key security components are security manager, device database and service database responsible for the following functions:

i. Storage of service security information
ii. Storage of device security information
iii. Responding to access requests by protocol implementations or applications (access can be granted /refused)
iv. Enforcing both authentication and encryption before connecting to the applications.
v. Initiating or processing input from an ESCE (External Security Control Entity.) in order to set-up trusted relationships at device level.
vi. Starting pairing and querying of PIN entry both by the user or by an application

This is the area in which this thesis will focus on. The components that this thesis will focus on are the user interface, the security manager, device database and service database. The security architecture has the following shortcomings.

- It does not support legacy applications in all scenarios since they do not make calls to the security manager; instead a Bluetooth-aware adapter application is required to make all necessary security related calls on their behalf.
- It is the device that is authenticated and not its user, other means of application level security features needs to be employed in order to authenticate a user. This thesis seeks to provide a solution based on this limitation.
- No authorization per service. This architecture allows for implementation of a flexible security policy without making changes to Bluetooth protocol stack however, it will involve making changes to the security manager and the registration processes. Where devices can be registered as trusted meaning that they can access services fully and others untrusted meaning that they cannot access services fully.
- In the approach access control is only allowed at the connection set-up and access check can be done one way, but once devices are connected the data is allowed to flow bidirectional, hence it is not possible to enforce unidirectional traffic in this architecture.

Currently there are three security services specified by Bluetooth, which must be met in order for Bluetooth communications to be secure. These services are (Panigrahy, Jena, & Turuk, 2011);

i. **Authentication**-It verifies the identity of communicating devices. User authentication is not provided by Bluetooth. This creates the basis of this project.
ii. **Confidentiality**-Confidentiality is where only authorized devices can access and analyze data. Confidentiality prevents eavesdropping hence ensuring the integrity of information.
iii. **Authorization**-Controls access and usage of resources by ensuring that a device is authorized to use a service before permitting it to do so.

In order to achieve the above capabilities, Bluetooth offers three security modes (Gehrmann & Nyberg, 2002)

i. **Security Mode 1**: Provides no security and access is granted to all devices. Its open and no authentication and encryption is required.

ii. **Security Mode 2**: Requires authentication only and authorization is not necessary. Access to an application is allowed only after an authentication procedure, individually addressed traffic is encrypted using encryption keys based on individual link keys and broadcast traffic is not encrypted.

iii. **Security Mode 3**: Requires authorization and authentication and automatic access is granted only to trusted devices, untrusted devices need manual authorization in order to access resources. Traffic in this mode is encrypted using an encryption key based on the master link key.

2.6.2 Piconet device authentication procedure

Bluetooth device authentication procedure is in the form of a challenge-response scheme. Each device is in an authentication process is referred to as either the claimant or the verifier. The claimant is the device that wants to prove its identity, while the verifier is the device that validates the identity of the claimant. The challenge-response protocol validates devices by using the knowledge of a secret key, i.e. validating Bluetooth link key (Kitsos, Sklavos, Papadomanolakis, & Koufopavlou, 2003). The challenge-response verification scheme can be shown conceptually by the figure bellow.

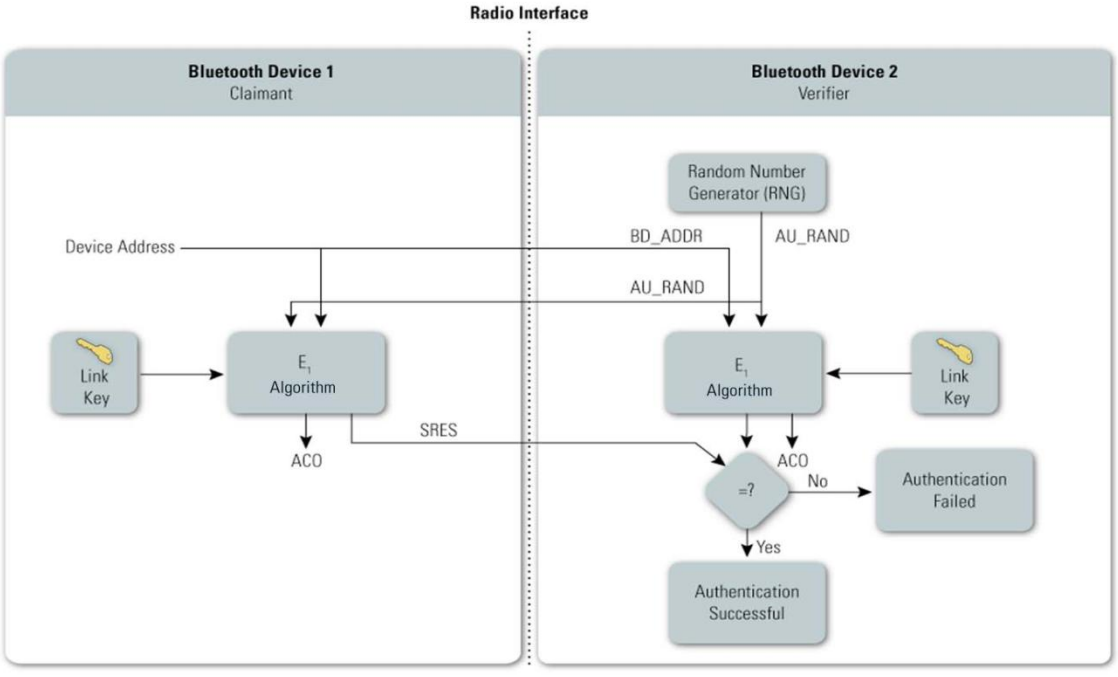

Figure 8: Bluetooth Piconet Authentication Process (John, & Karen, 2008)

The above process is as follows.

- **Step 1.** The verifier device sends a 128-bit random challenge (AU_RAND) to the claimant device.
- **Step 2-** The claimant device uses the E1 algorithm to generate an authentication response using its unique Bluetooth device address (BD_ADDR 48 bit), the link key and AU_RAND are the inputs. The verifier device performs the same process and only the 32 most significant bits of the E1 output are taken and used for the authentication process. The 96 bits of the 128-bit output will remain and are used later to create the Bluetooth encryption key, known as the Authenticated Ciphering Offset (ACO) value.
- **Step 3**-The claimant device sends to the claimant the most significant 32 bits of the E1 output as the generated response, SRES.
- **Step 4**-The verifier device receives the SRES from the claimant and compares it with the value that it had generated.
- **Step 5-** If the two SRES values are equal, the authentication process will be considered successful while if not equal, the authentication process will have failed.

The steps above are a one-way authentication but the Bluetooth standard allows for two-way mutual authentication to be performed. If mutual authentication is to be conducted then the above process is conducted again with the verifier device and claimant device switching roles.

2.6.3 Pairing mechanisms

Pairing mechanisms have changed significantly with the introduction of Secure Simple Pairing in Bluetooth piconet. The following summarizes the pairing mechanisms(Khan, 2010):

- **Legacy pairing**: This is the only method used by Bluetooth version 2.0 and before. Each Bluetooth device must generate a PIN code and pairing process between the two devices is only considered successful only when both the devices generate the same PIN code. Also any 16-byte UTF-8 string may be used as a PIN code; however some Bluetooth devices don't have the capacity of entering all possible PIN codes.
- **Limited input devices**: A good example of this class of devices is a Bluetooth Hands-free headset, which has got a limited input capability. These types of devices are usually manufactured with a fixed PIN hard-coded into the device, for example "0000" or "1234".

- **Numeric input devices**: Mobile phones are good examples of these type devices. They have input capability to allow a user to type numeric values of up to a maximum of 16 digits long.
- **Alpha-numeric input devices**: PCs and smartphones are good examples of these devices. They have capability to allow a user to enter a PIN code of full UTF-8 text. In pairing process with a less input capable device the user needs to have knowledge of the input limits on the other device, but on the other hand there is no mechanism available for a capable device to determine how it should limit users input.
- **Secure Simple Pairing** (SSP): This type of pairing process is used by Bluetooth Version 2.1. If Bluetooth Version 2.1 device is involved in a pairing process with a Version 2.0 device then it will be forced to use legacy pairing so as to interoperate with a or earlier device. In SSP, public key cryptography is used, with the following modes of operation:

    - **Just works**: This method just works without any user interaction being required, but in some situations the device may ask the users confirmation in the pairing process. This method is mostly used by headsets, because they have limited input/output capabilities. This method is more secure than the fixed PIN mechanism, usually used for legacy pairing by input/output capability limited devices. This method has no protection against the Man in the Middle (MITM) attack.
    - **Numeric comparison**: This method is used when both devices have a display and either one device can accept a binary Yes/No user input, Numeric Comparison can also be used. In this method a 6-digit numeric code is displayed on each device for the user to compare the numbers and ensure that they are identical. If identical the user confirms the pairing process on the device(s) that can accept an input. If the user checks properly and confirms the correct value on both devices, this method provides MITM protection.
    - **Passkey Entry**: This method may be used between a device with a display and a device with numeric keypad entry (such as a keypad), or in a situation where both devices have a numeric keypad entry. In the first case, the display is used to show a 6-digit numeric code to the user, who then types the displayed code on the

device with keypad. In the second case, the user of each device types the same 6-digit number using the keypad. In both cases, MITM attack is proteced.

- **Out of band** (OOB): This method, an external means of communication (such as NFC) to exchange some information used in the pairing process is employed. Pairing is achieved using the Bluetooth radio, but it requires information from the OOB mechanism to be provided. This protects against MITM that is availlble in the OOB mechanism.

2.7 Bluetooth Weaknesses and Threats

Even with the above security measures Bluetooth has been found to be weak and hence exposing its users to risks of losing their data to hackers and unauthorized users. Security threats in Bluetooth piconet can be divided into three categories i.e. the integrity threat, the disclosure threat, and the Denial-of-Service (DoS) threat. Disclosure threat is where, information leaks from the target system to an eavesdropper who is not authorized to access the information while integrity threat is where there is intended alteration of information in order to mislead the recipient. Denial of Service threat involves denying access to a service by making it either unavailable or by severely limiting its availability to an authorized user. (Tarique, 2011)

2.7.1 Bluetooth weaknesses

Bluetooth weaknesses and their mitigation recommendations can be summarized using the table that follows.

Security Issue or Vulnerability (Which affects all versions)	**Remarks**	**Security Recommendation**
1. Improper storage of Link keys.	An attacker can read or modify Link keys if they are not stored in a secure location and protected via controlling access to them.	Default settings should be changed, because they are not that secure, a careful review of security settings should be always done to ensure that unauthorized access to Link keys is limited
2. Unlimited repeated attempts for authentication. .	Limited authentication feature should be incorporated in the Bluetooth specification to prevent limit authentication requests. The Bluetooth specification currently requires a time-out period between repeated attempts that is increased exponentially.	Encrypt device address. Limit the number of authentication time.
3. Unknown strength of the pseudorandom challenge/response Generator is not known.	The Random Number Generator (RNG) may produce static number or periodic numbers that may compromise the effectiveness of the authentication process.	Statistical tests should be done on the pseudorandom to detect if it meets the quality of non-repeating and random generation requirements. Its quality should be tested.
4. Negotiable Encryption key length. .	The specification provides for devices to have the capability to negotiate encryption keys even up to a minimum of one byte. Hence calling for a more robust encryption, the key generation procedure needs to be incorporated in the specification.	Encryption key sizes should be set to a maximum allowable key, and a minimum key size for any key negotiation process should be established. For a more robust encryption, the key generation procedure needs to be incorporated in the specification
5. The master key is shared.	A better broadcast keying scheme needs to be incorporated into the specification.	Broadcast scheme of the shared keys should be changed.

Security Issue or Vulnerability (Which affects all versions)	**Remarks**	**Security Recommendation**
6. No user authentication scheme is available.	The specification provides only for device authentication. This calls for application-level security, including user authentication, to be incorporated via overlay by the application developer.	Add application level security and employ user authentication.
7. Weak E0 stream cipher algorithm that performs Bluetooth encryption. .	A more strong encryption algorithm needs to be incorporated in the specification.	Change the cipher by replacing it with other advanced scheme for better encryption strength.
8. Lead compromised Privacy, if the Bluetooth device address (BD_ADDR) is captured by hacker and associated with a particular user.	Once the BD_ADDR is associated with a particular user, that user's activities could be monitored or revealed, resulting in a loss of privacy.	Device address must be encrypted to make it secret. Limit the entry number of the list.
9. Device authentication is simple shared-key challenge-response.	One-way-only challenge/response authentication is subject to MITM attacks. Bluetooth provides for mutual authentication, which should be used to provide severity that users and the network are legitimate.	Make sure that device mutual authentication is performed at all connection establishments.
10. End-to-end security is not performed.	Only individual links are encrypted and authenticated. Data is decrypted at intermediate points. End-to-end security on top of the Bluetooth stack can be provided by use of additional security controls.	Access check at all the phases and mutually. Check-consistent data flow direction.
11. Security services are limited.	Audit, non repudiation of services and transactions, and other services are not specified in the standard. If they are needed, then can be incorporated in an overlay fashion by the application developer.	Modify the security manager and the registration processes. To accommodate more security features
12.Discoverable and/or Connectable devices are prone to attack.	A Bluetooth device that must go into discoverable or connectable mode in order to pair must only go for a minimal period. A device should never be in discoverable or connectable mode when not in use.	Bluetooth devices should be configured by default as, and remain, undiscoverable except if wanted for pairing process. Turn off Bluetooth devices when not in use.

Table 1: Piconet Weaknesses and Proposed Mitigation Mechanisms (Tarique, 2011) (Mishra & Gupta, 2012), (Panigrahy, Jena, & Turuk, 2011) and (Patheja et al. 2012).

2.7.2 Bluetooth threats

In this section, some of the major threats that affect the functionality of Bluetooth device most are discussed (Panigrahy et al. 2011);

- **Bluejacking-**Is categorized under denial of service attack. In this attack, attacker temporarily hijacks another person's cell phone by sending an anonymous text message to it using Bluetooth wireless piconet system. Bluejacking is started by an attacker sending unsolicited messages to a user of a Bluetooth enabled device. The actual messages do not cause harm to the user's device, but they are used to attract the user to respond in some fashion or add the new contact to the device's address book. If the user responds to a bluejacking message that is sent with a harmful intent then harm can be achieved, but if ignored no harm will be done. This message-sending attack resembles spam and phishing attacks conducted against email users (Gupta, Joshi, & Misra, 2009).
- **Reflection attacks-**Also referred to as relay attack which is based on the impersonation of target devices. An attacker does not have to know any secret information, it only relays (reflects) the received information from one target device to another during the authentication process.

- **Bluesnarfing/Bluestumbling-**is where a hacker hacks into a Bluetooth-enabled mobile phone and copies its entire contact book, calendar or any data stored in the phone's memory, without alerting the owner. By setting the device in non-discoverable, it becomes more difficult to find and attack it. The software tools used to carry this attack are widely available on the Web, and knowledge of how to use them is growing (Patheja, Akhilesh & Nagwanshi, 2011).
- **Warchalking and Wardriving-**Because of the nature of wireless communication, it is possible that private signal can be picked up by unintended people and that outsiders could connect into private WLANs. War driving involves using a laptop computer with a WLAN card and wireless scanner software in a car to detect wireless networks. When they have discovered a wireless network with external connectivity some individuals will share the details using symbols chalked on pavements or walls (i.e. warchalking) (Mishra & Gupta, 2012).

- **Bluespamming-**Is the sending of unsolicited commercial messages (i.e. direct marketing via email, telephone and SMS spam, etc) without consent(Kotadia, 2004).

- **BluePrinting-**Is an attack used to determine the manufacturer, model, and firmware version of the target device. For example, an attacker can use Blueprinting to generate statistics about Bluetooth device manufacturers and models, and to find out whether there are devices in the range of vulnerability that have issues with Bluetooth security to launch an attack (Mishra & Gupta, 2012).
- **Bluetoothing-** Bluetoothing is typically social networking in a short range, but there is a possibility of harassment from the security point of view. Bluetooth PIN code can also be cracked by programmers as well (Patheja et al. 2011).
- **Bluebugging-**In this attack, an attacker try to read data on a Bluetooth enabled cell phone, eavesdropping on conversations and even sending executable commands so that the hacker can actually make phone calls, add or delete contact info, or eavesdrop on the phone owner's conversations (Hossain, Kabir & Rahman, 2011).
- **Bluetracking-**In this an attacker tries to track people's locations by following the victims Bluetooth devices signal.
- **Bluesnipping-**in this, an attacker does scanning with a Bluetooth scanning device that looks like a sniper rifle with an antenna instead of a barrel(Kotadia, 2004).
- **Off-Line PIN Crunching-**In this attack, attacker try to intercept the traffic of the initial pairing process and after that trying to calculate the correct SRES (Signed Response; i.e. authentication result) value by guessing different PIN values until the calculated SRES equals to the intercepted SRES.
- **Man-in-the-Middle Attack-**It is a form of active eavesdropping in which the attacker makes independent connections with the victims and relays messages between them, making them believe that they are talking directly to each other over a private connection, when in fact the entire conversation is controlled by the attacker or typically bombarding the device with requests to the point that it causes the battery to degrade. (Hypponen, & Haataja, 2008).
- **Battery Exhaustion attack-**It is based on the idea of making the target device busy in such a way that it consumes its battery power quickly than normal.

- **K. Mabir**-This worm uses both Bluetooth and MMS to multiply. It also sends an MMS in reply to any received SMS, which is a clever technique to fool the user into installing the received application (Sharma, 2008).
- **L. "Backdoor" hacking-**This is where un trusted device/no longer trusted can still gain access to the mobile phone. It gains access to data and services such WAP etc or as like Bluesnarfing (Sharma, 2008)

2.8 Related Work

In a paper done by (Khan, Ariful, & Mohammad, 2010), they have analyzed Man-In-The Middle attack on Bluetooth Secure Simple Pairing, and done modification the Secure Simple Pairing in order to improve the security of pairing and authentication process of Bluetooth in an effort to prevent Man-In-The-Middle attack. Also In another paper done by (Iqbal, Kausar, & Wahla, 2010) they have highlighted that the current Bluetooth security architecture is vulnerable to MITM attack, and have presented different forms of DoS attacks and hostile intruder detection in a piconet. Consequently, they have devised a protection mechanism against all these attacks. They protect aginst MITM by keyed hash of the link key (unit key) before the generation of the encryption key. They also present the concept of cookies to protect aginst the DoS attack due to many authentication requests, and provide a hostile intruder detection mechanism carried out by including piconet specific information in SRES messages. And since its possible for DoS attack after a device has been authenticated they overcome it by setting the flag “on” against the connected device MAC ADDRESS, which helps in detecting a DoS attack from the assumed attacker.

(Hossain et al. 2011) In their paper have presented an authentication approach for protection against the bluebug attack where they have modified the RFCOMM procedure with existing process. RFCOMM uses a database for storage of Bluetooth device address or Mac address. The database stores all Bluetooth device address with location, company name, description, manufacturer date, it's time or when it was manufactured and any other necessary information about the device. After scanning, when there is a match with the device address that is already registered or in database store, then RFCOMM gives permission to communicate with the particular device, otherwise it should not get permission.

In a paper done by (Patheja et al., 2011) they analyzed the 128-bit symmetric stream cipher called E0 and prove its weaknesses under some conditions, they found that it can be broken under certain conditions with the time complexity of (2^{64}). To improve it they have proposed a hybrid encryption technique. A triple DES for encryption of the key for which they use Tiger algorithm. The Tiger algorithm provides a double protection of Data using triple DES and with the help of this algorithm the security of data transmission between Bluetooth devices is enhanced.

All the above solutions do not offer any solution to the problem of lack of user authentication, in a paper by (Patheja et al., 2012) luck of user authentication has been listed as one of the Bluetooth weaknesses. This shows that not much has been done in addressing the security issues that arise from lack user authentication in the Bluetooth. This thesis work is intended to fill the gap that is there through the development of a user authentication model for the Bluetooth piconets. By having a user authentication then the Bluetooth device will be able to determine a genuine user (with password) and not a genuine user (without password).

CHAPTER THREE: METHODOLOGY

3.0 Introduction

The methodology will detail about the type of methodology, techniques, project requirements and thesis plan which will guide the researcher in meeting research objectives.

In this research there was no data collection from humans or quantitative data, instead secondary data was collected. A study was carried out in finding facts and information from books, journals, whitepapers and research reports dealing with Bluetooth architecture, Bluetooth security and user authentication methods. The journals, whitepapers and research reports were selected according to their year of publication, from the year 2002 to the year 2012. The research design used was descriptive in nature, which endeavored to provide an overview of the Bluetooth piconet architecture, its security and its vulnerabilities.

3.1 User Login Model Development

Different mobile phones use different operating system from each other this is because they are manufactured by different manufacturers whom their interests are vested in different operating systems. Some platforms and technologies used are such as Window Mobile OS, Palm OS, Symbian OS, Macromedia's Flash Lite OS, Python OS and Sun's J2ME (Java 2 Micro Edition) OS.

3.2 Review of Current Methods for Development.

Below is a summary of various options for application development for mobile phones that were considered. The option selected was limited by the mobile phone model used in this project. The methods were reviewed as follows.

Platform	Overview
Java ME	Second best reach, best overall development
Flash Lite	Good for graphics-heavy applications in supported markets
Symbian	Strong support from Nokia, best access to hardware
.NET	PocketPC + Windows Mobile Devices
BREW	The only option for CDMA networks
Python	Great for quick prototypes, still immature
WAP	Largest overall reach, lightweight functionality

Table 2: Summary for approaches of application development for mobile phones (Yen, 2011)

3.3 Evaluation of the Current Methodological Approaches.

There are some advantages in choosing J2ME because J2ME has the largest availability among the others by a large range and extensive developer community with lower time for re-implementation and porting. The chart below shows the relative comparisons between the different development options likely to be used.

Platform	**Programming Language**	**X-Platform**	**Learning Curve**	**Emulator Used**	**Use in Devices**
Java ME	Java	Average	Average	Free	1.5 billion
Flash Lite	AS	Excellent	Average	With IDE	77-115 million
Symbian	C++	Average	Steep!	Free	120 million
.Net	C, C++, VB.NET	WM	Steep!	IDE	4.5 million
Brew	C++	CDMA only	Steep!	Simulator	????
Python	Python	Free	Gentle	Add-on	Nokia-only
WAP/Mobile Web	XHTML, WML	Free	Gentle	Free	2 billion +

Table 3: Comparisons of the possible approaches

3.4 Selection of Appropriate Approach.

It is difficult to develop an application with the capability to run across all the different operating system. However, Java and J2ME are designed to provide a platform independent development tool. J2ME platform runs on many mobile devices, which have installed a Java Virtual Machine. The benefit of using the Java Platform for mobile device development is that, portable code that can run on multiple platforms is produced. Not always can all functionalities of an application be implemented in all mobile devices, because of limitations in terms of processing power, memory, battery life, display size, and network bandwidth and this must be taken into account during development of this model. The available mobile phone for testing the login application for user authentication is a Nokia Asha 200 which supports the J2ME applications. Therefore, the method used was selected based on the available phone, and the best choice used was Java ME. In developing a Java based login application for mobile phones, one important aspect must be considered: The J2ME programming skills. In order to run Java application on mobile device, J2ME code is necessary. Java micro media edition is a Java platform specification which is special in focusing on providing a certified collection of Java APIs for small, resource-limited devices such as mobile phone, PDAs and set-top boxes.

3.5 Characteristics of the Selected Method

In developing a Mobile midlet and Bluetooth based application that work on real time on a mobile phone requires a selection of nice techniques. The mobile platform will be limited by its computation speed and storage capacity. J2ME is one of the most common platforms for mobile application and games because it has software libraries that are able to provide high-level user interface features such as lists and checkboxes. This characteristic enables developers to create better and user-friendly interface for their application. The applications developed under J2ME are portable between different operating systems. Therefore, J2ME application is the best way in developing mobile phone application. The following graph show where J2ME is ranked in java development world.

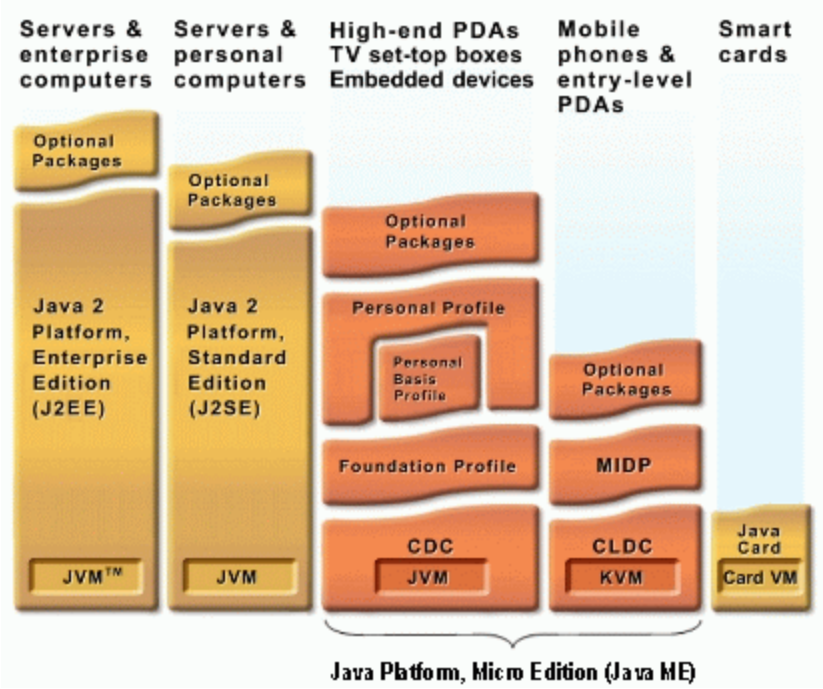

Figure 9: The J2ME Platform (Yen, 2011)

The above figure shows where J2ME lays in the Java development world. There are configurations, profiles, and optional API packages in Java micro media edition. Configurations mainly concerns about the configuration of device memory and the minimum of APIs needed for development of applications that can run on a range of devices. J2ME configuration is divided into two i.e. Connected Device Configuration (CDC) and the Connected Limited Device Configuration (CLDC).

3.5.1 Connected limited device configuration (CLDC)

CLDC was created for small end of the consumer electronics range of devices. An example of CLDC platform is a cell phone or a PDA with around 512 KB of memory capacity, hence CLDC is closely related with wireless Java, which is used in letting the mobile phone owners/users to buy and download small Java programs known as MIDlets to their mobile devices. Many and growing number of mobile phone manufacturers have entered agreed with Sun Microsystems to let them to start using this technology; this has resulted in an increased number of mobile devices with the capability to support Java.

3.5.2 Connected device configuration (CDC)

CDC is implemented on high-end PDAs and smart phones, residential gateways, web telephones, and set-top boxes. CDC satisfies the needs of devices that lie in between those supported by CLDC and those that support desktop systems, running J2SE. These are devices that have bigger memory (2 MB or more) and more powerful processors than those supported by CLDC, and therefore they are capable of supporting a much more complete Java software environment. Each configuration contains of a Java virtual machine and a collection of core Java classes for programming environment for the application software. K Virtual Machine (KVM) is a specification implementation of CLDC which is based on a small JVM. The KVM only includes a subset of the bytecodes validation and it does not allow native methods to be added at runtime. Its primary focus is on Mobile Information Device Profile (MIDP) and there are MIDP 1.0 (JSR 37) and MIDP 2.0 (JSR 118). MIDP 2.0, which enables the support on multimedia (java.microedition. media), game user interface API (java.microedition.lcdui.game), and many other important features for image transferring application of mobile phones. The phone available and to be used on this project will need to support MIDP2.0.
MIDP hardware minimum requirements are as follows.

- 128KB of RAM for the JRES
- 256KB of ROM for the MIDP API libraries
- 8KB of non-volatile writable memory for persistent application data
- Screen size of 96x54 pixels with 1-bit color depth (black and white at least)
- Some input device, either a keypad, keyboard, or touch screen
- Bluetooth enabled

Some optional API packages include functionality that will only be supported on certain devices. Other APIs are Advanced Graphics and User Interface (JSR 209). There are some APIs which provide access to particular features and functionality. MIDP applications are normally called MIDlets, the diagram below shows the lifecycle of a MIDlet.

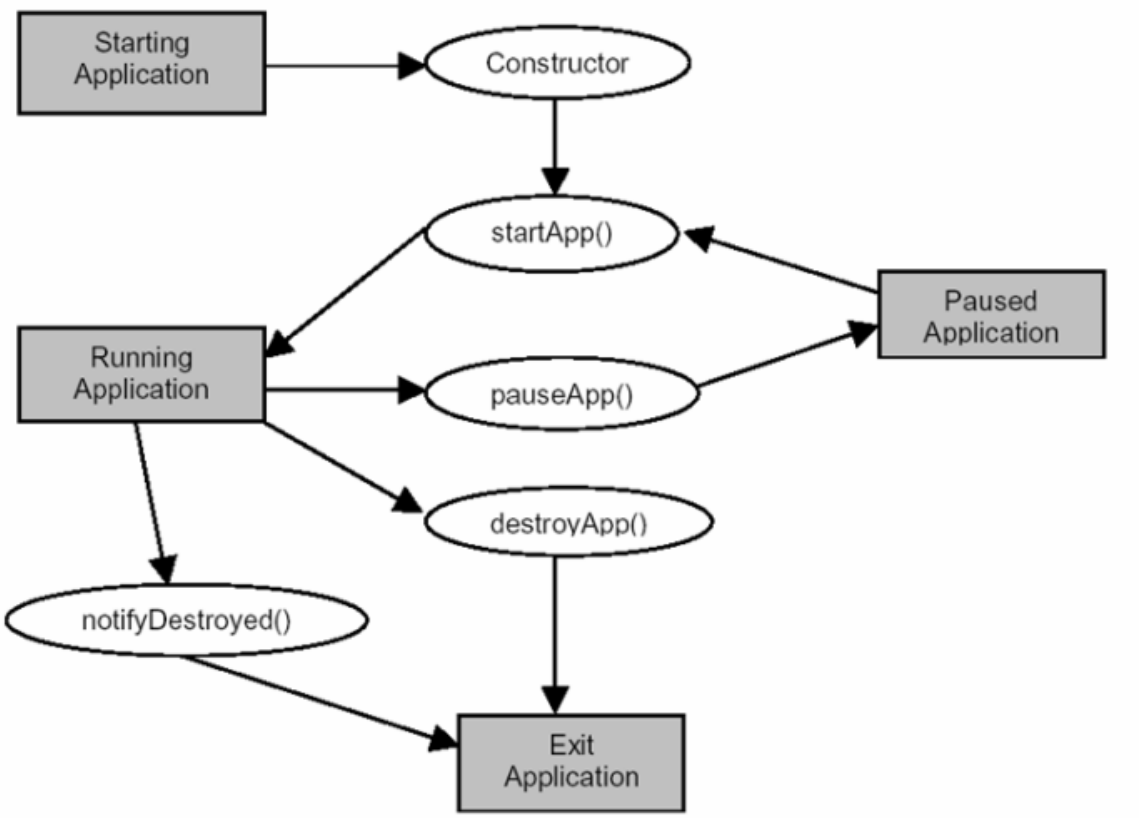

Figure 10: The States of a MIDlet and their transitions (Yen, 2011).

J2ME is a reduced version of J2SE, its size is smaller and it does not contain classes like swing and awt, its User Interface is based on succession of screens and are not subsets of AWT/Swing. MIDP provides limited UI elements that are Form, Alert, Choice and List, StringItem, ChoiceGroup, TextField, DateField, Guage, TextBox and Ticker.

3.6 Multi-Threading

Threads are lightweight processes in which each thread is able to run independently from the other, or they can run at the same time (parallelism). Multi-threading happens when a single program uses many threads to run different sections of the program at the concurrently. Threads will be used to avoid resource wastage during the times when a process is blocked, leaving processing resources idle.

CHAPTER FOUR: DESIGN

4.0 Introduction

The following factors have to be put into consideration when designing the user authentication model this is due to the fact that the login system to be developed will be developed and be tested on a mobile phone device which has special requirements.

4.1 Functioning of the Login System.

The User Authentication model will have three functional objects, the user, the login manager and the database. The user will invoke the login system which will send inquiries to the database through the login manager. The activities of the login system start when the user wants to put the Bluetooth on in order to use it. If the Bluetooth is being used for the first time the user will be requested to provide a username, password and timeout value for the Bluetooth, hence resulting in creation of an account. A message of successful creation of an account will be displayed to the user. If it is for the second time or other consecutive times the Bluetooth is being in use, then the account must have been created before. All the user needs is to remember the password and enter it to the device for verification. Once entered if its correct login will be successful and if entered wrongly, login will be denied and a message displayed for unsuccessful login. Putting off the Bluetooth will not request for a password and the user will be the responsible for remembering the password for later use. Changing the password will be allowed but only on a condition that the user enters both the old password for verification and the new password to effect the change. The username, password and the session timeout value will be stored on the device database or the service database, a decision on which database to use will be made later. Session timeout values will be provided to the user as options for the user to select from but it will be recommended for the user to use the shortest values, so that once the device has been lost/stolen, and the owner had logged on then the device will be able to log off within a very short time before an harm can be launched against the other devices it had connected with.

4.2 Data Requirement

The data requirements for this model will be of access control in nature. A user interface will be provided to allow the user for easier and guided input of the data. The login manager will be able to process the queries to the database where the data will be stored. Any changes made will also be catered for and stored in the database.

The username and the timeout value will be stored in plaintext but the password will be encrypted and made long to avoid it being cracked or guessed. However the application will not be able to guard against the following situations.

- Where unauthorized person gains access to the device and reads the information stored in the device, by reading the devices password file.
- Unauthorized person secretly observes the user entering the password and uses it later, to access the Bluetooth device without the owner knowing. Also in situations where password checking program is used and password cracking.
- Where the user reveals the password to a friend or where an easily guessable passwords are used.

4.3 Software Requirements

The following are the software tools that will be required in order to develop the user authentication application.

- NetBeans IDE version 7.2
- Java Development Toolkit version (JDK) 7 update 6
- Java Runtime Environment (JRE)
- Java Micro Media Edition (J2ME)
- The login system for user authentication will be developed on a laptop with Windows 7 Ultimate Edition platform. Microsoft Office will be used to create the thesis documentation and UML program for design purposes.

4.4 Hardware Requirements

The login system will be able to perform well if it meets the requirements of the underlying device. Considering the fact that mobile phones are constrained in terms of memory, processing power, displays and the battery power, the device emulators provided by the J2ME software will be used to emulate the device hardware. The application will be tested on the emulated devices and then if found working correctly it will be deployed to the real device. The real device will be required to have Bluetooth capability. The laptop to be used in coding and testing the application will be a Pentium Dual Core with processing capability of 2.30 GHz and a RAM of 2.0 GB.

4.5 Conceptual Model for the Login System

The conceptual design for the proposed model contain three main components namely user, login manager and a database as shown below.

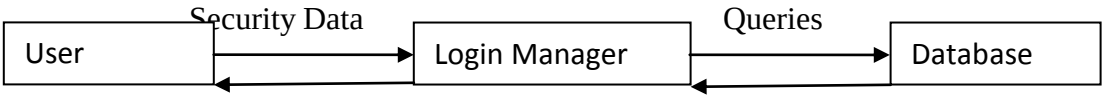

Figure 11: The conceptual model

4.5.1 User

The user will be required to enter the security data into the login system which will be used to secure the Bluetooth; any other user without the required data will not be authenticated to use the Bluetooth. Authorized user must have a password.

4.5.2 Login manager

The login manager will be responsible for managing the following

i) Enforcing user authentication to the Bluetooth device before using Bluetooth system.

ii) Answering access requests by the user (access can be granted /refused)

iii) Responsible for updating and extracting information from the security databases. Other important tasks that will be handled by the login manager will be to require the user to enter a username and password before granting access and to issue an authorization response when a remote device tries to connect to a service that requires authorization. The login manager must also provide a user interface to change username or password on the device.

iv) Enforcing logoff when the timeout value set by the user expires.

4.5.3 Database

In order to keep track of which user is authorized, login information needs to be stored in security databases. Two databases are used in Bluetooth, one for devices and one for services. The database will be responsible for the storage of user login-related information.

3.6 Use case Diagram for the Model

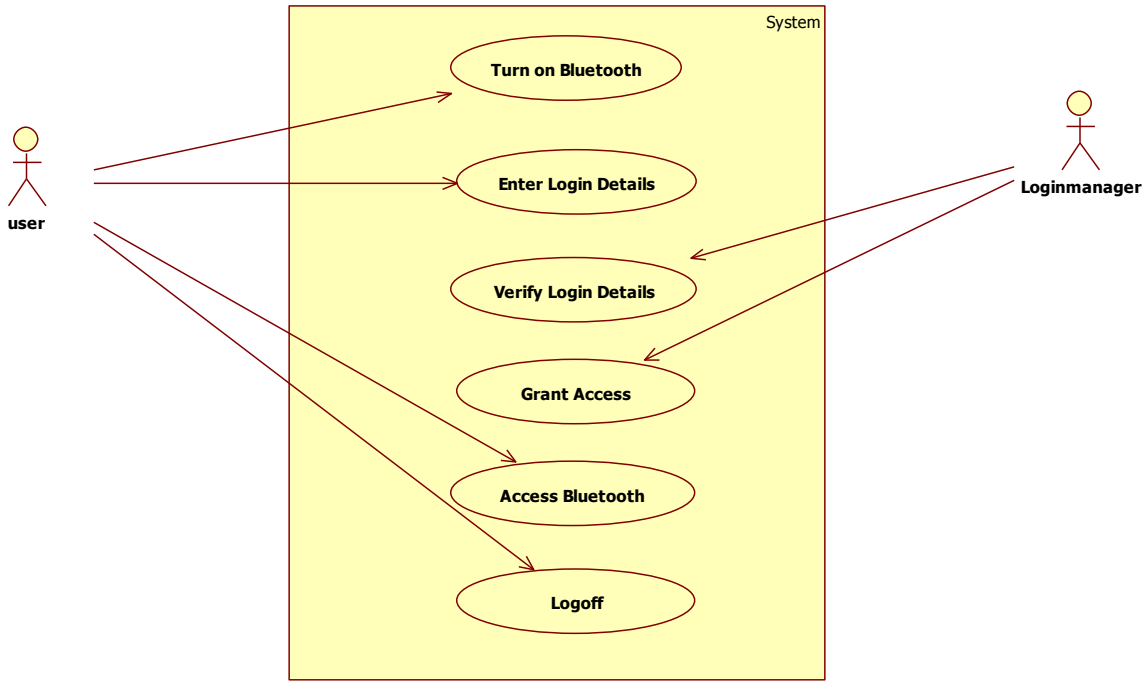

Figure 12: Use Case Diagram.

The Logon steps can be specified by the following steps:

i. **Use-Case name: Turn on Bluetooth**-The user turns on the Bluetooth by clicking on its icon or pressing the Bluetooth button, the selects turn on option. The login dialog is displayed to the user automatically.
ii. **Use-Case name: Enter Login Details-**The user enters the login details in the login dialog then presses the login button to continue.
iii. **Use-Case name: Verify the login Details**-The login manager then verifies the login credentials by querying the details from the database to check if they are correct.
iv. **Use-Case name: Grant access-**If the login credentials are correct the login manager grants access to the Bluetooth system. If login credentials are not correct access will be denied.
v. **Use-Case name: Access Bluetooth-**The user will access Bluetooth normally after logging in, and use it for data communication.

4.6 Collaboration Diagram

User login collaboration diagram emphasizes the order in which things happen when the user logs into the Bluetooth system.

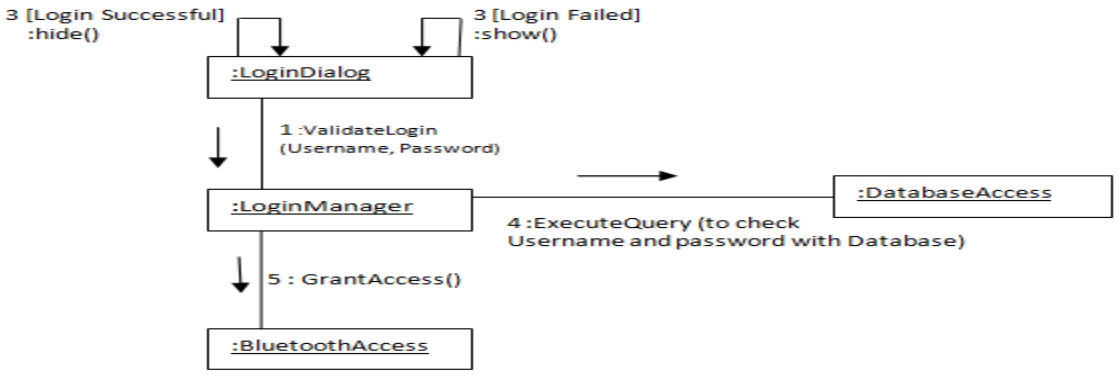

Figure 13: The Collaboration Diagram of the User Authentication Model

The order in which the login system works is as listed below.

1. Login dialog is shown to the user
2. User enters user name and password
3. User clicks on OK or presses the enter key
4. The user name and password are checked and approved
5. The user is allowed into the Bluetooth system

Alternative: Login Fails in step 4, if the user name and password are not approved, the user will be allowed to try again up to three times.

4.7 Sequence Diagram

The sequence diagram shows the system interactions from the user, and the messages exchanged.

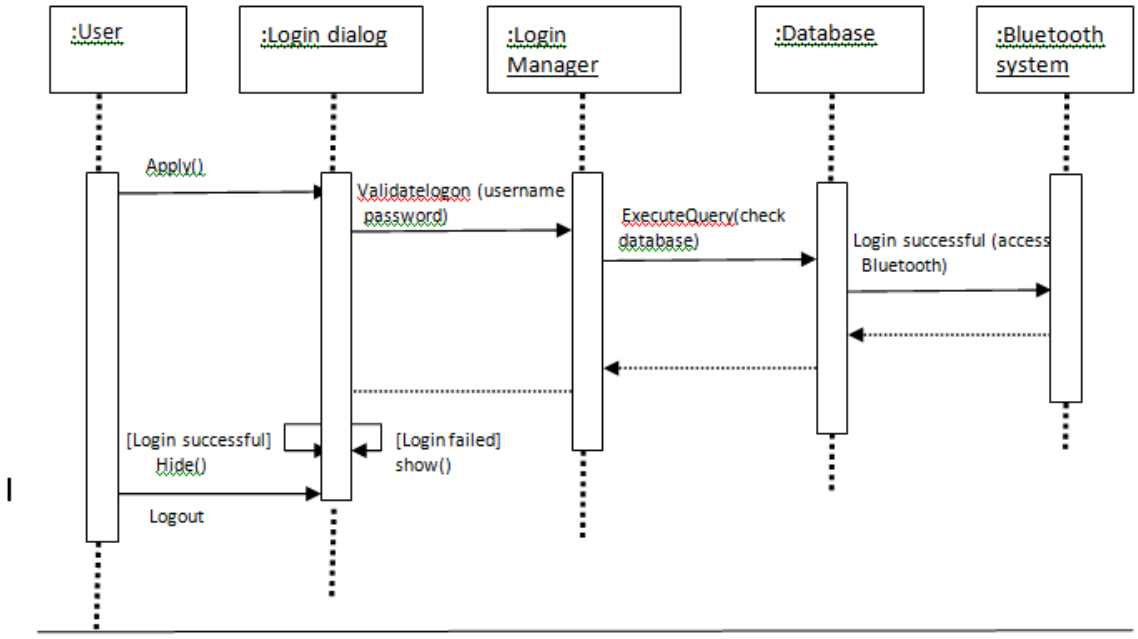

Figure 14: The sequence diagram of the model

The user logs on to the system by entering username and password, if they are correct the system responds to the user showing a login successful message and the grant an access to the Bluetooth system. If the username and password is not correct the system shows a login failure message to the user and denies access but offers the user another chance to try to login until three times.

CHAPTER FIVE: IMPLEMENTATION AND TESTING

5.1. General Introduction

This chapter shows the login system implementation, one of the goals of this thesis is to achieve implementation. In order to implement this project J2me programming language was used for coding.

5.2 The Implementation Diagram.

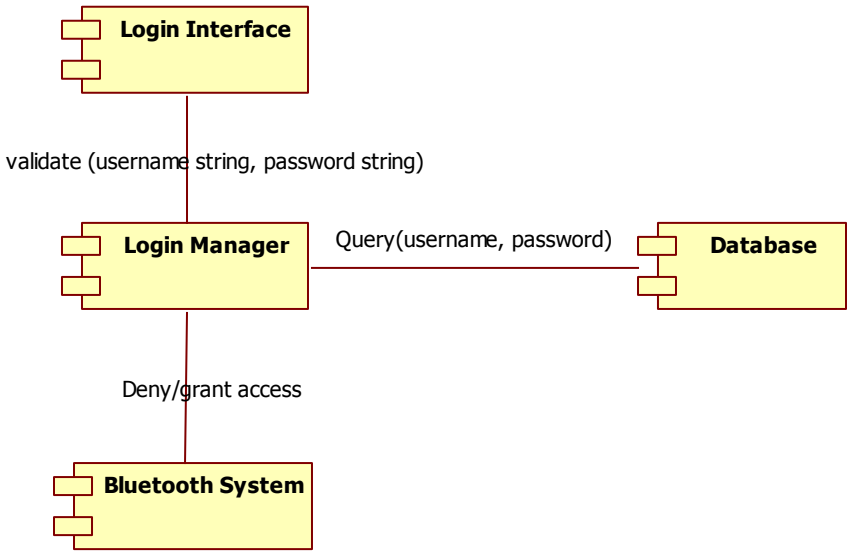

Figure 15: Implementation Diagram of the login model.

5.2.1 Log in interface

The user interface program is able to interface with login manager module. It will provide user friendly and easy to use graphical interface, and enable the use of commands easily.

5.2.2 Login manager

The login manager is responsible for checking if the username and passwords are correct or incorrect and the granting or denying access respectively to the Bluetooth piconet.

5.2.3 The database

The database will store the username and the password set by the user any changes must be effected by the login manager through queries.

5.3 Testing and Evaluation

To ascertain whether the login system for the user authentication to Bluetooth piconet using password and username was working properly, the system was tested using some few usernames and passwords. The system tested to be working properly.

5.3.0 Incremental testing

Incremental testing is where codes of a software system are tested as they are being written. Incremental testing was used in the development of this model and activities involved, testing the stability of the code, saving a copy of the stable code and falling back to the stable code should something go wrong with the code later. This form of testing helps the programmer to quickly identify problems in the code, debug them, and establish stable builds of the code hence reducing the time spent on debugging.

5.3.1 Module testing

Module testing was used where each module was tested while being coded or after it had been completely implemented, this was in order to establish whether it was providing its intended functionality with no errors. Due to the modular design approach taken in this thesis, each module was tested independently, hence easily isolating problems in the system.
Sequences of tests done during module testing are outlined as bellow.

- Login Interface: To test if the login interface module was working properly
- Login Manager Module: To test whether the login manager was performing its functions correctly.
- Database module: To test whether the database was working correctly and executing queries as required.

5.3.3 Integration testing

In Integration testing the modules were combined and tested as a whole. It was highly possible that integration testing of certain modules could fail, though each module passed the module testing. An incremental approach was used during the integration testing. That is, modules are added to the system in an incremental fashion, so that the modules causing problems could be detected, in case one should occur. For example, the system was tested by adding the login interface and login manager modules to the system first. After all was working fine, the database module was added. It was not possible to integrate the login system with the existing Bluetooth system in the device that was used in testing hence it was not possible to protect the existing Bluetooth system form unauthenticated users.

5.3.4 Final testing

Final testing involved testing the entire software system as a whole. This was mainly done to iron out any last minute bugs that could be discovered, system performance and usability were also tested. More effort was put on debugging the programs, recompilations and errors correction for the system to work properly.

5.3.5 Deployment

After testing stage the final stage was deployment of the complete login system directly to the mobile device. There were two ways to do this that were considered. The first was to deploy the login system via the internet and the second one was to deploy via a network connection between the development computer and the handset. This was either via a USB cable or a Bluetooth wireless connection. Most Java-enabled devices will allow you to install J2ME applications via this connection. Since the computer used in development was Bluetooth enabled, the login system was transferred to the mobile device through Bluetooth wireless connection. The application was installed and worked as required on Nokia Asha 200 phone.

CHAPTER SIX: FINDINGS AND CONCLUSIONS

6.1 Discussion of Findings

Bluetooth is a new and interesting technology, a technology that has no boundaries when it comes to data communication. Bluetooth can overcome the odds of language barriers with help of translator programs in situations where people of different languages want to exchange data or conduct business with each other. A Bluetooth piconet can experience many security threats if the underlying Bluetooth protocol is not well implemented. Different manufacturers implement the protocol differently. The user authentication model designed in this thesis can protect the Bluetooth piconet from unauthorized users; hence ensuring that the device owner's data is protected before and after a device was lost or stolen. This model can also ensure that the owner's other devices that the stolen device had paired with before are protected. The login model will ensure that Bluetooth is not open to anybody who holds the device as before since each user must be authenticated first in order to use it, be it a genuine user or a malicious user. The Bluetooth Special Interest Group needs to ensure that Bluetooth is a secure protocol. There is not much literature concerning Bluetooth user authentication as a topic. Most literature addresses Bluetooth security as whole topic

6.2 Conclusion

The project met its objectives as the user authentication model was developed, tested and was found to be working well. This model is intended to protect the Bluetooth piconet from any user who is not authorized. The model was designed and implemented using J2ME language for demonstration purposes. The model design is object oriented meaning that it is possible to implement the same model using other object oriented languages. It was not possible for the password and username to be encrypted since encryption was not a priority in this project, however due to the modular approach taken in this project it's possible to integrate an encryption algorithm into the model.

6.3 Organization of the Study

The study is organized into six chapters:

CHAPTER ONE: Is an introduction chapter, showing introduction to background of the problem, problem statement and purpose and scope of the study, objectives, research questions and justification.

CHAPTER TWO: Describes what other researchers have done in relation to the current project. It also shows how objectives one, two and three were met. This chapter gives an overview of the Bluetooth technology and different aspects of the technology are discussed, starting with a general overview of the Bluetooth architecture. Important concepts such as Bluetooth piconets, Bluetooth services, Bluetooth profiles, page scan process and the inquiry process is discussed. Chapter 2 also looks into the Bluetooth security model, weaknesses, attacks and proposed mitigation mechanisms.

CHAPTER THREE: This chapter shows the research methodology solution techniques used in developing the user authentication model. Here J2me language is discussed and the requirements analysis of the application.

CHAPTER FOUR: This chapter deals with the design of the user authentication model revealing how the model will look like and its functioning requirements.

CHAPTER FIVE: This chapter concerns the implementation of the user authentication model and the tests that were carried out to test the model.

CHAPTER SIX: This very last chapter deals with the discussion of the findings, the conclusion, critical review and the recommendations for further work and improvement.
APPENDICES: This will be highlighting the tools used with all necessary details on data collection for this research project.
REFERENCES: This will be indicating the sources of information used in this thesis.

6.4 Further Work

During the work with this thesis an investigation has been done in an effort to improve the Bluetooth piconet security. Implementing Bluetooth security at the application level is a bit complex since the developer is not fully in control of the underlying core protocols and considering the fact that knowledge of Bluetooth programming is a bit scarce. Advancing the new model to incorporate biometrics rather than mere password and username will be an interesting venture, especially with the availability of more powerful devices such as Smartphone's and ipads. The model can also be improved in order to authenticate Users in both ends using biometrics. Looking into how the Bluetooth security manager is implemented and how it can be improved, can also be an exciting subject for researchers.

REFERENCES.

Ahmed, J., 2009. The Device Discovery in Bluetooth Scatternet Formation Algorithms., 2009.

Ajay Jangra, K.K.B., 2010. IEEE WLANs Standards for Mobile Ad-hoc Networks (MANETs): Performance Analysis. *Global Journal of Computer Science and Technology*, Vol. 10(Issue 14 (Ver. 1.0)), pp.42-43.

Ajay, S., 2008. Bluetooth Security Issues, Threats And Consequences. In *Proceedings of 2nd National Conference on Challenges & Opportunities in Information Technology (COIT-2008)*., 2008.

Aroackiasamy, M.L.a.S., 2010. Analysis of Malicious Detection in Bluetooth Enabled Devices Exploiting Wireless Personal Area Networks. *Global Journal of Computer Science and Technology*, 10(1), p.11.

B. Gupta, C.J.M.M., 2009. Defending against Distributed Denial of Service Attacks: Issues and Challenges. *Information Security Journal*, vol. 18(issue 5), pp.223-48.

Bluetooth, S., 2006. *Bluetooth*. [Online] Available at: http://www.bluetooth.com [Accessed April 2012].

Christian Gehrmann, K.N., 2002. Enhancements to Bluetooth Baseband Security. *Ericsson Mobile Communcations AB, Ericsson Research and Nokia Research Center,* , pp.1-7.

Colleen, R., 2006. Bluetooth Security. *East Carolina University*.

Dahlberg, A., 2002. Traffic Engineering in a Bluetooth Piconet.

Dahlberg, A., 2002. *Traffic Engineering in a Bluetooth Piconet*. MSC Thesis. Australian Telecommunications Research Institute.

Frazer Bennett, D.C.J.B.E.A., 2010. Piconet Embedded Mobile Networking. pp.12-13.

Gelzayd, Y., 2002. An alternate connection establishment scheme in the bluetooth system.

Haataja, K.M.J., 2006. Security in Bluetooth, WLAN and IrDA: A comparison.. pp.5-6.

Haataja, K., 2009. *Security Threats and Countermeasures in Bluetooth-Enabled Systems*. University of Kuopio.

Hasbiha Hossain, U.K.a.S.R., 2011. Modified Approach of RFCOMM Implementation to Protect Bluetooth Technology from Bluebug Attack. *IJCIT*, OLUME 01(SSUE 02), pp.22-23.

Holmburg, M.C.a.E., 2004. File Transfer using Bluetooth.

Hypponen., K.H.a.K., 2008. Man-In-The-Middle attacks on Bluetooth: A Comparative Analysis , A Novel Attack and Countermeasures.. *ISCCSP, March 2008.* , p.pages 1096–1102.

Isaksson, L., 2004. Improved Performance of Bluetooth with Focus on Ad-Hoc Applications. pp.23-36.

John Padgette, K.S., 2008. Guide to Bluetooth Security. *the National Institute of Standards and Technology* , Special Publication 800-121 , pp.21-43.

Karen Scarfone, J.P., 2008. Guide to Bluetooth Security. *National Institute of Standard and Technology, US Department of Commerce*, Special Publication 800-121, pp.13-14.

Karl E Persson, D.M., 2003. Secure Connections in Bluetooth Scatternets. In *Hawaii International Conference on System Sciences*., 2003.

Khan, M.A.A.a.M.I., 2010. Security Enhancement of Pairing and Authentication Process of bluetooth. *International Journal of Computer Science and Network Security,* VOL.10(No.6).

Kotadia, M., 2004. Bluesnarfing tools spreading quickly. *ZDNet UK*.

Lewis, J., 2005. Bluetooth Security.

Mian Muhammad Waseem Iqbal, F.K.a.M.A.W., 2010. Attacks on Bluetooth Security Architecture and Its Countermeasures. pp.195-97.

Mihai Doinea, M.Z., 2010. Bluespam Filtering. *Economy Informatics* , vol. 10(1), pp.34-35.

Mohammed Tarique, 2011. A Secured Bluetooth Based Social Network. *International Journal of Computer Applications*, Volume 26(No.1), pp.16-19.

Niem, T.C., 2003. Bluetooth And Its Inherent Security Issues. *SANS Institute*, pp.5-6.

Nishant Mishra, V.G., 2012. An overview of Bluetooth Security, Issues and Challeges. *Journal of Global Research in Computer Science* , 3(3), pp.73-77.

P S Patheja, A.A.W.S.N., 2011. A Hybrid Encryption Technique to Secure Bluetooth Communication. *International Journal of Computer Applications*, pp.24-27.

Paraskevas Kitsos, N.S.K.P.a.O.K., 2003. Hardware Implementation of Bluetooth Security. *IEEE CS and IEEE* , pp.21-27.

Patheja P.S, A.A.W.S.N., 2012. Current trends and research issues in Bluetooth communication. *IJREAS*, Volume 2(Issue 2), pp.3-4.

Patheja, P.S.A.A.W..S.N., 2011. A Hybrid Encryption Technique to Secure Bluetooth Communication. In *International Conference on Computer Communication and Networks CSI-COMNET-2011*., 2011.

Preetha, G., 2010. A Novel solution to the short range Bluetooth communication. *Department of Information Technology*.

Saroj Kumar Panigrahy, S.K.J.a.A.K.T., 2011. Security in Bluetooth, RFID and Wireless Sensor Networks. *ICCCS*, pp.628-32.

Sharma, A., 2008. Bluetooth Security Issues, Threats And Consequences. *National Conference on Challenges & Opportunities in Information Technology*, pp.78-80.

T. J. Thompson, P.J.K.C.B.K., 2008. Bluetooth Application Programming with the Java APIs Essentials Edition. pp.8-10.

Welke, K., 2006. Building a secure, mobile VoIP network over Bluetooth.

Yelena, G., 2002. AN ALTERNATE CONNECTION ESTABLISHMENT SCHEME IN THE BLUETOOTH SYSTEM. pp.14-30.

Yen, A.B.T., 2011. *Mobile Phone Controlled Robot T-I*. Universiti Tunku Abdul Rahman.