Bluetooth Pairing Vulnerabilities & Suggestive Measures

By

Fahad Haleem

A THESIS SUBMITTED TO THE FACULTY OF COMPUTER SCIENCE DEPARTMENT, MILITARY COLLEGE OF SIGNALS, NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY, ISLAMABAD, PAKISTAN, IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MS IN COMPUTER SCIENCE (SOFTWARE) ENGINEERING

SEPTEMBER 2019

بِسْمِ اللهِ الرَّحْمٰنِ الرَّحِيْمِ

In the name of Allah (SWT) the most beneficent, the most merciful

DEDICATED TO

1. My Parents who always support me and have faith in me.
2. My Spouse Nazia, who instead of her busy official military life, supported me during my entire MS morally and intellectually.
3. My Teachers, specially Mam Mehreen Afzal, who taught me everything during the course of MS.

Abstract

The most widely and cheap mode of transmission over short distances is by using Bluetooth device. Bluetooth is most common mode to transmission between mobile phones. To make data secure for both sides of transmission, it uses key exchange algorithm named Elliptic Curve Diffie-Hellman algorithm for short term and long term key development. During the design phase of Bluetooth device, its security was not focused, hence it was vulnerable from hacker's point of view. In the era where information is the most powerful tool, any eavesdropping may lead to the bigger threat. Bluetooth devices lack key feature authentication and is thus vulnerable to Main in the Middle (MITM) attacks. In this dissertation, we analyzed the vulnerability present in Bluetooth v 4.0 and above from both exploitation and encounter methods. The new variant of invalid curve attack that preserve only x-coordinate of public keys is recently presented, this new attack is successful against all present Bluetooth paring devices protocols. Thus both of the pairing devices are vulnerable to this attack. This thesis also includes the suggestive measures against this vulnerability that can be considered as key exchange features for the development of upcoming Bluetooth v 5.0.

Declaration

I hereby declare that this work, neither as a whole nor as a part there of has been copied out from any source. No portion of the work presented in this report has been submitted in support of any application for any other degree or qualification of this or any other university or institute of learning.

Fahad Haleem

Reg # 00000240993

Acknowledgement

First of all I bow my head to ALLAH, Who has bestowed me with His blessings.

It is though boundless and infinite mercy of Allah and His Prophet Muhammad (PBUH), that I have been able to complete this Thesis.

I would like to express deep gratitude to my dedicated supervisor, Mam Mehreen Afzal for guiding this research with clarity and that priceless gift of getting things done by sharing his valuable ideas as well as his knowledge. He is a great teacher and source of inspiration for me. I am very gratefully to my parents who guided me and remain keen for my health and success.

Contents

List of Figures

List of Tables

1. Introduction:

Mobile phone have proved themselves to be the greatest gift to the mankind. They play very important and inseparable part in human's everyday life. The most common feature that almost all modern devices used is Bluetooth. Bluetooth is a wireless technology that allows data exchange between mobile devices over fixed range. The Bluetooth cover small distance using short wavelength UHF radio waves, that ranges from 2.400 to 2.485 GHz including guard bands 3 MHz wide at bottom end and 3.5MHz at top. This was included in the IEEE standardized Bluetooth IEEE 802.15.1 [1]. Bluetooth consists of 79 channels, and transmit data in the form of packets through these channels using frequency-hopping spread spectrum. The bandwidth of each channel limits to 1 MHz and it performs almost 1600 hops per second. Bluetooth is considered as low energy device that uses 2 MHz spacing.

Bluetooth is the master/slave based technique that follows packet based protocol. One master may communicate with up to seven slaves, although not all versions of Bluetooth have this capability. The master/slave devices can switch roles by agreement, master can become slave and slave can become master [2]. During the transmission between both devices, data transfer is only possible between two devices. The choice of slave is always with master and slave just like real life slave listen to the every slot received.

Bluetooth is considered as reasonably secure protocol when used with precautions, proper encryption between devices prevent casual eavesdropping from other devices. Although Bluetooth provide encryption, but they often shift radio frequency while paired caused opening for attack. Bluetooth device provide setting to limit its connections, the device level security allows protection to device by limiting connection to particular device, while service level security provides all kind of activities that your device can permits. Like all other devices and methods, there is always some flaw that involved security risk. With advance tools available, hackers have devised a variety of malicious attacks to involve Bluetooth security [3]. To give an insight, there are plenty of attacks that can cause security damage, like 'blue bugging' is term used when attacker take control of your mobile phone and all its functions. 'Blue snarfing' is

term used when hacker gain authorized access to your device using Bluetooth [4]. Although common people are not affected by these attacks when used with precautions, but whenever there is vulnerability there is always risk of threats.

1.1. Problem Statement

From the invention of Bluetooth devices, there are number of flaws that were present in the structure of Bluetooth. These flaws were used by hackers to damage security, or data theft. Recently Bluetooth version 4.2 was introduced which have vulnerability in the key exchange algorithm, this key exchange algorithm is based on Epileptic Curve Diffie-Hellman (ECDH) algorithm. This vulnerability was present from the start of version 4.2 as its algorithm doesn't validate each points present on the curve according to ECDH, thus attacker can exploit this vulnerability.

In this research work, we will consider methods for the exploitation of this vulnerability by studying methods to retrieve short term key or long term key that can be used for interception of message for MITM. The success rate and failure rate will be determined as the end product of this research using various methods.

1.2. Objective of research

The main objective of this research is to study methods for exploitation of ECDH vulnerability form attacker point of view. Keeping in mind these exploitation we will give suggestive measures against these exploitations that can help Bluetooth manufacturer's in overcoming this limitation for upcoming Bluetooth version 5.2.

1.3. Significance of the research

Bluetooth is one of the most commonly used low power short range wireless device that has applications in almost every field. The vulnerability present in current version of Bluetooth can be exploited by attacker to breach security of your device. Personal or secure data is not anymore secure in the presence of vulnerability so this must be removed in future Bluetooth versions to make this protocol safer. Thus more optimal suggestive measure must be needed to cope with this problem.

Chapter 2

2. Bluetooth Architecture

2.1. Specifications and features

Special Interest Group (SIG) defines all the features and specifications for Bluetooth device. This was introduced on 20 of May 1998. Downward capability is adopted for each and every Bluetooth device that allows new versions to cover standards of all older versions. Bluetooth have several different versions from the start of this technology that are defined in the form of versions [5]. Each new version consist of previously introduced devices, with few new set of capabilities.

2.1.1. Version 1.0 and 1.0B

The first version of Bluetooth technology were version 1.0 and 1.0B, they were not that successful due to many limitation present in the core specification of these versions. Due to new introduction of Bluetooth device, manufacturers faced numerous problems regarding operation of their device [2]. Mandatory Bluetooth hardware was needed for this service to be used, that was difficult task keeping in mind the production process of devices. Like other wireless devices, it also consist of transmitter and receiver, for the transmission Bluetooth Hardware Device Address (BA_ADDR) was introduced in connecting phase, that made anonymity very difficult in protocol level [6]v. This was the main setback for the manufacturers in introducing many features that was not now possible due to this limitation.

2.1.2. Bluetooth version 1.1

To improve the previous model, version 1.1 was introduced. Bluetooth version 1.1 improved many errors present in its predecessors. New technology introduced new set of features, RSSI measurement was included on receiving end to monitor the power of received signal [7]. RSSI introduction also produced faster connection, adaptive frequency hopping, faster discovery, and fast transmission speed. Among other features, the main feature the included was possibility of non-encrypted channels.

2.1.3. Bluetooth version 1.2

Every new version was introduced to add more features, or to overcome limitations present in old versions. Bluetooth v1.2 ratified IEEE Standard 802.15.1–2005 and included adaptive frequency hopping spread spectrum that improved the radio resistance by hopping crowded frequencies in consecutive manner [8]. Transmission rate was also improve from v 1.1 to 721 Kbits/s. Voice quality of audio links was improved by providing Extended Synchronous Connections (eSCO). This increased audio latency for concurrent data transfer. The most important feature was flow control and retransmission modes for L2CAP.

2.1.4. Bluetooth version 2.0

Version 2.0 was released in 2004, Enhanced Data Rate (EDR) was the key new feature introduced for faster data transfer at the rate of 3Mbits/s that was much more than previous rate of 2.1Mbits/s. This data rate was determined for inter-packet time and acknowledgment [9]. EDR combines Phase Shift keying (PSK) and GFSK with variants of 8-DPSK and $\pi/4$-DQPSK. This result in low power consumption due to the less duty cycle. This new EDR technology was not necessary of every device, SIG specification only mention EDR as optional feature [10].

2.1.5. Bluetooth version 2.1

SIG adopted Bluetooth v2.1 in 2007 with the most prominent feature called Secure Simple Pairing (SSP). SSP increased security for Bluetooth devices during pairing giving protection in Man in the Middle (MITM) phase using Diffie-Hellman algorithm [11]. Among other new feature, Extended Inquiry Response (EIT) and sniff sub rating were also introduced which produced filtering of device before connection and less power consumption respectively.

2.1.6. Bluetooth version 3.0

SIG adopted Bluetooth version 3.0 in 2009 with successive enhanced L2CAP mode and Alternative MAC/PHY (AMP) feature. Data transfer speed was increased to 24Mbits/s to increase the throughput of Bluetooth that was still less than other protocols. By introducing AMP, Bluetooth devices were allowed to use 802.11 link which was created by the use of alternating channel [12]. Establishment and negotiating was still present like previous versions,

802.11 used different protocols in physical layer and link layer. This result in high transmission rate and increase in range. L2CAP enhanced mode was implemented using reliable Enhanced Retransmission Mode (ERTM). Former Streaming Mode (SM) uses unreliable channel with no flow control and retransmission [13]. Alternative MAC/PHY made it possible to transport profile data using alternative MAC and PHYs. For small data, it uses radio along with device discovery, profile configuration and initial connection. For large transportation of data it uses MAC/PHY 802.11 that is associated with Wi-Fi. This new innovation indicate that during idle state low power connection is established, while for larger quantities faster radio is used. AMP links are required for enhanced L2CAP modes [14]. Version 3.0 also introduced the concept of power control that enable closed loop power control and go straight to maximum power options.

2.1.7. Bluetooth version 4.0

Bluetooth version 4.0 was adopted by SIG form 2010 onwards, it included ultra-low power protocol named Bluetooth Low Energy (BTLE). BTLE consider modulation mode for link layer packet formation for low power embedded devices. With introduction of BTLE in Bluetooth, various consumer devices adopted this version included mobile phones and wearable technologies. Due to high speed and low energy protocols, it was called Bluetooth smart that consist of innovation in classic Bluetooth protocols [15]. Unlike former versions, it aims for very low power application powered by coin cell maintaining same transmission range. Manufacturers designed single and dual mode for implementation for low power implementation by changing the IC design [16]. This change in design effected in the form of less cost highly integrated compact devices that were light weight and provided ultra-low power idle mode operation with secure encrypted connection.

2.1.8. Bluetooth version 4.1

Specification of version 4.1 were adopted by SIG from 2013 that included only incremental software update for Bluetooth v4.0. This update was intentioned to provide better consumer usability by providing multitasking, large data exchange, and support for LTE. Some of the introduced specification were already incorporated in version 4.0, but many new specifications were added in version 4.1. BTLE PHY was improved to provide RF spectrum of 40 channels

which consist of 2MHz width from 2402 MHz to 2482 MHz Only these 40 channels were labeled for discovering and pairing packets, while rest of the channels were labeled as data channels for connection establishment and transmission [17]. In Bluetooth v4.1, link layer was also redesigned for new pairing protocol.

2.1.9. Bluetooth version 4.2

SIG introduced core specification for Bluetooth version 4.2 in 2014 with improved specifications for BTLE to make in main protocol for Internet of Things (IoT) [18]. These specifications included latest LE secure connection mode for low energy link layer privacy and data packet security enhancement. Bluetooth v4.2 was designed in a manner that new version may adopt it specification using firmware updates [19].

2.1.10. Bluetooth version 5.0

In December 2016, SIG introduced specifications for Bluetooth version 5.0. This version improved several features in physical layer including high throughput, extended range, higher advertisement capacity and low energy. Version 5.0 mainly focused on emerging technology of IoT, latest mobile phones including Galaxy-S8 and Apple Home Pod were released with Bluetooth version 5.0. In version 5.0 range was compromised for speed of 2Mb/s for burst by increasing the packet length.

2.1.11. Bluetooth version 5.1

Latest version of Bluetooth was presented by SIG in January 2019. In this version tracking and location of devices were improved by using angle of departure and angle of arrival. HCI support was used for LE secure configuration. The improvement from former versions were done in mesh based model hierarchy.

The most important features used in latest versions are SSP and LE secure connections that are used for BR/EDR and Bluetooth low energy. These are only secure pairing protocols to date.

In order to understand in-depth view of Bluetooth for vulnerability exploitation point of view, the architecture must be focused in detail. The complete functionality of Bluetooth is observed by understanding its architecture that include both software and hardware. Following section

include detail analysis of Bluetooth architecture. The architecture of Bluetooth is the heart of its low power and less costly design. Bluetooth is radio system that depend on both hardware and software. The software defines the connection between different layers of it architecture, the stacks defined by software are the center point of this whole network. How Bluetooth works is defined in protocol stack that is set of layered program. Each layer in protocol stack only communicate with its consecutive above layer, and to the layer above it. The architecture of Bluetooth is explained diagrammatically in Figure 1.

Bluetooth consist of upper and lower layer stacks that defined the linkage between different modules.

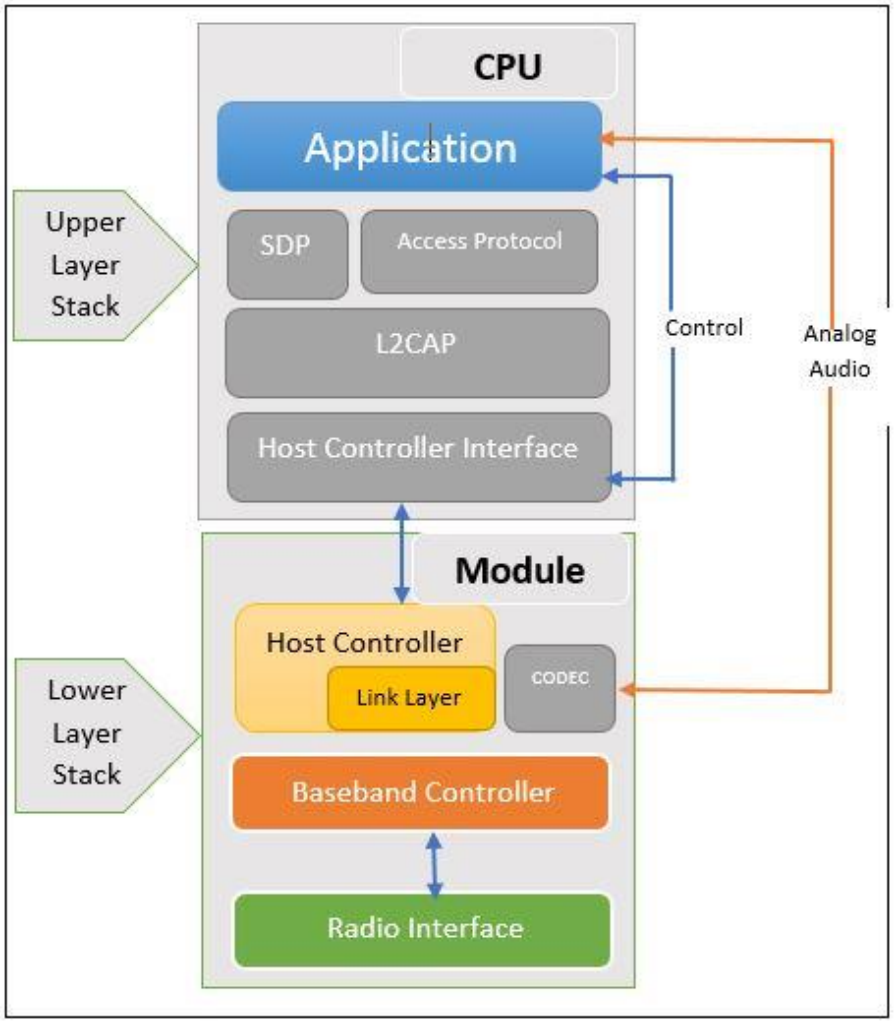

Figure 1: Bluetooth Architecture

2.2. Lower Layer stack

The lower layer stack consist of Bluetooth functionality defined by core specifications. As shown in Figure. 1, the base of lower stack layer is radio interface which characterize the physical behavior of transceiver. Bluetooth uses 2.4 GHz band, over this band radio module is responsible for receiving or transmitting of data through modulation/demodulation using physical wireless connection. For the security of this layer, transmission band are splited into 79 channels to perform fast frequency hopping. Directly above radio layer is baseband and link controller/link manager protocol (LMP). Baseband forms a linkage for transmission from or to the radio layer based on packets, timing, frames and flow control that is defined in baseband layer. LMP enforces fairness among the slaves by providing power management and connection management.

2.3. Upper layer stack

Upper stack layers uses core technology to specify the profiles on which communication between devices is done. The lower most layer in upper stack layer is human controller interface (HCI) that is mode of communication between software and hardware parts. Above HCI is logical link control and adaption protocol (L2CAP) that checks the flow of data packets. L2CAP is the must layer for every Bluetooth device. After L2CAP layer, the order variated for different devices. The practical implementation of these layers is Bluetooth headsets that only uses module and host portions of the stack due to need of small size. HCI is only used when there is need of device testing.

2.4. Bluetooth protocol profiles

Bluetooth profile consist of instructions defined by protocol stack that is different for need of device, i.e. FAX machine uses FAX profile while mobile phone might uses headset profile. Protocols are set of instruction that define how to use device, how to achieve specific goals, and how to implement Bluetooth combining hardware and software. To be more specific, for wireless transmission Bluetooth device must interprets among many Bluetooth profiles certain profile depending upon the desired activity. Profiles are definitions of possible applications, and specify the behaviors that Bluetooth-enabled devices use to communicate with each

other. Non-technical definition of profile is instructions or settings to be followed for communication. To the very least, each profile in Bluetooth contains following information:

1. Other profile dependencies.
2. Formats for user interface.
3. Specific parts of the Bluetooth protocol stack used by the profile.

At stack level, each layer perform specific tasks and options for Bluetooth profile. To give better perspective, let take a look at this example of Bluetooth headset. In order to accomplish high quality audio streaming, Advanced Audio Distribution Profile) uses L2CAP and LMP protocols to capture the radio interface for desired signal transmission. Same process will be followed in reverse direction on the receiving end. Few of the many Bluetooth profiles are described below:

1. **A2DP** – Advanced Audio Distribution Profile
2. **AVRCP** – Audio/Video Remote Control Profile
3. **GAVDP** – General Audio/Video Distribution Profile
4. **PAN** – Personal Area Networking
5. **HFP** – Hands-Free Profile
6. **HSP** – Headset Profile
7. **CTP** – Cordless Telephony Profile
8. **VDP** – Video Distribution Profile
9. **FTP** – File Transfer Profile
10. **RFCOMM** – Radio Frequency Communications
11. **TCS** – Telephony Control Protocol
12. **WAP** – Wireless Application Protocol
13. **SDP** – Service Discovery Protocol

14. TCP/IP – Transmission Control Protocol/Internet Protocol

With the advancement in technology, the size and compatibility of Bluetooth devices are increasing with exponential rate. Till now Bluetooth is nearly most low power device for short range use, recently number of Bluetooth devices have been applied inside streamer or relay used in hearing aids to achieve Bluetooth head set like application that can be accessed through mobile phone.

The protocols provides link layer level access authentication and confidentiality, due to flexibility and enhanced embedded security, Bluetooth has become one of the most popular choice for communication in mobile phones. While Bluetooth devices have advantage of low power wireless communication, they also have disadvantage of data interception along with any other data sent on low-power radio waves. In this dissertation new set of cryptographic attack are studied using Epileptic Curve Diffie-Hellman (ECDH) protocol which effect all present Bluetooth devices, with sufficient amount of MITM authentication.

Chapter 3

3. Literature Review

In communication technologies information security is critical, and Bluetooth devices have no exemption. The increase in number of wireless technologies have also increased number of threats, Bluetooth specially have wide range of vulnerabilities that needed to be understand in order to take suggestive measures against it. Bluetooth device works on master slave principle, successive pairing is needed in order of start communication. However, pairing process of two devices is prone to variety of attacks that not only can harm Bluetooth device but also its user. A significant amount of research has already start to develop attacks for vulnerabilities and then to remove this vulnerability by taking suggestive measures [11-16]. However these studies have not yet yielded sufficient outcomes to appropriately address these security threats. The most common attacks are in the form of MITM attacks, new variant of invalid curve have still remain prominent in breaking the Bluetooth protocol and can be used as gateway for attackers.

This section reviews literature concerning MITM attacks and their countermeasures. Deep knowledge of Bluetooth pairing mechanism is prevailed by understanding the key specification and protocols. Researchers and scientists are constantly developing attacks to check the security of Bluetooth devices, for given output key of length O (264), [23] proved that Bluetooth stream cipher with 128-bit key can be broken in O(264) steps. Jakobsson and Wetzel [13] were the first one to devised proper attack on Bluetooth version 1.0B using MITM that was successful till version 2.0. This was mainly successful due to the absence of security specification protocols. The scenario in which this attack was devised contain sender, receiver and attacker in close proximity of circle and attacker knows the link key used between both. They also derived number of other vulnerabilities that were considered as milestone for future device manufacturing. Moreover, authors also demonstrated the process of obtaining link key using offline PIN crunching attack via passive eavesdropping on key establishment protocol [13]. The work done by authors [13] was considered as milestone and in [24], addressed this issue by introducing anonymity mode to prevent location tracking. Using same concept Kugler [25], improved the attack proposed by [13], using same channel hopping sequence for both

victims using different clock settings, in this manner victims will only see message sent by attackers not each other's. He also suggested avoiding unit keys as they are rarely changed due to storage in non-variable memory. To address the short coming presented in [13, 25 and 27], [28] proposed Diffie-Hellman key exchange that provide security using one way cryptographic function.

In [28], authors introduced the concept of using user friendly PINs ranging between 5 to 12 along with ECDH, however this was only successful against wiretapping and offline attacks, so SIG suggested in increase the number of PINs to stop dictionary attack from both passive and active approaches. In [29] reflection attack was introduced that can impersonate victim's device. This attack can be one sided or both sided impersonating both victims. Using only one victim's Bluetooth device address (BD_ADDRs), attacker can reflect the information received during authentication process. This is also one of main type of MITM attack against authentication rather than encryption. Same attack was suggested by [30] and for its counter measure use of BT_EC_SRP was recommended that can successfully create strong initialization key. Author in [31] also pointed out several Bluetooth vulnerabilities including susceptibility of key sharing for eavesdropping attack and short PINs. Due to absence of IP configuration in Bluetooth devices, validating device address is quite impossible, making it more vulnerable to spoof address. Furthermore, Bluetooth device also lacks end to end encryption, a limited encryption key length, a weak E0 stream cipher algorithm, weak pseudo random generator and weak mutual authentication. All this limitations make is most popular choice for attackers to invade user's devices. Proper measure must be taken against these vulnerabilities

Authors in [33] exploited short PIN by successfully devising paradigm in which attacker easily finds and decode PIN during pairing process. To address present flaws, [33] provided recommendations that Bluetooth must contain PIN code not less than 16-bits, encryption should be enabled by default, and the default security level of Bluetooth device should never be public. Author in [34] suggested use of bypass security model to involve peer authentication, low power key negotiation, and key generation. Providing statistical analysis, [34] suggested that adopting this methods results in cost effective and fast model that ECDH. In [35], author used flaw present in E0 resynchronization, to present fasted attack in encryption.

Haataja and Toi [17] derived couple of new attack on MITM authentication to effect I/O capabilities and mislead the victim by forcing to adopt less secure model rather that secure one. OOP authentication model was designed to avoid MITM attack, but using this approach attacker can falsify victim to avoid OOB model and adopt less secure JW authentication model. Latest Bluetooth devices have now capability to change device's address and others can actively discover hidden devices [20]. This threaten all four SSP models for MITM attacks. For the first time in [39], human authentication protocol was conducted, by demonstrating that concurrent paring attempts may lead to failed authentication. This was improved by providing simple pairing that preserve authentication model. The most common attacks that can be identified from literature are: eavesdropping attack, MITM attack, exploiting short PIN vulnerability, and exploiting weak link and encryption keys. Scientists and researchers have proposed solutions to these attacks, authors in [28] suggested used of Diffie-Hellman key exchange for better security, [34] suggested use of interlock protocols and elliptic curve cryptography to provide alternative power limited security model. Later, in [43], MITM attacks is addressed by proposing key agreement scheme during key encryption establishment. In [45], authors suggested use of 'au ID' for pairing and authentication.

Using 16-bit alphanumeric PINs and ECDH cryptography provides protection against eavesdropping by increasing security of BLE and Bluetooth v4.0. MITM attacks can be avoided by using user assisted numerical comparison and numeric method [47]. In [52], author presented SSP-Delayed Encryption Input Output (SSP-DEIO) protocol to divert the MITM attack in I/O exchange process. This results in better security and increase in time for pairing. Bluetooth version 5.0 is also considered vulnerable to MITM attack, for this authors in [53], proposed very easy counter measure of MITM attack in SSP pass key entry. Using this when user tries to reuse passkey loophole will appear, using this model passkey reused issue is improved. Table 1 presents the tabular form of each vulnerability, attack pattern, harms, and counter measures for different scenarios.

Chapter 4

4. Bluetooth security

After detail analysis of Bluetooth device from literature review, it is suggested that Bluetooth security require more improvements [13, 14, 17-18]. The overview of Bluetooth security protocols and its vulnerabilities are discussed in this section. For the case of this thesis, only Secure Simple Pairing (SSP) and LE Secure Connections (LE SC) pairing protocols are discussed in details.

4.1. Security of Bluetooth device

Security is not main feature of Bluetooth device, this is mainly due to less range of transmission that developers doesn't thought of Bluetooth security well enough. Whenever there is wireless mode of transmission there are some viable risks involved that cannot be avoided but through proper analysis and measurements they can be mitigated. As Bluetooth is sender receiver device so it is vulnerable to external hackers even in its short range (usually up to 10m). The security architecture of Bluetooth can be seen in Figure 2.

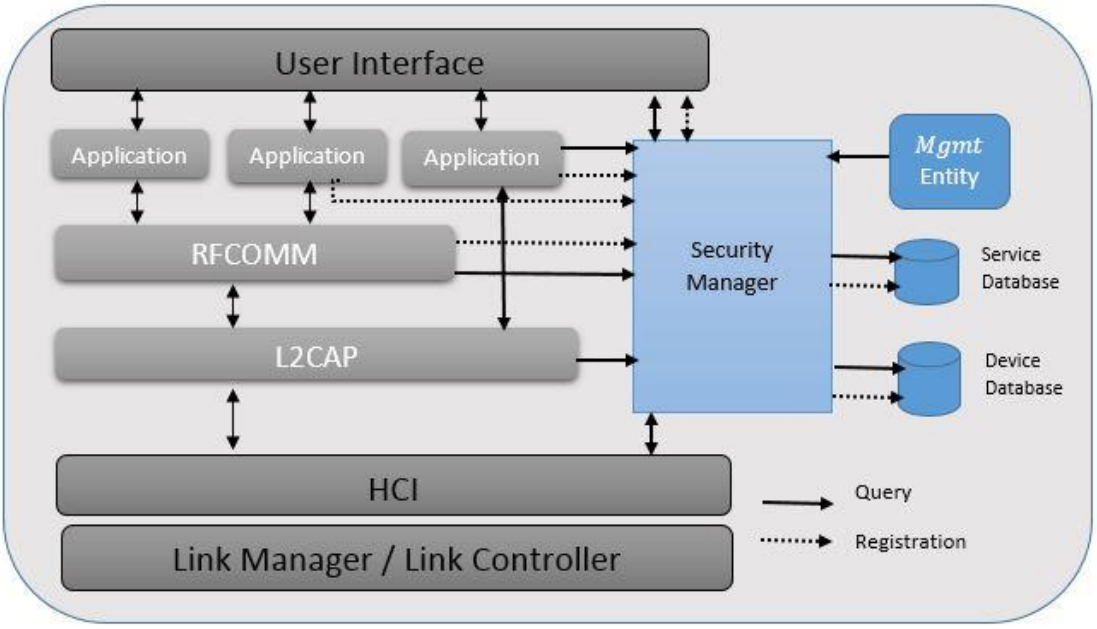

Figure 2: Bluetooth Security Architecture

The functionality of layer protocols and patterns are already discussed in section 1.3. Here we only discuss Bluetooth security protocols. Bluetooth is encryption and authentication type

security device, both pair of devices have secret key that is based on features extracted from encryption and authentication. This key is generated when both devices are connected for the first time and pairing is done. The security of Bluetooth device has two security levels,

1. **Authentication**: By pairing procedure, link is developed which is authenticated for which device is on other side of communication. The authentication procedure starts without user intervention depending on the level of service. Authentication is done only when the link is established between two devices through key pairing.

2. **Authorization**: Authorization consist of procedure for giving access to devices, already paired devices are connected without authorization while level of trust is determined for new/untrusted devices through authentication.

In order to give protection to user's communication, first layer of security is provided in Bluetooth technology itself. There are four different modes for connection that are chosen by user for connection.

1. **Low Energy**: This mode was designed for privacy, in this mode devices can communicate in connectionless mode.
2. **Silent**: This mode is only for traffic monitoring of Bluetooth and no connection is formed in this mode.
3. **Public**: This is public mode in which device is discoverable and accepts connections requests from other devices
4. **Private**: In this mode device is not discoverable and no connection is formed between new devices, communication can only done with already known addresses.

The level of security is determined by the paring key that is generated through encryption, key security is increased by level of designed security protocols. Secure Simple Pairing (SSP) is the standard pairing protocol that is followed in Bluetooth devices. Four different modes of security are adopted among each Bluetooth device, these security modes cannot be used in parallel. These are as follows:

1. **Mode 1**: This is the least secure mode because Bluetooth device allow pairing without any encryption or authentication. This mode was used in Bluetooth v2.1+EDR.
2. **Mode 2**: In this mode encryption and authentication process after pairing is achieved. This mode is implemented in all Bluetooth devices.
3. **Mode 3**: This mode is similar to mode 2 except security protocols were added on Link Management Protocol (LMP) layer. This enable security before physical connection.
4. **Mode 4**: This mode is only different from mode two in terms of extra service level security that offers SSP. SSP uses Elliptic Curve Diffie-Hellman (ECDH) base key exchange algorithm to create significant resistance in MITM attacks.

4.2. The elliptic Curve Diffie-Hellman protocol

The Elliptic Curve Diffie Hellman (ECDH) protocol is key exchange algorithm which was first introduced by Koblitz and Millter [13, 20]. ECDH exploits the elliptic curve for algebraic structure for finite field, this is done in order to exchange secure symmetric keys for public communication. Order q is the domain parameter of ECDH for the field $\mathbb{F}_q$ and base point P. The ECDH characteristic curve is based on equation

$$y^2 = x^3 + ax + b$$

The domain parameters $D = \{q, a, b, P, n\}$ must be same for both ends of communication for using ECDH. Both devices generate ECDH-pair key, which is the first step for key pair generation. This generated key-pair is combination of both public point PK and scalar points SK. To validate this statement, the public points are repeated addition of scalar points over points P such that,

$$PK = \{SK\}P$$

Using this scheme, both sender and receiver share their public points to each other and multiplying own scalar point with received public point private scalar is generated. To give better perspective of ECDH sharing key, *Figure 3* gives the outline of flow.

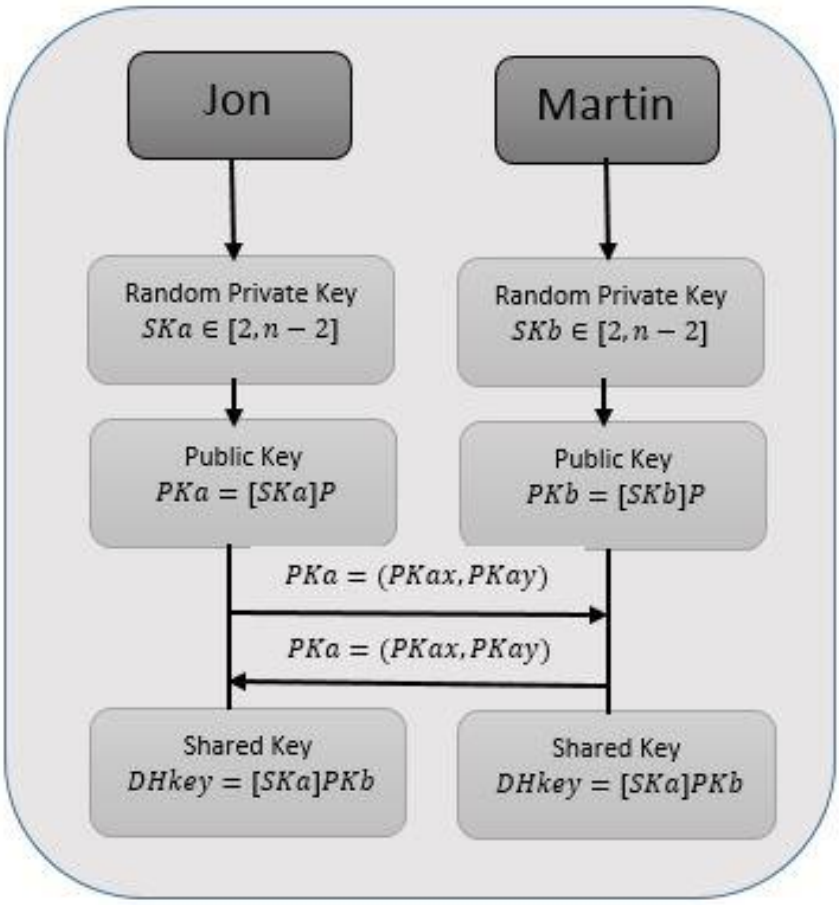

Figure 3: Flow diagram of ECDH protocol

In elliptic curves, the commutative resultant is formed by multiplication of two scalars. This steps is verification for the same key exchange between both ends. Mathematically,

$$[SKa]PKb = [SKa]([SKb]P = [SKb]([SKa]\ P)\ =\ [SKb]\ PKa$$

ECDH is derived from Diffie-Hellman key exchange algorithm [3], which uses multiplicative modulus groups, using elliptic curve based DH results in better efficiency. $O(\sqrt{n})$ group operations are needed for the best scenario of attack on discrete elliptic curve algorithm, which are far less than multiplicative modulus groups, hence ECDH is space efficient compared to previous DH algorithm. Due to this property of ECDH, it is most suited in embedded devices for low bandwidth secure communication.

4.3. Secure Simple Pairings

Secure Simple Pairing (SSP) is the standard pairing protocol for every Bluetooth device. This pairing protocols is not simple process, it is quite complex. From Bluetooth version 2.1+EDR, SSP was adopted instead of legacy pairing process. Unlike legacy pairing process which uses Personal Identification Number (PIN) for authentication, SSP offer authentication via visual confirmation of integer code [13-20]. This was not available in legacy process and hence through visual verification success rate against MITM attacks was increased. The SSP protocol is used to prevent intrusions during communication and provide security. ECDH is main part of SSP link layer key generation, which is formed using physical Bluetooth address and public-private key pair. MITM attacks are effectively stopped as combinations and permutations requires private key decryption which is resource-intensive. Depending on the input/output capability of Bluetooth device, SSP offers four type of models, these models are described as,

1. The Just-Works model: Automatic connection is generated as a result of no input and display capability of Bluetooth devices in association model, due to which this mode is more prone to MITM attacks.

2. The Out-of-Band Model: In this model cryptographic key are generated after finding visible Bluetooth devices within range. This mode is especially for devices that uses different wireless technologies i.e. Near Field Communication (NFC).

3. The Passkey Entry Model: This model was designed for Bluetooth devices that have either input capabilities in both or only one device have display capability. This model can be utilized using two scenarios, first is case when same six digit passkey is entered in both devices and second case is when passkey is copied from one device screen to other device keypad.

4. The Numeric Comparison Model: This model was designed for Bluetooth devices having both input and display capabilities. Using this model, passkey is shown on display of both devices and after confirmation pairing process is completed.

The security procedure of SSP require six types of phases to allow trusted pairing between two devices.

1. Exchange Capabilities: The dynamic pairing model is selected on the basis of results obtained from exchange of information between devices that have never been paired before.

2. Exchange of Public Key: Public/private key pairs are generated between communicating Bluetooth devices. After public key sharing between both devices, Diffie-Hellman key is computed.

3. Authentication Phase 1: First level authentication is done to avoid intrusion in the form of MITM attacks by providing checksum integrity. This authentication process depends on the model adopted by devices.

4. Authentication Phase 2: Second level of authentication involves exchange and verification of public keys between two Bluetooth devices.

5. Link Key Calculation: Secure link key is generated by Bluetooth devices that are communicating with each other, this link key is formed by public key, Bluetooth address and Diffie-Hellman key exchange.

6. LMP Authentication and Encryption: Encryption keys are generated by both Bluetooth devices.

It is previously discussed that advance mobile technologies are using Bluetooth for short range communication, but providing secure connection between two devices is still constant challenge due to presence of several security loopholes. These vulnerabilities create opportunities for intrusions such as MITM attacks. During MITM attack, the physical layer is jammed through malicious or random data overload, which effect piconets as shown in *Figure 4*.

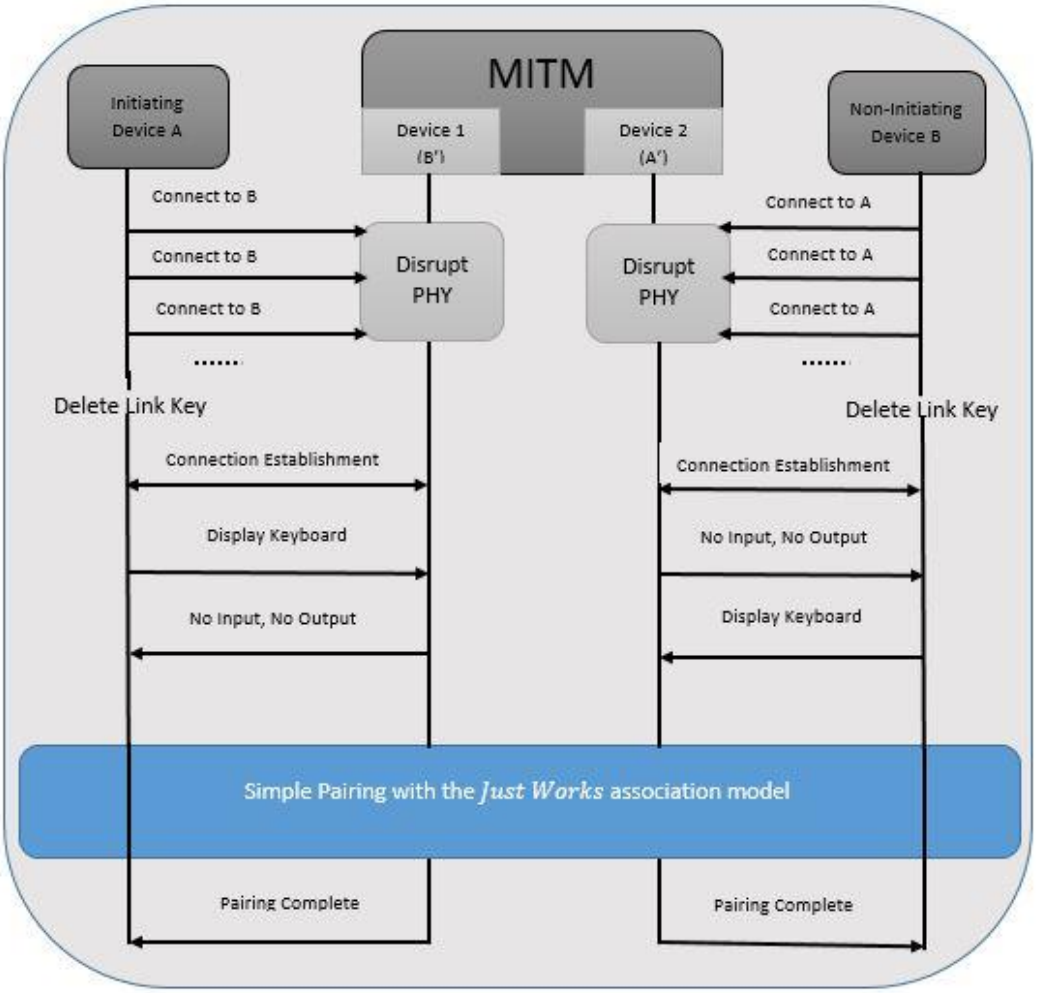

Figure 4: General Man in Middle Attack (MITM) architecture.

4.4. LE Secure Connections

Bluetooth have two paring schemes other than SSP, legacy pairing and LE secure connections (LE SC). Legacy pairing technique was used before development of version 4.2 and it didn't offer any security to eavesdropping. New LE SC was designed to mitigate eavesdropping like MITM attack by cryptography based on key exchange primitives and Message Authentication Code's (MACs).

LE SC have same models types like SSP and offer 'Just work', 'Out of band', 'Passkey entry', and Numeric Comparison' models. In comparison to SSP, LE SC functioning is just like SSP with minor differences, LE SC was motivated by SSP pairing protocol. There are mainly two differences between SSP and LE SC, first one is that LE SC uses AES-CMAC

based HMAC functions while SSP uses SHA-256 based functions. Second difference is that SSP support P-192 and P-256 curves for key while LE SC not uses these curves. Other than these, LE SC and SSP are almost same and follow same inputs and functions. To introduce new novel attack scheme, first better understanding of pairing schemes should be done. For the context of this thesis where new attack is introduced, we only explore phases of LE SC pairing protocol. First phase is feature exchange which is irrelevant so it is not discussed, Out of bound model is not discussed as it include technology for different wireless technologies. The Just-work mode is same as Numeric comparison mode except no authentication is required. To get a good grip on subject, different phases of pairing are discussed. Few important notations are discussed before analysis of different phases.

4.4.1. Definitions and Notations

Variables used in protocols.

1. $\boldsymbol{A}, \boldsymbol{B}$ – The BD_ADDR of each device consist of 6 bytes
2. $\boldsymbol{IOcapA}, \boldsymbol{IOcapB}$ – Input output capabilities of each device, this is usually exchanged during first phase. It consist of 1 Byte.
3. $\boldsymbol{PKa}, \boldsymbol{PKb}$ – Each device's public key, it consist of 64 Bytes.
4. $\boldsymbol{SKa}, \boldsymbol{SKb}$ – Each device's private key, it also consist of 64 Bytes
5. $\boldsymbol{PKax}, \boldsymbol{PKbx}$ – Each device's x coordinate for public key of 32 bytes
6. $\boldsymbol{DHkey}$ – Diffie-Hellman shared key of 32 bytes
7. $\boldsymbol{Na}, \boldsymbol{Nb}$ – Numeric comparison method using Nonces, 16 bytes
8. $\boldsymbol{Nai}, \boldsymbol{Nbi}$ –Passkey entry model using Nonces, 16 bytes
9. $\boldsymbol{rai}, \boldsymbol{rbi}$ –Single bit pass key
10. **Function $\boldsymbol{f4}$** –Commitment value generation function, defined as

$$f4(U, V, X, Y) = AES - CMAC_X(U \| V \| Y)$$

11. **Function $\boldsymbol{g2}$**–User confirm value generation function, defined as

$$g2(U, V, X, Y) = AES - CMAC_X(U \| V \| Y) (mod\ 2^{32})$$

12. **Function $\boldsymbol{f5}$**–Key derivation function, defined as,

$$T = AES - CMAC_{SALT}(DHKey)$$

$$f5(DHKey, N1, N2, A1, A2) = AES - CMAC_T(0||'btle'||N1||N2||A1||A2||256)||$$

$$AES - CMAC_T(1||'btle'||N1||N2||A1||A2||256)$$

Where salt is constant defined value

13. **Function $f6$**–Check value of generation function, defined as,

$$f6(W, N1, N2, R, IOcap, A1, A2) = AES - CMAC_W(N1||N2||R||IOcap||A1||A2||)$$

4.4.2. ECDH Key Exchange

Using standard NIST curve P-256, ECDH key pair is exchanged for domain parameters between both devices. The key exchange is followed by protocol described in section 4.2.

4.4.2.1. Authentication Phase 2

Authentication model in phase 2 is adopted using numeric comparison method with devices ability to display 6 digit decimal number so that at least one of them have option to deny or accept connection. The authentication procedure is followed as follows:

1. Nonce $Na, Nb,$ are selected by each party
2. Non-initiator commits (Cb) are assigned to public keys and Nb, such that,

$$Cb = f4(PKax, PKbx, Nb, 0)$$

3. Initiator followed by non-initiator reveal their nonce.
4. Commitment is decided by validation of nonce
5. Va and Vb are user defined values, whose six least significant digit are displayed by both sides using $g2(PKax, PKbx, Na, Nb)$.
6. After confirmation from the user, access or denial is made.

During this whole process, it is noted that y-coordinate is not validated, which can be exploited. Authentication using numeric comparison is shown in *Figure 5*.

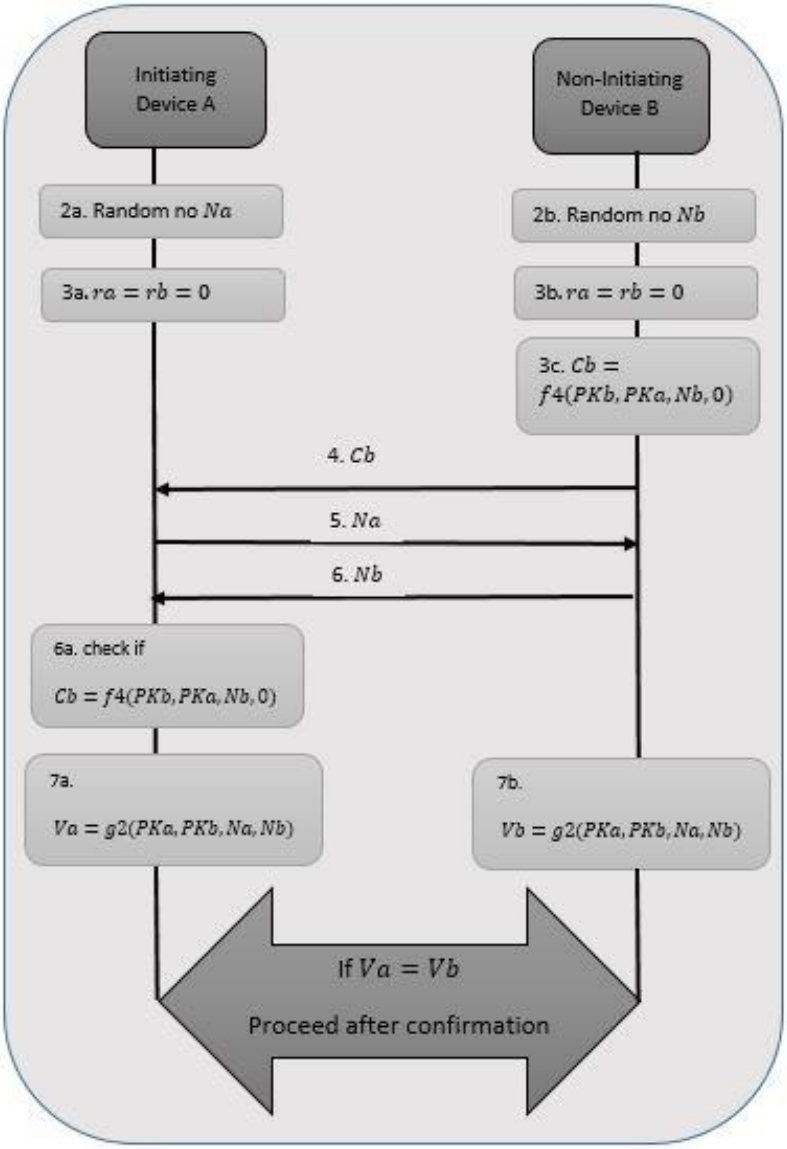

Figure 5: Numeric comparison Authentication

4.4.2.2. Authentication Phase 3

In phase 3 authentication model, passkey entry association model. This model is presented when only one of the available devices have numeric input capability and the other one have six digit display capability. This is described as follows,

1. Only one device generates passkey which has display, this is then pass to other device so that user can enter key.
2. Random nonce Na1 and Na2 are selected randomly by each device.

3. Using function $f4$, each device commits its public keys, nonce, first bit of passkey first for initiator $\left(f4(\ PKax, PKbx, Na1, ra1)\right)$ and then for non-initiator$(f4(\ PKbx, PKax, Nb1, rb1))$.
4. Following same procedure for communication, both devices reveal their nonce and validate.
5. For 20 bit passkey, step 2-4 is repeated 20 time.

Similar to previous case, this method does not validate y-coordinate. The block diagram of passkey method is shown in *Figure 6*.

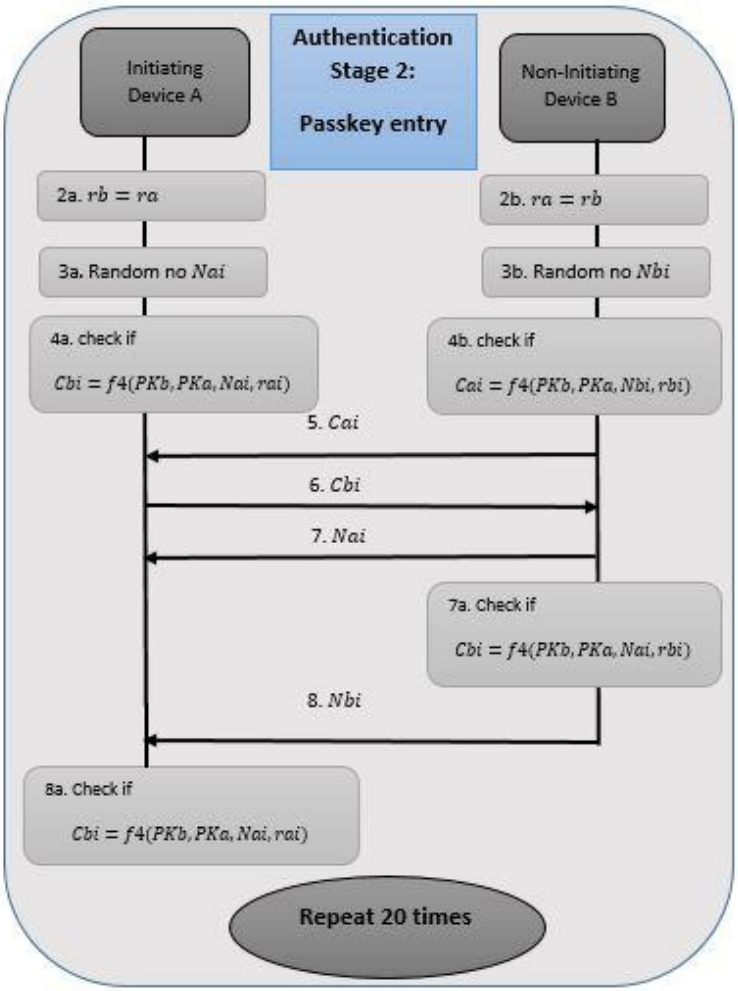

Figure 6: Passkey authentication method

In this section we have outlined both authentication methods, the comparison of both methods are shown in *Table 1*.

Table 1: Authentication comparison for function f6

Passkey authentication		Numeric Authentication
Ea	$f6(\ MacKey, Na, Nb, 0, IOcapA, A, B)$	$f6(\ MacKey, Na20, Nb20, rb, IOcapA, A, B)$
Eb	$f6(\ MacKey, Nb, Na, 0, IOcapB, B, A)$	$f6(\ MacKey, Nb20, Na20, ra, IOcapB, B, A)$

4.4.2.3. Authentication for session key generation

The final stage of authentication is responsible for the generation of session key.

1. Session key ($LTK\ and\ Mackey$) are derived from $DHkey$ using function $f5(\ DHKey, Na, Nb, A, B)$.
2. Checksum value for each device, Ea and Eb is computed using function $f6$, as shown in *Table 1*.
3. Checksum values is sent to non-initiator from initiator, which results is Check value.
4. Checksum value is validated by each side.

The block diagram of final phase is shown in *Figure 7*.

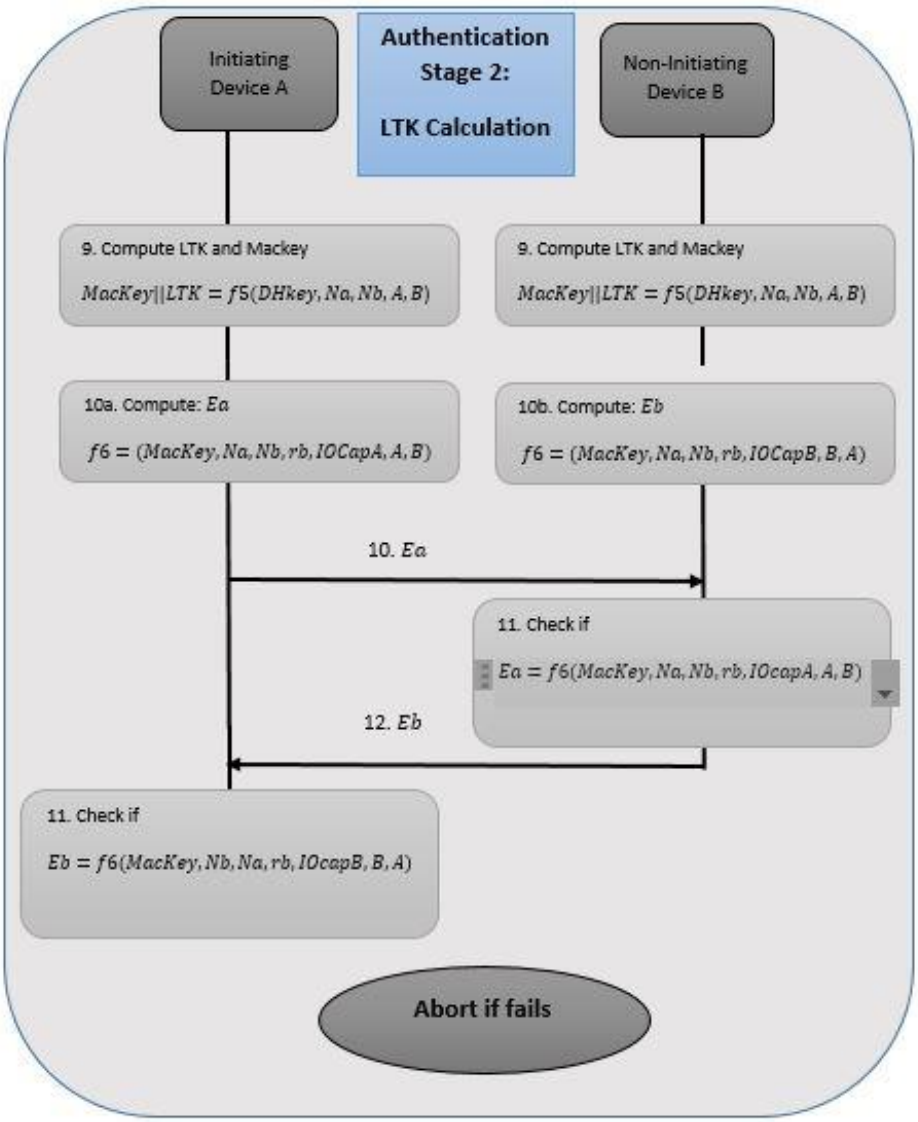

Figure 7: Session key authentication

Chapter 5

5. Attacks

This chapter provides comprehensive analysis of proposed attacks exploiting the vulnerabilities present in present Bluetooth versions. The solution is also discussed for proposed vulnerabilities, previous studies have focused on consistent and dangerous MITM attacks which have been most effective against all wireless mode of communication most importantly Bluetooth devices. In [19], author's explored the vulnerabilities present to exploit during paring process of SSP and possible encounters to avoid damage. The improved version of ESSP is also proposed in [19], two novel MITM attacks are also part of this paper. First attack was on the response of human cognitive capability, and second was on the database generated after successful pairing. Similarly countermeasures for these attacks were also proposed by making length of password maximum and introduction of new command to permit data access.

To reform the present security protocols of Bluetooth, [54] proposed extra channel for pairing to authenticate the process of pairing. This is different from SSP and concept of extra channel was introduced to mitigate the MITM attacks. Secure Device Pairing (SDP) was proposed by [54], that was four phase solution to current security problems. The concept of secure extra virtual channel for communication and authentication has proved to be more secure than conventional SSP.

The vulnerabilities present in JW model of Bluetooth SSP were discussed in [55], to countermeasure these vulnerabilities suggestive measures were also proposed. These proposed precautionary steps made device more secure for MITM attacks, the proposed method computes only after the validation of exchange data, this makes proposed method more secure. This extra level of protection mitigates the risks of MITM attacks.

Using steganography based approach, [56] proposed novel secure method against MITM attacks. Using cover object (steno-image), key is sent only to receiver. Receiver only receive key which he send back to sender for verification. After exchange shared key is generated and verified on both devices, to make it more secure, shared key must be used to view the message and check it originality. The whole process of verification is completely done internally without any external interaction.

A list of invalid curve attacks have been introduced by researchers over the years to exploit the vulnerabilities present in Bluetooth devices. The first invalid curve attack was introduced by Biehl et al. [2], and improved by Antipa et al. [1], to exploit the vulnerabilities present in the ECDH implementation. Invalid curve attacks in general belong to larger family of attacks namely, *small subgroup key recovery attacks.* This family of attacks extract non-ephemeral secret information by utilizing small subgroups of finite groups.

5.1. Background

In order to improve the quality of Bluetooth device, several methods are adopted. First attack is devised to address the vulnerabilities and then solution is provided for its countermeasure. To expose Bluetooth vulnerabilities for MITM attack, SSP protocol is heavily criticized. To address these vulnerabilities, SSP is reshaped using new starting point. To give security against MITM attack, fixed coordinate invalid curve attack is proposed which is new variant of invalid curve attack. Invalid curve attack are set of attacks that forge or preserve the x-coordinate of public key. Keeping in mind this vulnerability, new attacks are proposed that can be implemented on both SSP and LE SC. Using proposed attack method, attacker enforces a key to be confined to an unexpectedly small group. Although various methods are introduced to enhance the security of Bluetooth devices by bridging the gap between architecture and pairing processes, but still proper review of MITM attacks is not addressed till now. With the introduction of pairing in Bluetooth devices, Ericsson did not consider different level of security threats, therefore up till now there are still threats present in the architecture of Bluetooth. Keeping in mind the security and effectiveness aspect of Bluetooth devices, this thesis proposed different method to encounter MITM attacks.

Elliptic curve uses scalar multiplication by repeating operations, these repeating operations are point doubling and point addition. For domain $D = \{a, b, P, q, n\}$, the group operations are defined as,

14. Point doubling

Let point $P = (Px, Py) \in E$. Adding point to itself $R = P + P = [2]P$, where $Py \not\equiv 0$ is defined by drawing the tangent line of curve at point P. The sum of points R is

calculated by representing this point across the x-axis. Following computation is done for formula:

$$s \equiv (3Px^2 + a)(2Py)^{-1}$$

$$Rx \equiv s^2 - 2Px$$

$$Ry \equiv Py - s(Rx - Px)$$

The result is identity i.e. $\infty \in E$ if $Py \equiv 0$.

Curve parameter b is not involved in formulation.

15. Point Addition

Two points $P = (Px, Py)$, amd $Q = (Qx, Qy)$ such that $P \neq Q$. The point addition $R = P + Q$, where $P \neq -Q$ is obtained by drawing the line over the intersection of P and Q. Resultant point R is the reflection of points across x-axis. The computation is done in following manner:

$$s \equiv (Py \;\; Qy)(Px \;\; Qx)^{-1}$$

$$Rx \equiv s^2 - Px - Qx$$

$$Ry \equiv Py \;\; s(Rx - Px)$$

Result is identity $\infty \in E$ if $P = -Q$.

From these both formulations, it is clear that both methods does not involve curve parameters a and b in formulation.

5.2. First Attack (Private Key retrieval)

Using curve equation $y^2 = x^3 + ax + b'$, for different group E' such that point $Q_1 \in E'$. The scenario for attacker is created when attacker presents Q_1 as his ECDH public key and the private key SK of victim is calculated as $x = [SK]\, Q_1$ and send $H(x)$, where H is publically known one-way function. Due to the low order of Q_1, attacker can comparatively easily search value of x. The resultant discrete log $a1$, gives information of $x = [a_1]Q_1 = [SK]\, Q_1$. So $SK \equiv a_1$. Repeating same procedure, attacker generate key for different pattern Q_1, for different prime order p_i till the product of prime satisfies $\prod_{i=1}^{k} p_i > n$. In the end using Chinese Reminder theorem, attacker retrieve the victim's key.

5.3. Second Attack (MITM attack on ESSP)

Using ESSP instead of SSP by [61] security is thought to be increased be sharing public keys chosen by users. However this process is vulnerable to proposed MITM attack. The first phase of MITM attack exploited the weakness in password based system during pairing process, which is vulnerability present in this system, otherwise it is quite difficult to crack password based systems. This line of attack is designed by keeping in mind the cognitive behavior of humans, which is considered as the weakest link in the field of information security. This design of this attack is shown in Figure 8.

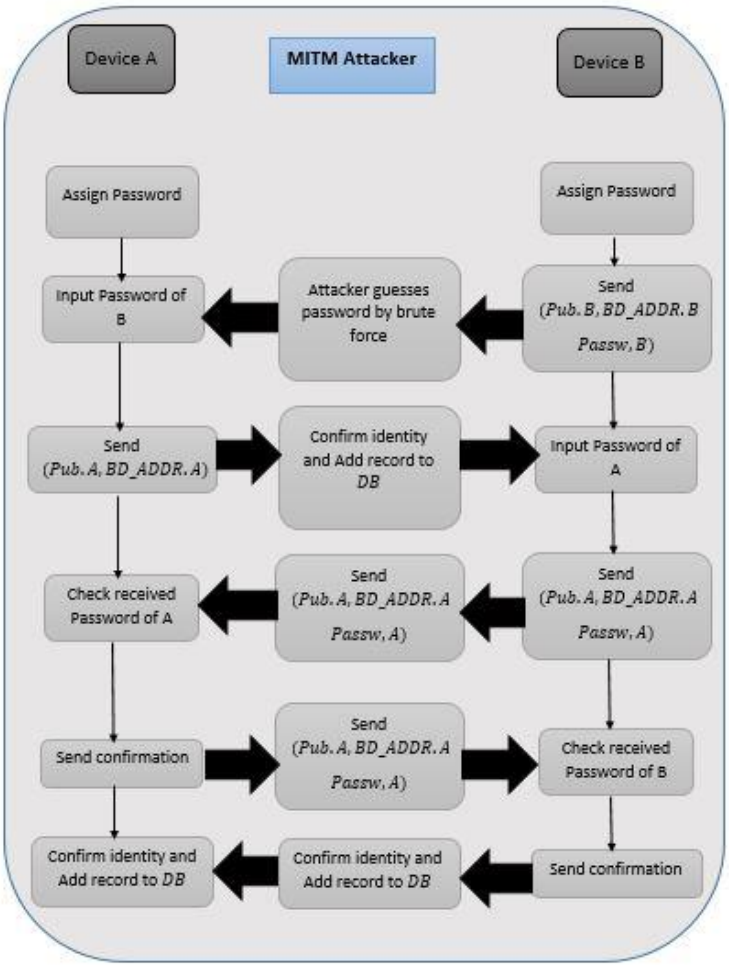

Figure 8: First phase of MITM attack

The second phase of MITM attack is quite risky, in the process of pairing data base is stored in each device, and this data base is used during pairing and remain in device after pairing. This data is not needed when already paired device connects again, targeting this data can

jeopardize the entire process of pairing. After collecting keys of all pairing devices, attackers can pair and exploit each of the connected victims. The block diagram of this phase is shown in Figure 9.

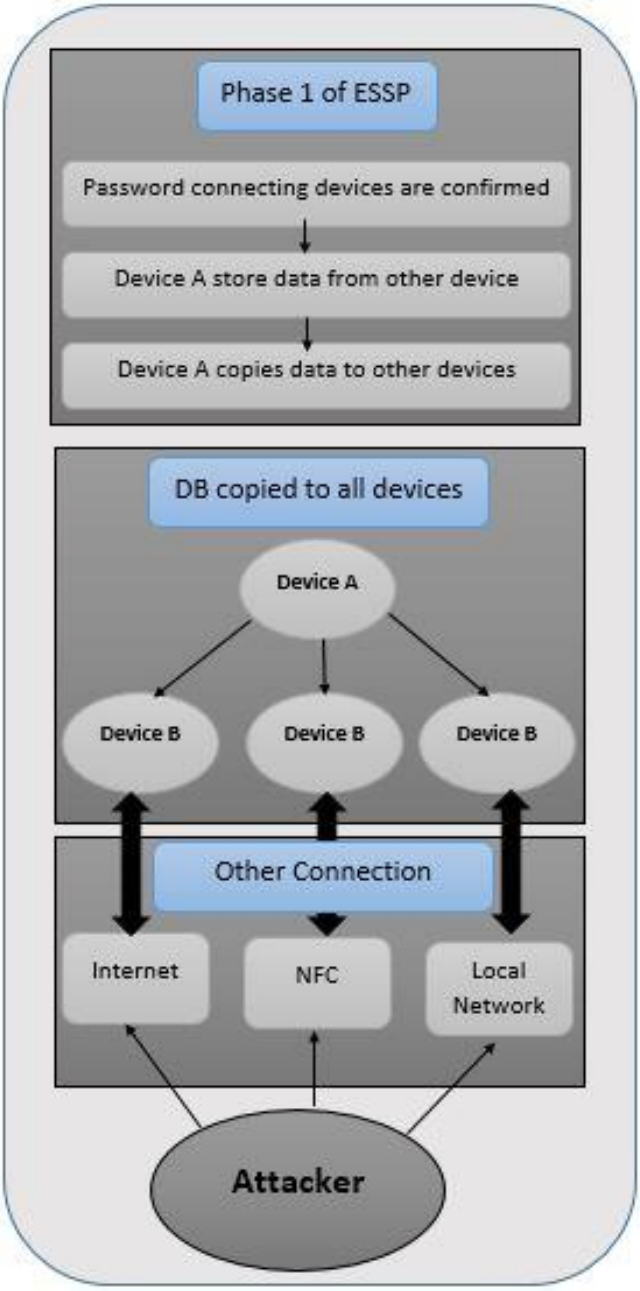

Figure 9: Second phase of MITM attack

5.4. Third Attack (Attack on JW model)

Through validation of process it is observed that modified JW model poses more resistance against MITM attack compared to scenario where parameters are not changed. However all these counter measures were calculated for SSP model, new approaches for thwarting MITM attacks are necessary. Table 2 and Table 3 show specifications of both scenarios. During the pairing process, first step is to compare I/O capabilities of both devices, all the further processing is decided by this step. As discussed earlier, this point is exploited by attacker to force victim in using JW model. As a result no verification is needed during pairing process as JW model doesn't proved protection against MITM attacks. To address this issue, four phase solution is derived with an introduction of new virtual channel, as shown in Figure 16.

Table 2: Specification of first scenario

Device address and name	**Device capabilities**	**Attacker modified I/O capabilities**
GT-P522 (E4:92:FB:0C:6E:A9)	Yes/No display and keyboard	Yes/No display and keyboard
SM-T535 (A0:b4:A5:68:78:EC)	Yes/No display and keyboard	Yes/No display and keyboard

Table 3: Specification of second scenario

Device address and name	**Device capabilities**	**Attacker modified I/O capabilities**
GT-P522 (E4:92:FB:0C:6E:A9)	Yes/No display and keyboard	Yes/No display and keyboard
SM-T535 (A0:b4:A5:68:78:EC)	Yes/No display and keyboard	No In No Out

.

To further explore the idea of security and vulnerabilities of Bluetooth device, fixed coordinate invalid curve attack is also part of this thesis. Fixed coordinate invalid curve attack is designed for two cases, Semi-passive attack and fully active MITM attack.

5.5. Fourth Attack (Semi-passive attack)

Using the information discussed in section 5.1, it is noted that ECDH in case of both point doubling and point addition do not use parameter b. If elliptic curve group E, and a point $Q = (Qx, Qy), Q \in E$ is such that $Q' = (x, 0)$ is the projection along x-coordinate, different elliptic equation can be easily derived as $y^2 = x^3 + ax + b'$, which has same parameter a, but different parameter b'. The inverse of point Q is formulated by reflecting its projection across x-axis $Q^{-1} = (Qx, -Qy)$. This results in every point with x-axis and zero y-axis equal to its own inverse, hence order two. This is illustrated in Figure 10.

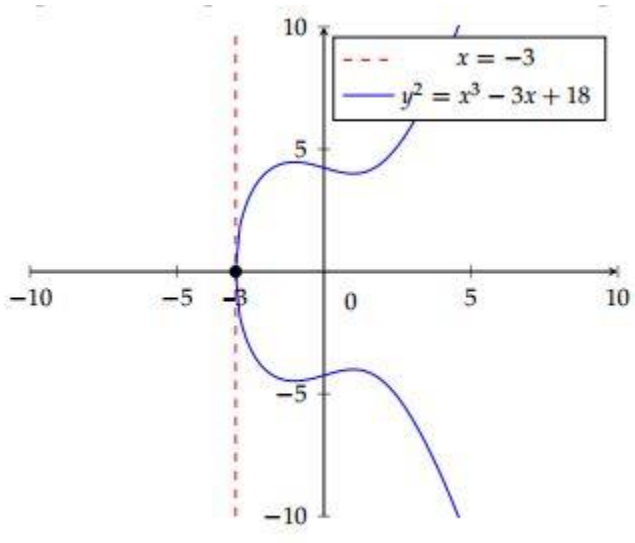

Figure 10: Example of an elliptic curve with an order-two point at (3, 0)

The new curve parameter $b' \equiv -x^3 - ax$, y-axis is forced to be zero on given points. Keeping in mind this vulnerability, new invalid curve attack is discussed, as there is no verification of y-axis, attacker can manipulate configuration by mapping each of public keys on x-axis. The success of this method relies on the values of private keys, SKa and SKb. Only in case of both private keys

are even, this attack is used with success probability of 0.25. This method is totally not detectable as x-coordinate is not changed. To describe semi passive attack, following pattern is followed.

1. During pairing, eavesdropping is done.
2. Without any interruption, pairing process is executed smoothly.
3. Exchange of ECDH keys are made between both devices
4. Y-coordinate is made zero during transmission for both devices.
5. If pairing is successful, next step is executed otherwise attack is aborted.
6. Derive ECDH key using public key and public parameters.
7. After successful pairing, decrypt the information flow between both devices to derived keys.

Message interception in phase two is shown in Figure 11 and Figure 12 shows the process after interception in phase two.

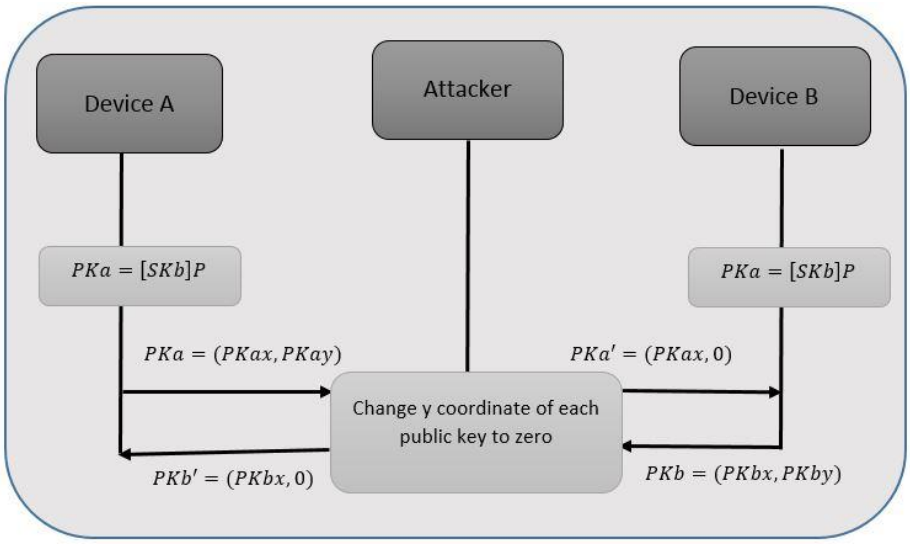

Figure 11: Phase two of fixed coordinate invalid curve attack

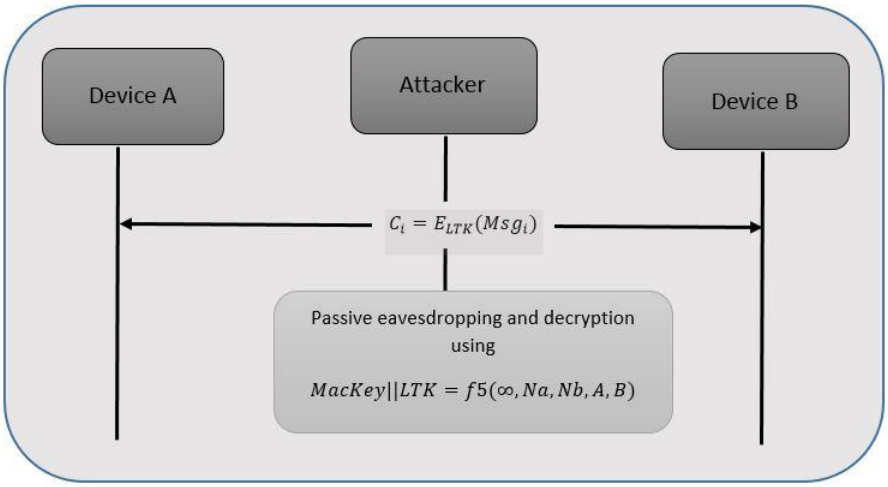

Figure 12: Passive Eavesdropping during semi passive MITM attack.

5.6. Fifth Attack (Fully active MITM attack)

The semi passive attack only works during second stage with probability of 0.25, to further improve the probability of success up to 0.5 by also intercepting message in fourth stage. For fully active MITM attack, four possible Diffie-Hellman keys are used.

Table 4: The possible values for shared keys

DHKey_a	DHKey_b
∞	∞
∞	PKa'
PKb'	∞
PKb'	PKa'

As discussed earlier, in phase four both devices exchange check values for validation of session key. Using $DHKey$, nonce, addresses of devices, session key is derived. Before phase 4, all values are publically available except $DHKey$ and addresses of devices. Passkey entry require 6-digit decimal number consist of 128-bit integers. This key is unknown to attacker but can be retrieved by small effort. This can be done by iteratively extracting all possible values from phase three. This is achieved using computation of 2.20=40 options. In phase four check byte of sender is validated by receiver using $DHKey_a$ from function $f6$. This make it possible for attacker to derive $DHKey_a$ from two possible values, PKb' or ∞. For $DHKey_a = \infty$, same approach as semi passive attack is used. If $DHKey_a = PKb'$, the unknown value of $DHKey_a$ must be either ∞ or PKa'. As a result check value can be determined, and receiver will send its

check-value. In opposite case the receiver will send message 'Pairing Failed'. This attack is as follows,

1. Repeat same process of semi passive attack for step 1-3.
2. Leave out third step of pairing without any intervention.
3. In fourth step after receiving check-value, change so that it may not reach to receiver. This is seen in Figure 13.
4. Check either check-value is equal to $PKb' or \infty$.
    a) $DHKey_a = \infty$: send original check-value and continue as semi passive attack
    b) $DHKey_a = PKb'$: $DHKey_a = \infty \; or \; DHKey_a = PKa'$
5. Ea' is generated and transmitted in the place of $DHKey_a$, that is distorted in step 3.
6. Compute session keys with Diffie-Hellman keys.
7. If this fails, decrypt information flow and pass-on different information.

This modification in previous attack result better success probability as attacker remain present rest of the session during MITM attack. The relay operation is demonstrated in Figure 14.

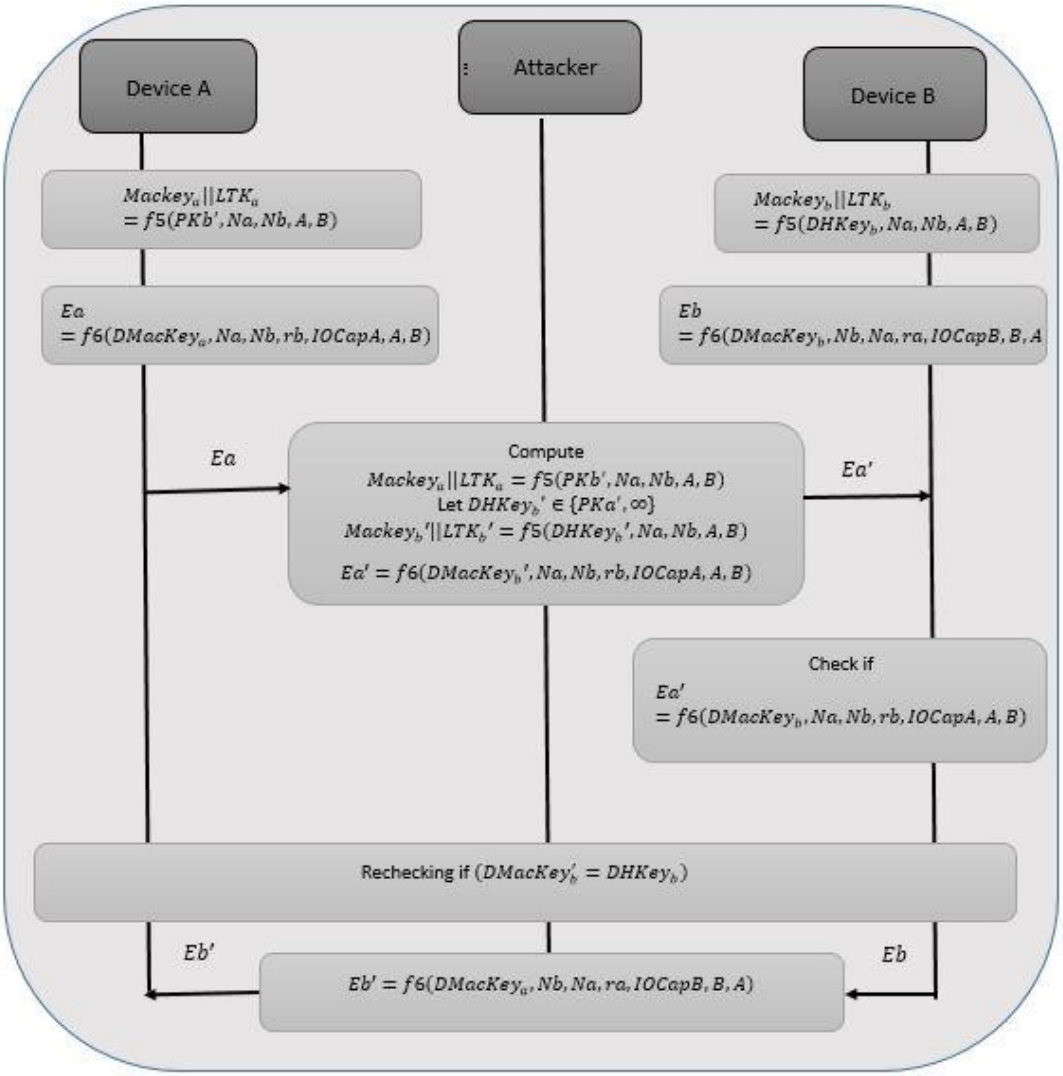

Figure 13: Phase 4 of fully active MITM attack

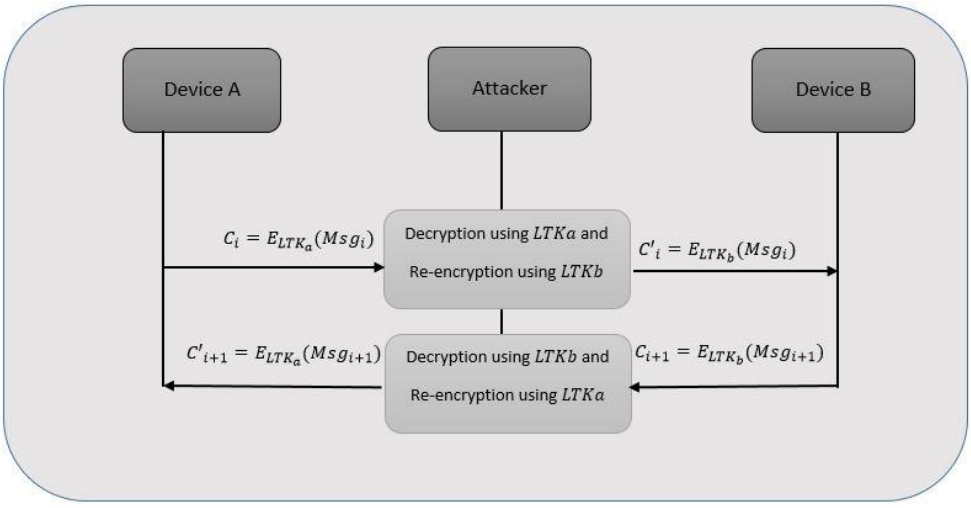

Figure 14: Relay of Fully active MITM attack

5.7. Success Rate

To calculate the success rate of both methods, it is assumed that private keys SK_x are randomly selected. In semi passive attack, described in section 5.3.1, it is assumed that both keys are subjected to infinity. Such that event $V = (DHKey_a = \infty) \bigcap (DHKey_b = \infty)$. The probability of this event is $\Pr(V) = 25\%$ as represented in Table 5.

Table 5: Success rate of semi passive attack

$DHKey_b$ / $DHKey_b$	‘ ∞	PKa'
∞	Success	Failure
PKb'	Success	Failure

Similarly in case of fully active MITM attack, private keys SKx are assumed to be randomly generated. The success rate depend on the success guess from attacker with event $U = (DHKey_a = PKb') \bigcap (DHKey_{b'} = DHKey_b)$. This event has success probability of 25%, and in full attack mode, $\Pr(U \bigcap V)\ 25\% + 25\% = 50\%$,as represented in

Table 6: Success rate of fully active MITM attack.

$DHKey_b$ / $DHKey_b$	∞	PKa'
∞	Success	Failure
PKb'	Failure	Success

Chapter 7

6. Suggestive Measures

This section includes the suggestive measure taken to avoid attacks discussed in previous section. The detail analysis of each suggestive measure is done along with graphical explanation.

6.1. Suggestive measures against ESSP attack

Considering these vulnerabilities, this thesis consider countermeasures to avoid exploitation of such vulnerabilities. Strengthening the ESSP structure, few countermeasures are proposed, first is to increase the length of password, hence increasing the entropy and resulting in difficult cracking. The second solution is to increase the extra level of security by assigning the commands to prevent access to database. The steps results quite effective against MITM attacks.

Attackers most frequently used JW model due to its vulnerabilities, to strengthen this model, three separate phases solutions are proposed.

Other countermeasures involves are incorporated in the JW association model. This method does not involve major changes in the SSP architecture. Attackers most frequently used JW model due to its vulnerabilities, to strengthen this model, three separate phases solutions are proposed. In first phase, to check the I/O capabilities of previous phase, inquiry is done to end user by comparing the I/O capabilities of both end user and selected device. If this comparison is not same, process is terminated, otherwise in case of match the fixed value is assigned according to answer. Phase two involve the generation of random number which include the first phase value, both devices share these values. Last phase protects the connection process.

The step wise implementation of three phase JW model is followed as

16. I/O capabilities of both devices are checked to verify if they are according to user's requirements. In case of mismatch, process is terminated and in case of successful match private ECDH keys are generated by devices. ECDH keys are skA and skB, as a result of match, initiator device sends $(IOcap\ A, Random\ number$ and $bluetooth\ address\ of\ initiator)$.

17. Similarly receiver device also computes its public key (PKB), which is share along with $(IOcap\ B, Random\ number$ and $bluetooth\ address)$.

18. After receiving key from other device, initiator device compute its own public key (PKA), along with $(IOcap\ B$ and $Random\ number)$.

19. Receiver device computes private key and shared key $DHKey$ for (PKA), $DHKey$ is computed using P192 function and receiver device computes its commitment value CA, as function $f1(DHKey, IOcapA, IOcapB, RNA, and\ RNB)$. The CA is send to receiver device form initiator device.

20. In similar fashion receiver device also generates it's CA and compare it with receivedCA, mismatch results in termination of process.

21. Link key (LK) is generated by both devices from function
$f2(DHKey, Noncemaster, Nonceslave, "btlk", BD\_ADDRmaster, BD\_ADDslave)$.

22. Finally from hash function $E3(LK, EN\_RAND, COF)$, encryption keys (KC) of both devices are created.

Following these steps results in secure network against MITM attacks. Figure 15, shows the process of JW modified model and SDP model.

6.2. Suggestive measure against JW model using SDP

The phase vise introduction of SDP method is described as follows,

Phase one

1. When both devices starts communication, virtual channel is established using combination of 4 digit PIN and random letter.

2. These combination of random number and PIN are then executed for three consecutive rounds with shuffling in each round.

3. Last values and values of each rounds are obtained for both devices.

4. Both devices generates their public keys and exchange it to create their own private Diffie-Hellman key.

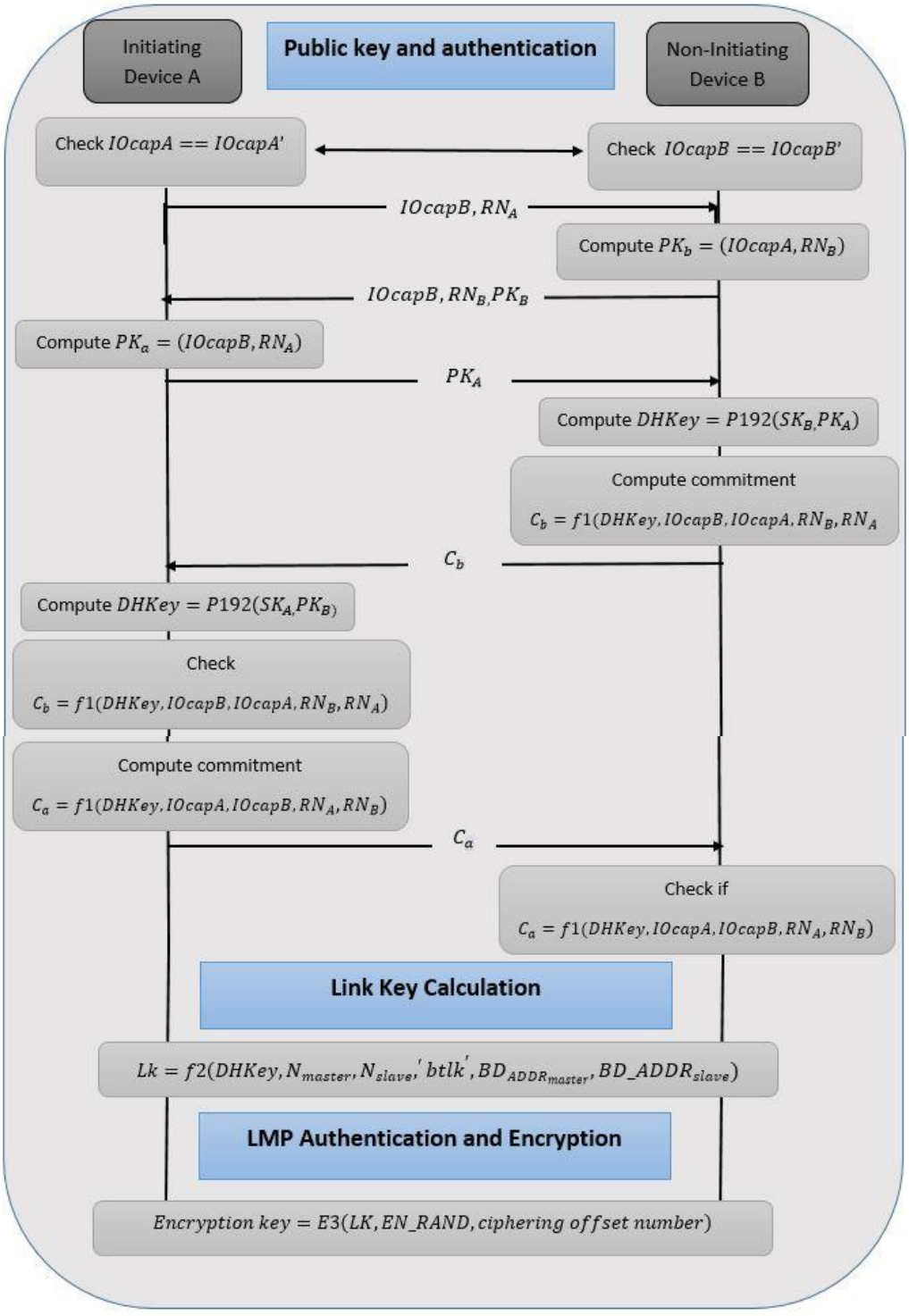

Figure 15: JW association model

Phase 2

1. The final set of values ($Value\_ID$) from virtual channel are then sent to each device.
2. Using function $f1(DHKey\ , PKB\ , PKA\ , Value\_ID, and\ PIN)$, receiver device computes it commits Cb.
3. Similarly sender device also computes it Cb and send it to virtual channel for verification. In case of fail verification, connection aborts and sender device computes CA using function $f1(DHKey, PKA, PKB, Value\_ID, and\ PIN)$.
4. In similar fashion receiver device verifies its results by sending it to virtual channel and in case of failure communication is terminated.

Phase Three

1. Both devices exchange set of letters generated gathered during the rounds, La and Lb.
2. Both devices builds matrices that are multiplied with each other. If verification of both side is not correct, connection is terminated.

Phase Four

Using function $2(DHKEY, NMASTER, NSLAVE, ROUNDI, BD\_ADDRMASTER, BD\_ADDRSLAVE)$, link key LK is generated by both devices. During final phase, one random value is selected and verified over virtual channel, this results as the final vetting point for connection.

This four phase solution provide and unbreakable protection against MITM attacks

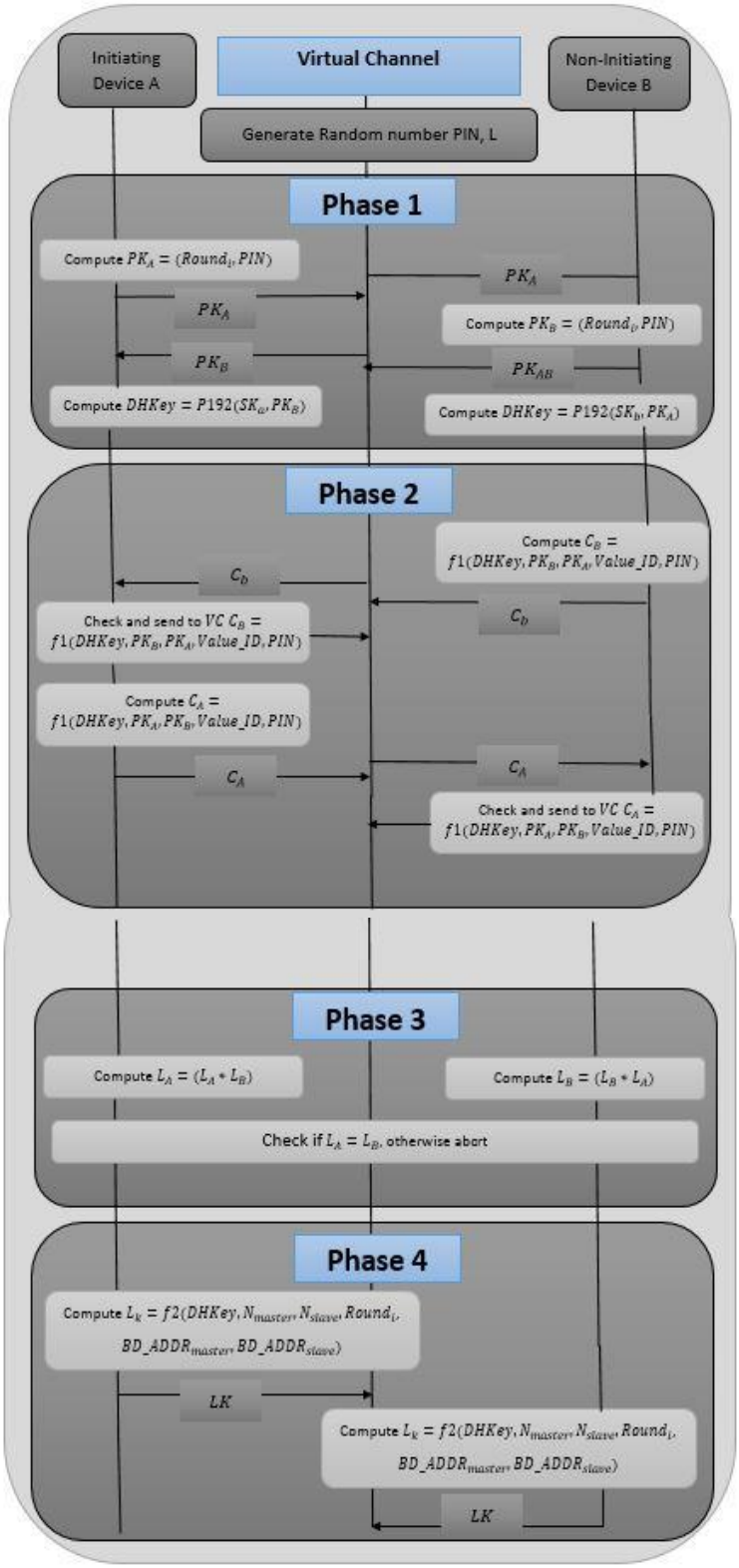

Figure 16: SDP structure

6.3. Suggestive Measures against Semi Passive Attack

Suggestive measures for avoiding semi passive attack is to validate elliptic curve and check either ECDH public key satisfies the curve equation. The main mitigation involve concept that order n of the base point is prime P. Similarly, to validate that either public key (PK) satisfies $[n]PK = \infty$. This measurement is important as it verifies the order of given key and prevent short public keys with small orders. This suggestive measure is successful on both LE SC and SSP.

The theoretical suggestion that can be implemented in future versions of Bluetooth to avoid semi passive attack include numeric comparison protocol. This protocol provide security in link layer, and negotiate link key after NC. Each device involve in the comparison of public key include key exchange, this public key is available to attacker. To avoid this using NC protocol, public key will be generated and shown only one time for any instance during pairing. User can confirm key and validate the process of pairing or terminate it, this result in more secure method that only make public key available for instance. Multiple comparison of course have more limited use in practice and alert the possible attacker. The general structure of NC protocol for the prevention of attack is discussed below.

The notations used for OOB and NC protocols are defined in

TERM	DEFINITION
$\boldsymbol{\Pi_1}$	NC protocol
$\boldsymbol{\Pi_2}$	OOB protocol
$\boldsymbol{\Pi}$	Any of Π_1 or Π_2
$\boldsymbol{\Pi^{A,B}}$	Bluetooth device A's Π_1 or Π_2 instance run with Bluetooth device B
$\boldsymbol{\Pi^{B,A}}$	Bluetooth device A's Π_1 or Π_2 instance run with Bluetooth device B
LK	Link Key
P256/P192	Connections
$\boldsymbol{ac_x}$	Decision of X instances,

$\boldsymbol{C_x}$	Commitment generated by Bluetooth device X
$\boldsymbol{N_x}$	Nonce (Unique random number)

Pre-protocol exchange: Bluetooth addresses of both devices A and B are exchanged along with IO capabilities, $IOcapA$ and $IOcapB$.

Phase 1: Public Key Exchange

1. The initiating device A generates its own private-public key pair (SKa, PKa) and sets ac_a= *. Here, the private-public key pair is generated only once per device and may be computed in advance of pairing. And then, the device A sends PKa to the non-initiating device B.

2. Similarly, the device B generates(SKb, PKb), sets ac_b= *, and sends PKb to the device A.

3. The device A computes $DHKey = P192(SKa, PKb) = SKaPKb$ or $P256(SKa, PKb) = SKaKb$. The device B computes $DHKey = P192(SKb, PKa) = SKb.PKa$ or $P256(SKb, PKa) = SKb.PKa$.

Phase 2: Authentication stage 1 for NC

1. The device A selects random Na and sets ra and rb to 0.

2. The device B selects random Nb and sets rb and ra to 0. The device B further computes $Cb = f1(PKbx, PKax, Nb, 0)$, where $PKbx$ and $PKax$ respectively denote the x-coordinate of the public keys PKb and PKa. Then, the device B sends Cb to the device A.

3. The devices A and B exchange their Na and Nb.

4. Upon receiving Nb, the device A checks if $Cb = f1(PKbx, PKax, Nb, 0)$. If the check fails, the device A sets ac_a = false and aborts the protocol execution.

5. The device A computes $Va = g(PKax, PKbx, Na, Nb)$ and displays$V\ a$. Similarly, the device B computes $Vb = g(PKax, PKbx, Na, Nb)$ and displays Vb.

6. User checks if $Va = Vb$ and confirms on each end. If user confirms 'yes', proceed the following phase; otherwise both the device A and the device B set ac_a = false and ac_b= false and terminate it, respectively.

Phase 3: Authentication stage 2

1. The device A computes $Ea = f3(DHKey, Na, Nb, rb, IOcapA, A, B)$ and sends Ea to the device B.

2. Upon receiving Ea, the device B checks whether $Ea = f3(DHKey, Na, Nb, rb, IOcapA, A, B)$. If check fails, the device B must set ac_b= false and abort the protocol execution. Otherwise, the device B accepts the integrity of the device A and sets ac_b= true, computes $Eb = f3(DHKey, Nb, Na, ra, IOcapB, B, A)$, and sends Eb to the device A.

3. Upon receiving Eb, the device A checks whether $Eb = f3(DHKey, Nb, Na, ra, IOcapB, B, A)$. If check fails, the device A must set ac_a= false and abort the protocol execution; else the device A accepts the integrity of the device B and sets ac_a = true.

Phase 4: Link key calculation

1. The device A computes $LK = f2(DHKey, Na, Nb, "btlk", BD\_ADDRa, BD\_ADDRb)$.

2. The device B computes $LK = f2(DHKey, Na, Nb, "btlk", BD\_ADDRa, BD\_ADDRb)$.

7.4. Suggestive Measures against Fully Active Attack

Fully active attack exploit vulnerability present in fourth phase as discussed in section 5.6. To avoid this attack mitigation process involve the validation of y-coordinate. This is achieved by checking whether y −cordinate equals zero. This mitigation might not be that effective against extended version of same attack using high order points, but for this scenario it protects against fully active attack. Thus mitigation model include process of zeroize the y −cordinate just before transmission. On the receiver end, same process is repeated and just after receiving the remote public key, zeroize its y −cordinate.

Suggestive approach for avoiding fully active attack in future versions of Bluetooth is to use out of band protocol (OOB). Using OOB channel provide security as attacker might not be able to listen messages in this channel. OOB channel also provide security against modification, injection, and delete messages. In SSP protocol the secret key is shared only after protocol execution, using OOB attacker will only succeed in case he knows link key, and passes the authentication process of association model according to protocol execution. Another

importance of using OOB is that it run in sequential manner that stop Bluetooth device to run multiple times. Bluetooth is run for multiple time due to only one available interface for display, comparison, and near field communication. OOB model is similar to NC model, only phase 2 is different, phase 2 of OOB model is shown below

Phase 2: Authentication stage 1 for OOB

1. The device A sets $ra = rand_1$ and $rb = 0$ and the device B also sets $rb = rand_2$ and $ra = 0$. Here, $rand_1$ and $rand_2$ are random numbers.
2. The device A computes $Ca = f1(PKax, PKax, ra, 0)$, where $PKax$ denotes the x-coordinate of the public key PKa. And the device B also computes $Cb = f1(PKbx, PKbx, rb, 0)$, where $PKbx$ denotes the x-coordinate of the public key PKb.
3. The device A sends $A, ra, and\ Ca$ to B through the human-aided OOB channel. And the device B also sends $B, rb, and\ Cb\ to\ A$ through the human-aided OOB channel.
4. Upon receiving $B, rb, and\ Cb$, the device A resets rb to the received value and if $Cb \neq f1(PKbx, PKbx, rb, 0)$ the device A sets ac_a = false and aborts the protocol execution. If step 3 received and B's IO capability does not indicate OOB authentication data present set, set $ra = 0$.
5. Upon receiving $A, ra, and\ Ca$, the device B resets ra to the received value and if $Ca \neq f1(PKax, PKax, ra, 0)$ the device B sets ac_b = false and aborts the protocol execution. If step 3 received and $A's$ IO capability does not indicate OOB authentication data present set, set $rb = 0$.
6. The device A selects random Na and the device B selects random Nb.
7. The devices A and B exchange Na and Nb to each other.

7.5. Security model

In this section security model is evaluated to secure SSP, this model simulates the networking of Bluetooth device and catches potential attacks during public channel conversations. The advantage of using this model is that it can be adjusted into any particular deployment model. The main application of this security model is demonstrated for home automation and entertainment (HAE) systems.

7.5.1. Background

The interaction of Bluetooth devices is not via remote channels, so short numeric comparisons and pass key entry is done using Bluetooth devices. To avoid any MITM attack, the non-remote channel behavior of Bluetooth can be of great use, this is done by combining NC and OOB protocol.

1). Each Bluetooth device involved in key exchange procedure holds a local public 'comparison variable'. The disadvantage of this process is that local public variable can be seen by attacker, so comparison variable is generated and shown only once in any instance of NC protocol. This allow user to only compare and respond one time, multiple comparisons would have alert the possible attacker.

2). Using OOB channel, attacker is only able to listen the messages in OOB channel. This make conversation secure as attacker is not able to modify, inject, or delete messages over the OOB channel.

The advantage of using proposed method is that SSP does not have public key infrastructure, which is in contrast to general shared secret setting. This make SSP secure against fully active and semi passive attacks. Conventional models allow attacker to involve in key exchange that enable him to obtain secret key generated at the end of a protocol execution. Using proposed method, the security is made stricter as attacker will need both ink key and also passes the authentication process of association model according to Phase 2 in each protocol execution.

7.5.2. Attacker

To avoid interference of attacker during communication, different set of oracles are implemented. By giving access to protocol instances, control of attacker is modeled. Following patterns are followed:

8. $Init(A, B)$: $\Pi^{A,B}$ and $\Pi^{B,A}$ for Π instances are initiated. For all scenarios, A is assumed to b initiating device, while B is on receiving end. For both devices, if an instance run, oracle doesn't do anything. In the meanwhile, the $Init$ oracle will initiate another instance with new protocol
9. Send ($\Pi^{A,B}, M$) or Send ($\Pi^{B,A}, M$): In this oracle, message M is sent to the device A's $\Pi^{A,B}$ or the device B's$\Pi^{B,A}$. The output is set after the execution of current message is

done, upon receiving under same execution time, the attacker can attack using send oracle protocol.

10. Execute (A,B): In this oracle, complete protocol execution is done. This protocol executes transcript in the form of complete message. This oracle is successful in simulating the adversary in passive eavesdropping. This will not provide any information to the adversary.
11. Reveal $\Pi^{A,B}$ or $\Pi^{B,A}$: Upon calling this oracle, both devices generates protocol execution at the end, if any of device doesn't generate protocol execution, then output is null. This indicate the exposure of key to the adversary. In order to make secure connection, this protocol needs to make independent link key for both devices.
12. Test $\Pi^{A,B}$ or $\Pi^{B,A}$: The oracle is designed to verify the security of the protocol. The success of this oracle is dependent on the random link key generation on the principle of coin flip. This random key is set independent of protocol, and attacker is thus only allowed to test oracle once. The random process involve the guessing of attacker about the link key more difficult.

Bluetooth home automation is an excellent example of applying proposed method into real world application. Figure 17 shows the application of Bluetooth in HAE system.

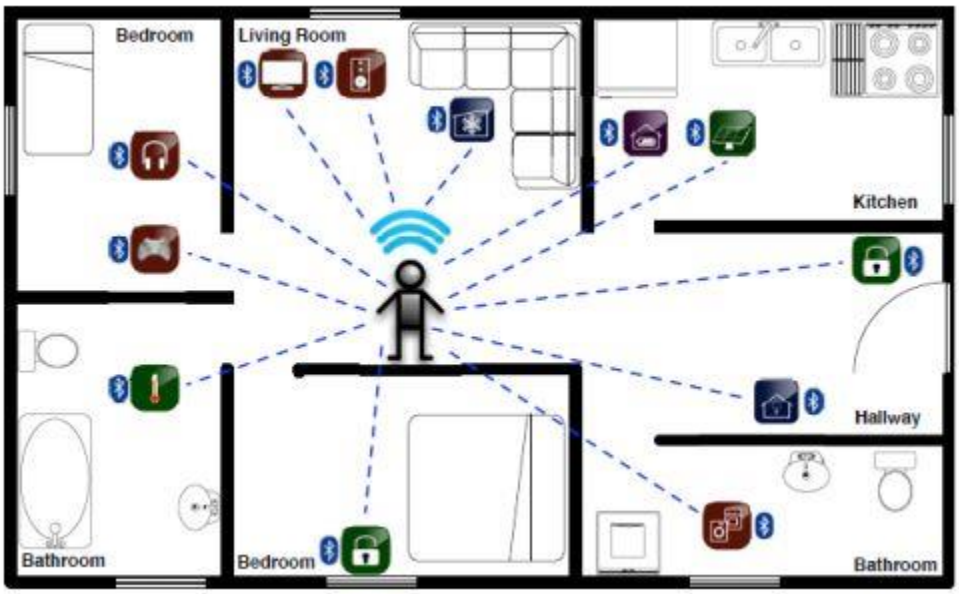

Figure 17: Bluetooth based home automation and entertainment (HAE) system

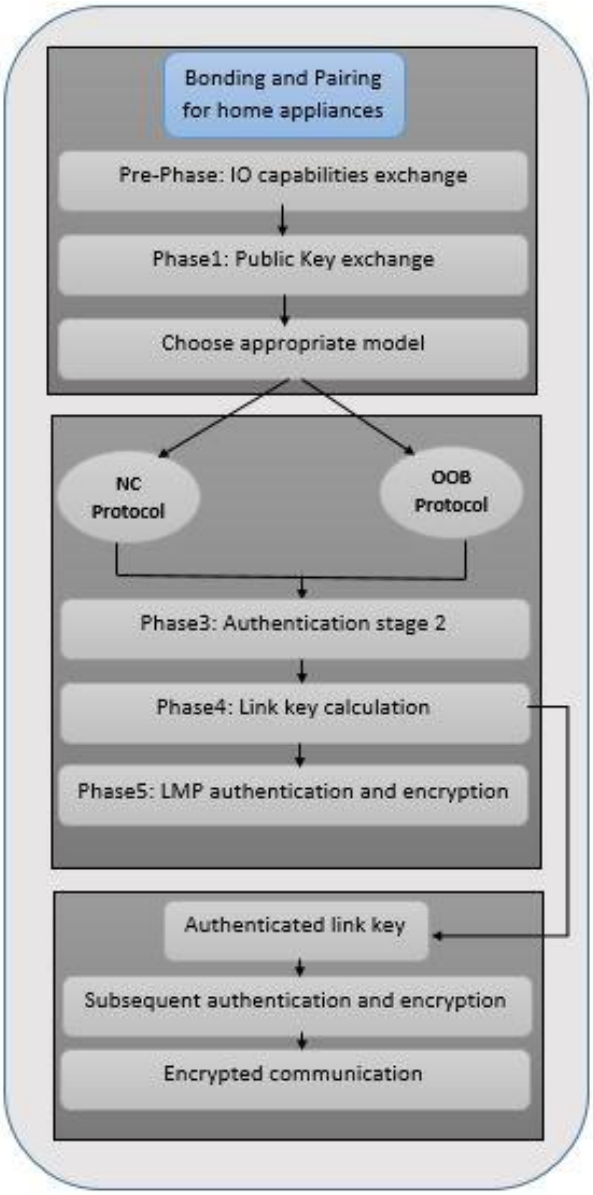

Figure 18: Secure HAE system

Chapter 7

7. Conclusion

Bluetooth is an innovative device that has revolutionized the way of today's communication. The wide network communication application has made Bluetooth center of attraction for hackers by exploiting its vulnerabilities. Bluetooth structure was designed by Ericsson without considering its security aspects, hence since its birth hackers and researchers have exposed many vulnerabilities present in Bluetooth architecture. Till today advance Bluetooth devices still have vulnerabilities that needed to be exposed for better security countermeasures. This thesis propose several MITM attacks that are present in present SSP structure of Bluetooth.

In this thesis it has shown that current structure of SSP needed changes to avoid MITM attacks. Previous methods present in literature are not sufficient enough to prevent MITM attacks during Bluetooth communication. To address such vulnerabilities, we have presented several attacks and their countermeasure that are not only efficient and feasible but also offer confidentiality and integrity of data during Bluetooth communication that can be successfully implemented in current Bluetooth versions.

First method is by using ESSP instead of SSP to introduce extra virtual channel for verification. This method improves the vulnerabilities and which stood better against proposed attacks in comparison to previous methods. To avoid vulnerabilities in using ESSP successful theoretically developed attacks and their counter measures are also discussed in this thesis.

Second method is focused on JW model, using JW model we have explored its limitation by discussing it vulnerabilities and then presenting countermeasures against it. Two other attacks and countermeasures are also proposed by focusing on the ECDH. Using fixed coordinate invalid curve attacks, new theoretical attacks are presented on ECDH during Bluetooth pairing process.

Third method is based on fixed coordinate invalid curve attack by introducing semi passive attack with success probability of 25%. This attack is based on vulnerability that in whole

process of paring using ECDH, y-coordinate is not validated, hence using this vulnerability effective MITM attack is proposed and it countermeasure is also discussed.

Fourth method is also developed for ECDH pairing process by eavesdropping and distorting check byte. This method has success probability of 50%.

During the case of this study, we have explored many vulnerabilities present in Bluetooth devices and discussed in details different scenarios of MITM attacks. It is recommended to Bluetooth special interest group fix these flaws for better security in communication.

It is very important to note that this research work is merely a discovery work, to discover countermeasures against MITM attacks. MITM attacks present during pairing process and their loopholes are discussed in detail with several new attacks presented to exploit existing loopholes in Bluetooth structure. Theoretical countermeasure are also derived to guide manufacturers in improving the existing Bluetooth devices by removing these vulnerabilities. However, this research work need to be relied on subsequent studied to confirm its applicability.

Chapter 8

8. Future work

Focus of this study was on MITM attacks exploiting vulnerabilities present in current Bluetooth devices. To avoid such attacks, this thesis has presented different countermeasures, although implementing them will improves Bluetooth security but there are some ideas that can be used as future direction for researchers.

First idea is to develop such empirical model that can estimate the level of damage done by MITM attack. Developing such model will yield an accurate estimate of damage done during attack, hence guide developers for an appropriate countermeasure. The latest Bluetooth devices (versions 4.2 and 5.0) vulnerabilities should be analyzed and evaluate security risks present in these models. The proper study of finding vulnerabilities present in latest model by introducing series of attacks and then giving countermeasures against it need of time.

Security of Bluetooth device is essential for future of Bluetooth device, pairing is most vulnerable part of Bluetooth devices and with introduction of many mitigation techniques is still not secure enough. Thus implementing these future ideas will draw roadmap for researchers to develop methods and tools for better security of Bluetooth communication.

9. References

[1] IEEE, “Specification of the Bluetooth System,” vol. 1.0, 1999.

[2] N. Davies, A. Friday, P. Newman, S. Rutlidge, and O. Storz, “Using Bluetooth Device Names to Support Interaction in Smart Environments,” *Proc. 7th Int. Conf. Mob. Syst. Appl. Serv. ACM,* pp. 151–164, 2009.

[3] K. Ritvanen and K. Nyberg, “Key Replay Attack on Improved Bluetooth Encryption,” *Nokia Res. Cent.*

[4] Y. Lu, “Short Notes on Security of Bluetooth Encryption Standard E0 Core Short,” no. July, pp. 2–5, 2016.

[5] M. A. Albahar, O. Olawumi, K. Haataja, and P. Toivanen, “Novel Hybrid Encryption Algorithm Based on Aes , RSA , and Twofish for Bluetooth Encryption,” *J. Inf. Secur.*, vol. 9, pp. 168–176, 2018.

[6] A. A. Veiga and C. J. B. Abbas, “Proposal and Application of Bluetooth Mesh Profile for Smart Cities ’ Services,” *Smart Cities*, 2018.

[7] J. Song, R. Poovendran, and J. Lee, “The AES-CMAC-96 Algorithm and Its Use with IPsec,” *Netw. Work. Gr.*, pp. 1–8, 2006.

[8] V. S. Miller, “Use of Elliptic Curves in Cryptography,” *Expioratory Comput. Sci. IBM Res.*, pp. 417–426, 1986.

[9] E. Biham and L. Neumann, “Breaking the Bluetooth Pairing – Fixed Coordinate Invalid Curve Attack,” *Tech. Inst. Technol.*, 2018.

[10] A. Antipa, D. Brown, and A. Menezes, “Validation of Elliptic Curve Public Keys,” *Int. Work. Public Key Cryptogr. Springer, Berlin, Heidelberg,* pp. 211–223, 2003.

[11] I. Biehl, B. Meyer, and V. Muller, “Differential Fault Attacks on Elliptic Curve

Cryptosystems," *Annu. Int. Cryptol. Conf. Springer, Berlin, Heidelb.*, pp. 131–146, 2000.

[12] K. V. S. S. S. S. Sairam, N. Gunasekaran, and S. Reddy, "Bluetooth in Wireless Communication," *Top. Broadband Access*, no. June, pp. 90–96, 2002.

[13] E. J. Candès, "Compressive sampling," in *Proceedings of the International Congress of Mathematicians Madrid, August 22–30, 2006*, 2009.

[14] B. SIG, "Bluetooth Core Specification Master."

[15] M. Ryan, "Bluetooth : With Low Energy comes Low Security," *Present. as part 7th {USENIX} Work. Offensive Technol.*, 2013.

[16] D. Sun, "Man-in-the-middle attacks on Secure Simple Pairing in Bluetooth standard V5 . 0 and its countermeasure," *Pers. Ubiquitous Comput.*, pp. 55–67, 2018.

[17] G. Card and E. N. Supkrieure, "A New Public-Key Cryptosystem," *Adv. Cryptogr.*, pp. 27–36, 1997.

[18] W. Ledbetter, "ANALYZING INHERENT VULNERABILITIES AND ASSOCIATED RISKS IN BLUETOOTH TECHNOLOGY," *Thesis, Univ. South Alabama*, no. May, 2017.

[19] J. T. Vainio, "Bluetooth Security," *Proc. Helsinki Univ. Technol. Telecommun. Softw. Multimed. Lab. Semin. Internetworking Ad Hoc Networking, Spring*, vol. 5, 2000.