Security Enhancement for ZigBee and Bluetooth

Chen Chen
University of Windsor

Follow this and additional works at: https://scholar.uwindsor.ca/etd

Recommended Citation
https://scholar.uwindsor.ca/etd/5973

This online database contains the full-text of PhD dissertations and Masters' theses of University of Windsor students from 1954 forward. These documents are made available for personal study and research purposes only, in accordance with the Canadian Copyright Act and the Creative Commons license—CC BY-NC-ND (Attribution, Non-Commercial, No Derivative Works). Under this license, works must always be attributed to the copyright holder (original author), cannot be used for any commercial purposes, and may not be altered. Any other use would require the permission of the copyright holder. Students may inquire about withdrawing their dissertation and/or thesis from this database. For additional inquiries, please contact the repository administrator via email (scholarship@uwindsor.ca) or by telephone at 519-253-3000ext. 3208.
Security Enhancement for ZigBee and Bluetooth

by

Chen Chen

A Thesis
Submitted to the Faculty of Graduate Studies
through Electrical and Computer Engineering
in Partial Fulfilment of the Requirements for
the Degree of Master of Applied Science at the
University of Windsor

Windsor, Ontario, Canada

2017

© 2017, Chen Chen
Security Enhancement for ZigBee and Bluetooth

by

Chen Chen

APPROVED BY:

J. Chen
School of Computer Science

H. Kwan
Department of Electrical and Computer Engineering

H. Wu, Advisor
Department of Electrical and Computer Engineering

May 23, 2017
AUTHOR’S DECLARATION OF ORIGINALITY

I hereby certify that I am the sole author of this thesis and that no part of this thesis has been published or submitted for publication.

I certify that, to the best of my knowledge, my thesis does not infringe upon anyones copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my thesis and have included copies of such copyright clearances to my appendix.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office, and that this thesis has not been submitted for a higher degree to any other University or Institution.
ABSTRACT

With the development of wireless technology, the social daily life has an increasing relationship with the wireless networks and the issue of wireless network security has caught more and more attention. In this thesis, two new protocols for ZigBee security are proposed. For the first time public key technology has been used to enhance the security strength for ZigBee master key establishment. The proposed protocols strengthened ZigBee master key establishment security, which subsequently secure the establishment of the network key and link key, both derived from the master key. By integrating unbalanced RSA into the key establishment protocols, the new methods can distribute different computation amount to the ZigBee devices in communication based on their computational capacities.

An efficient key establishment protocol for Bluetooth is proposed. It also utilizes public key technology to avoid the involvement of a third party and to improve the security strength. This proposed work requires fewer protocol steps and is thus faster than the existing similar works.
DEDICATION

To my loving parents:

Father: Li Ming

Mother: Sun Zhiyi
ACKNOWLEDGEMENTS

I would like to express my sincere gratitude and appreciation to everyone who helped make this thesis possible. First of all, I would appreciate my parents. Their support encouraged me to overcome all difficulties that I met during my study. They supported me to study aboard and gave me financial support, without whom, I could not achieve my study. Also, I am deeply indebted to my supervisor Dr.H.Wu, professor of Electrical and Computer Engineering at University of Windsor, for guiding me throughout the writing of this thesis. As one of best teachers I have ever had, Dr.H.Wu impressed upon me that a good teacher instructs students in matters far beyond those in textbooks. His broad knowledge and logical way of thinking have been of great value; without his detailed and constructive comments on my research, none of this thesis would be possible.

from the School of Computer Science also gave me a lot of help. He read through my thesis page by page and gave me advice about how to use words in technical writing. I really appreciate it.

I would also grateful to my colleagues and friends, Ruiqing Dong, Huyuan Li, Bingxin Liu, Chunyu Mao and Siyu Zhang who gave me their help and time in listening to me and helping me work out my problems during the difficult course of the thesis.

Finally, I wish to extend my gratitude to the faculties of Electrical and Computer Engineering at University of Windsor for their efforts and help during my study for the master degree. Moreover, my thanks would go to the financial support from the University of Windsor and my supervisor Dr.H.Wu.

Chen Chen
# TABLE OF CONTENTS

**AUTHOR’S DECLARATION OF ORIGINALITY** iii  
**ABSTRACT** iv  
**DEDICATION** v  
**ACKNOWLEDGEMENTS** vi  
**LIST OF TABLES** x  
**LIST OF FIGURES** xi  
**LIST OF ACRONYMS** xii  

## 1 INTRODUCTION 1  
1.1 Research Motivation ............................................. 2  
1.2 Research Contributions ........................................... 2  
1.3 The Scope and Organization of the Thesis ....................... 3  

## 2 TECHNICAL BACKGROUNDS 4  
2.1 ZigBee System and Security ................................. 4  
2.1.1 ZigBee Network System Devices .......................... 4  
2.1.2 Three Kinds of ZigBee Network Topologies ................. 6  
2.2 Bluetooth and its Security ....................................... 8  
2.3 RSA and Unbalanced RSA ....................................... 9  
2.3.1 Development of RSA and Unbalanced RSA ................. 10  
2.3.2 RSA Algorithm ............................................ 11  
2.3.3 Unbalanced RSA Algorithm ................................ 13
2.4 Security Strength and Requirements ........................................... 14
  2.4.1 Security Strength of Key Length ........................................... 14
  2.4.2 Lightweight and Ultra-lightweight Security ............................. 15
2.5 Comparison Between RSA and Unbalanced RSA ........................... 15
  2.5.1 Reasons of Choosing Unbalanced RSA ................................... 16

3 AN OVERVIEW OF RELATED WORKS ............................................. 17
  3.1 Security Options for ZigBee and Bluetooth Standards ................. 17
  3.2 Key Structure in ZigBee and Bluetooth .................................... 18
  3.3 On Confidentiality in ZigBee and Bluetooth ............................... 19
  3.4 On ZigBee Authentication ...................................................... 20
    3.4.1 Certificate Authentication .............................................. 20
    3.4.2 Message Authentication Code .......................................... 23
    3.4.3 Network Key Authentication ............................................ 27
  3.5 Existing Works on ZigBee Key Management ................................. 28
    3.5.1 Group key protocols ..................................................... 28
    3.5.2 Symmetric Key Establishments ......................................... 30
    3.5.3 Pre-installed Master Key ............................................... 31
    3.5.4 User-entered Master Key ................................................. 32
    3.5.5 Key-transport Master Key ............................................... 32
  3.6 Recent Works on Bluetooth Security ......................................... 33
    3.6.1 Initialization Key Transmission ........................................ 33
    3.6.2 Unit Key Transmission ................................................... 34
    3.6.3 Combination Key Transmission ......................................... 35
    3.6.4 Master Key Transmission ................................................ 36
    3.6.5 Diffie-Hellman Scheme ................................................... 37
    3.6.6 Unbalanced RSA Protocol ................................................. 38
    3.6.7 Authentication in Bluetooth ............................................. 40
3.6.8 Challenge-response Scheme

4 PROPOSED WORKS

4.1 New Master Key Establishment Protocol in ZigBee

4.2 Simplify Master Key Establishment Security Protocol in ZigBee

4.3 Simulation Results for RSA and Unbalanced RSA Algorithms

4.3.1 Simulation Results

4.3.2 Analysis of Results

4.4 Enhancement for the Unbalanced RSA Algorithm in Bluetooth

5 CONCLUSIONS AND FUTURE WORKS

5.1 Conclusions

5.2 Possible Future Work

REFERENCES

VITA AUCTORIS
LIST OF TABLES

2.1 Relationship of Security level and RSA modulus size ............... 14
2.2 Comparison between RSA and unbalanced RSA ..................... 15
3.1 Notation and Abbreviation in MAC ................................. 24
4.1 Simulation Results for 1024 bits of the RSA and unbalanced RSA algorithms .................................................. 46
4.2 Simulation Results for 2048 bits of the RSA and unbalanced RSA algorithms .................................................. 46
# LIST OF FIGURES

2.1 Star Topology in ZigBee system[1] ........................................ 6
2.2 Cluster tree Topology in ZigBee system[2] ............................... 7
2.3 Mesh Topology in ZigBee system[1] ........................................ 8
2.4 Topology in Bluetooth system[3] ........................................... 9
3.1 Security in ZigBee and Bluetooth stack[4] ............................... 20
3.2 Authentication protocol between TC and MCA ......................... 21
3.3 Authentication protocol between TC and MED through MCA ...... 21
3.4 MAC Authentication Protocol ............................................. 24
3.5 Network key authentication .............................................. 27
3.6 Group key protocol ..................................................... 29
3.7 Symmetric key protocol ................................................ 30
3.8 Initialization key transmission protocol ................................ 33
3.9 Unit key transmission protocol ....................................... 35
3.10 Combination key transmission protocol ............................. 35
3.11 Master key transmission protocol .................................. 36
3.12 Diffie-Hellman scheme .............................................. 37
3.13 Key establishment protocol in Bluetooth using unbalanced RSA . 39
3.14 Challenge-response scheme protocol .............................. 40
4.1 Master key establishment protocol by unbalanced RSA algorithm . 42
4.2 Improvement master key establishment protocol by unbalanced RSA algorithm .............................................. 44
4.3 Enhancement key establishment protocol in Bluetooth ................ 49
LIST OF ACRONYMS

ACE  active constellation extension
ADC  analog to digital convertor
ADSL asymmetric digital subscriber line
API  average power increase
BER  bit error rate
CC   computational complexity
CCDF complementary cumulative distribution function
CDMA code-division multiple access
CP   cyclic prefix
DAB  digital audio broadcasting
DFT  discrete fourier transform
DIF  differential item functioning
DRL  data rate loss
DSP  digital signal processing
DVB  digital video broadcasting
FFT  fast fourier transform
GI   guard interval
HDSL high-bit-rate digital subscriber line
HPA  high power amplifier
ICI inter-carrier interference
IDFT inverse discrete fourier transform
IFFT inverse fast fourier transform
ISI inter-symbol interference
LO local oscillator
LTE long term evolution
MC multi-carriers
OBR out-of-band radiation
OFDM orthogonal frequency division multiplexing
PAPR peak to average power ratio
PBISLM partial bit inverted SLM
PPPW-SLM partitioning and partial-phase-weighting-SLM
PRC PAPR reduction capability
PRT peak reduction tone
PTS orthogonal frequency division multiple access
PTS partial transmit sequence
QAM quadrature amplitude modulation
QPSK quadrature phase shift keying
RF radio requency
RMS  root mean square
SC  single-carrier
SC-FDMA  single carrier frequency division multiple access
SI  side information
SLM  selected mapping
SNR  signal-to-noise ratio
SSPA  solid state power amplifier
TI  tone injection
TPPW-SLM  time-domain-partial-phase-weighting-SLM
TR  tone resercation
TWTA  traveling wave tube amplifier
VC  variance of correlation
WLAN  wireless local area networks
WMAN  wireless local and metropolitan area networks
1 INTRODUCTION

The Internet of Things (IoT) is the inter-networking of different physical devices and objects embedded with sensors, electronics, and even software, which reflects the constant development of the Internet from human to things since 1999 when the idea of combination of the RFID technology with sensor technology to set up a IoT was first proposed [5]. After this, IoT has received increasingly fast development in recent years to include wireless standards, i.e., WiFi, Bluetooth, and ZigBee, as its main networking technologies.

Bluetooth wireless technology was proposed in 1997 [5] and later standardized in IEEE 802.15.1 [6]. This networking technology provides easy wireless connection between devices in short distance, i.e., up to 10 meters, and with the bandwidth enabling transmission of audio, still image, and control signals. Bluetooth technology is very popular both in office and home environmental for pairing computer, phone, speaker, camera, printer, fitness equipment, etc. The concern on its security is receiving more and more attention due to its increasingly popular and wide uses in IoT.

ZigBee technology was first proposed in 1998 [7], one year later than Bluetooth. ZigBee provides wireless networking with greater distance and lower bandwidth and power consumption, compared to Bluetooth. ZigBee is very popular in IoT for smart home and healthcare applications. This technology is standardized by IEEE in 2003 as in 802.15.4 [7]. When used for healthcare for transmission of a patient’s private medical data, its security requirement and implementation is highly anticipated. Due to the fact that very simple devices are often used in a ZigBee network, any security consideration for these simple devices has to be with low computation complexity and requires low power consumption.
1.1 Research Motivation

ZigBee and Bluetooth technologies have wide applications in IoT, especially in smart office, smart home and smart healthcare, where data, often of a confidential nature, are transmitted over the wireless network. On the other hand, there are a number of security attacks that pose threats to the transmitted confidential data, for example, eavesdropping, impersonating, signal jamming, and violation of data integrity [8].

Eavesdropping is a scenario that an attacker secretly listening to the private conversation of others without their consent [8]. A security service called confidentiality is often employed to safeguard private data from eavesdropping attack. If a malicious third party attempts to illegally modify the transmitted data between the sender and the receiver, it is a case of violation of data integrity [9]. Impersonating is an very common threat that an attacker impersonates a legitimate user to send or receive data [9]. To prevent the system from either violation of data integrity attack or impersonating attack, we need the security service of authentication [10].

In order to provide security services like confidentiality and authentication, a security system must implement certain secure and efficient key establishment scheme to facilitate the subsequent encryption/decryption and/or authentication process. In this thesis, we focus on the research work on secure and efficient key establishment for ZigBee and Bluetooth systems.

1.2 Research Contributions

Research works and contributions in this thesis include two new ZigBee master key establishment protocols with enhanced security and one efficient key establishment protocol for Bluetooth.

- Proposed ZigBee master key establishment protocol with enhanced security using public key cryptography technology. To the best of our knowledge, it is
the first time that public key technology has been proposed for ZigBee master key establishment. The protocol not only implements key distribution between ZigBee parties but also provides strong authentication service to thwart possible impersonating attack.

- The second proposed ZigBee key establishment protocol is a modified version of the first one by trimming the authentication steps in the protocol. This simplified version is faster than the first one and suitable for being used between the authenticated ZigBee devices.

- A fast Bluetooth key establishment protocol is proposed, which is requires fewer protocol steps without sacrificing the security strength, compared to the similar work in [11].

1.3 The Scope and Organization of the Thesis

An organization of the rest of the thesis is as follows: in Chapter 2, the ZigBee and Bluetooth technologies will be discussed and their security requirements and public key cryptographic system RSA are reviewed. A modified version of RSA, unbalanced RSA is also introduced in this chapter. Then, the Chapter 3 will give an overview of the related existing works. New key establishment protocols are proposed in Chapter 4, along with the simulation results. Finally, Chapter 5 will provide the possible future works.
2 TECHNICAL BACKGROUNDS

In this chapter, the ZigBee and Bluetooth technologies and mathematical backgrounds will be introduced. This chapter has two parts, introduction to ZigBee and Bluetooth technologies and mathematical backgrounds.

2.1 ZigBee System and Security

2.1.1 ZigBee Network System Devices

For understanding the key establishment in the ZigBee system, it is important to understand the devices kinds in the ZigBee network. There always exist two different kinds of devices, Full-function device and Reduced-function device, in the ZigBee system and will be introduced in the next:

- **Full-function device (FFD)**: It means this kind device has the full-function calculation ability, which means the CPU and memory of this kind device is powerful to do complex calculations to support the major tasks in a ZigBee network system. In the ZigBee system, the nodes can be used for a wide variety of applications. Additionally, the FFD devices can perform any kind of applications defined by ZigBee standard in the MAC layer [12].

Furthermore, in the ZigBee system, except the FFD device, another kind is defined RFD (Reduced-function device).

- **Reduced-function device (RFD)**: RFD means the device calculation capacity is less than the FFD, which operates with a limited applications of the IEEE 802.15.4 MAC layer, enabling it to consume less power. Additionally, it also operates at the low power situation, and it only consumes power while transmitting data. Otherwise, the devices will fall in the sleep to save the power consumption. [12].
For getting a deep understanding of the ZigBee devices, there will be introduced the ZigBee devices in the next part:

- **Coordinator:** Coordinator is the corn of ZigBee network system and it must be the FFD device because it is responsible for the whole network to provide following functions: choosing the channel to be used by the network, starting the network, assigning how addresses are allocated to nodes or routers, permitting other devices to join or leave the network, holding a list of neighbors and routers and transmitting application packets [12].

- **Router:** Router is one kind of FFD device, too, which is used for forming or extending the network topology of the ZigBee system. Additionally, the router also has the responsibility to choose the best route path to transmit the data to the destination device. For the simple explanation, the function of router is similar of coordinator, except the router do not have the ability to set up the ZigBee network because it is the coordinator job [12].

- **ZigBee trust center:** In the ZigBee system, this device is used to provide ZigBee network security management, such as security key distribution and device authentication. In many ZigBee network system, the ZigBee trust center is combined with the coordinator together because the security question normally appears the network starting period [12].

- **ZigBee end device:** The ZigBee end device can be FFD or RFD, which is based on the function of the special end device. For ZigBee network system characteristic-low power consumption and low cost, the end devices typically fall into sleeping model except the data transmission period. Not only these, because the RFD end device does not have the enough calculation ability as the FFD, it can only communicate with FFD, not to RFD directly. In the ZigBee system, if one RFD wish to talk with another RFD, the communication should
be involved into routers to setup the shortest route path though other FFD
devices to help the connection with RFD to another RFD [12].

2.1.2 Three Kinds of ZigBee Network Topologies

Since the key establishment is the core security question in the ZigBee system, these
network topologies decide the key establishments methods in the ZigBee systems. For
understanding these, those kinds of topologies will be introduced in the following
part[13]:

- **Star topology**: Star is the basic model in the ZigBee system for understanding
  the star topology working rules, it will be shown in the following:

  ![Fig. 2.1: Star Topology in ZigBee system[1]](image)

  In the Fig. 2.1, this kind of topology is the basic model in the ZigBee network
  system, which is consisting only two kinds of devices, the coordinator and the
  end devices. In addition, this topology is the basic model of the ZigBee network
  system. In this model, the data transmission coming from the end devices is
  processed by the coordinator every time because, in this topology, the coordi-
  nator is the core point in this kind of network.

- **Cluster tree topology**: cluster tree is one update network topology of star
topology, which is shown in the following graph.
Fig. 2.2 indicates the relationship among the coordinator, routers and end devices in the cluster tree topology. In this network topology, the data path usually needs to involve the coordinator to help the data transmission. Additionally, the data communication among end devices is really depending on routers, which is used for data path connection among the independent end devices. Therefore, the routers in the network communication play a very important role since if one router meets any question, some end devices connected with the router will be lost connection in the network.

- **Mesh topology**: The following kind topology is defined mesh topology. In this kind topology, this is consisted by star topology. For getting understanding of this kind topology, the following graph is the mesh topology of the ZigBee system.
In the Fig. 2.3, there is only one coordinator existing in the mesh topology to control the ZigBee network system. There are many routers in this system to help with the data transmission in this mesh topology. Not only these, this topology is a multi-hop network. Additionally, since there are many routers in this topology, the data path from the source to destination is alternative during the data transmission; therefore, if one router path is blocked, the data transmission will be success because the other router path, maybe not the shortest one, can help with the data transmission. Furthermore, since the router number is bigger than in other topologies, the data transmission path is more effective than in other two kinds. However, since the mesh topology is more complex than star and tree topology, it needs more complex routing protocol than the other two.

2.2 Bluetooth and its Security

The Bluetooth network system devices is shown below. The computation capacity in the bluetooth is not the question in the network system because Bluetooth devices are not the devices with low computation capacity. In the Bluetooth system, the topology of the Fig.2.4 shows the master and slave construction.
In the Fig.2.4, the M donates master device, the S donates slave device. The differences between master and slave devices is shown in the following three aspects:

1. Master device can send data to any of its slaves and request data from them as well.

2. Slaves are only allowed to send and receive from their master device and they can not talk to other slaves in the piconet directly.

3. One piconet master device can play slave role in other piconets.

In the next chapter, the mathematical preliminaries will be introduced.

2.3 RSA and Unbalanced RSA

In cryptographic area, there are two kinds of individual key systems, symmetrical and asymmetrical key systems [8]. Detailedly, the encryption and decryption key of symmetrical key system are the same and this kind of key system is used for providing confidentiality and privacy security service usually. Conversely, the asymmetrical key system is used for key establishment and the delegates of this kind are RSA and ECC algorithms. The differences of these two key systems come from the features. The features of symmetrical key system are high efficient; simple algorithm; low cost and short key length. Therefore, it is suitable for encrypting and decrypting large data size. Another one is suitable for small size data transmission because the operation speed is slower and the key size is larger then the symmetrical key system, but the
security level of asymmetrical key system is higher than the symmetrical one [14].

In this chapter, the unbalanced RSA algorithm and RSA algorithm are introduced. Because the unbalanced RSA algorithm is another version of the RSA algorithm, the RSA algorithm will be shown first, then the unbalanced RSA algorithm is introduced in the next subsection. For the deep understanding of these two algorithms, the RSA and unbalanced RSA algorithms will be introduced and the generation of the RSA and unbalanced RSA algorithms parameters will be shown, too.

In the RSA algorithm, the first step is choosing two large prime numbers \( p \) and \( q \), then the multiplication of \( p \) and \( q \) will be calculated as the modulus \( n \). For making sure the RSA algorithm more effective, the minimum size of modulus \( n \) should be 1024 bits. Then, the user needs to choose the private key \( d \) and calculate the public key \( e \) through the private key \( d \) [15]. The size length of private key should be at least half of the length of \( n \) to guarantee the RSA algorithm parameters are secure. However, since the private key and public key can be replaced with each other in the RSA algorithm, the public key is normally using the number, which is 65537.

Additionally, these characteristics also appear in the unbalanced RSA algorithm since these parameters of unbalanced RSA are the same with RSA. The differences are the decryption part formula and the message length of unbalanced RSA is limited by the unbalanced RSA parameters. To get a deep understanding of the differences between the unbalanced RSA and RSA, these detailed algorithms are discussed in the following parts.

### 2.3.1 Development of RSA and Unbalanced RSA

RSA was proposed by Ron Rivest, Adi Shamir and Len Adleman in 1977, which is one of the first practical public-key cryptosystems and has been used widely in wire and wireless communication system security areas. Because they are from MIT, the name of the algorithm is the combination of the first letter from, Ron Rivest, Adi Shamir
and Len Adleman, these three proposers. Additionally, this kind of cryptographic technology is also used for digital signatures. This kind formulation also used a shared secret key created from exponentiation of some number and this is based on large prime number factorization to make sure the key system is secure. Not only these, only the short RSA key can be brute force attacked; therefore, the RSA algorithm has the high effect in current society and has been recommended as the public key data cryptographic standard by ISO. Furthermore, the RSA algorithm is based on the simple fact in number theory: two large prime multiplication is easy, but factorization of the multiplication result is hard [16].

After the publication of RSA, one publisher named Adi Shamir issued another version of RSA, named unbalanced RSA in 1995[16]. In the unbalanced RSA algorithm, the author found out if the message length is shorter than the two large prime number in the RSA system, the decryption part can be reducible. Because of this, the unbalanced RSA is more suitable for devices with low power or limited calculation ability than the RSA algorithm[16].

2.3.2 RSA Algorithm

RSA algorithm is based on the fact that two multiplication of large prime is easy to be calculated, but the factorization of multiplication of two large prime result into these two large prime number is hard. Specially, the RSA algorithm involves four steps: key generation, key distribution, encryption and decryption. The first step is the key generation. In this part, the RSA system should choose two distinct prime numbers, defined \( p \) and \( q \) [17]. For making the system secure, the prime numbers should be chose randomly and the length should be long enough to make factoring hard. Then, the next step is calculating the \( n \), which is the multiplication results of
and $q$ and this step can be expressed as follow:

$$n = p \times q$$  \hspace{1cm} (1)$$

Note that the multiplication result, $n$ is used as the modulus for the public and private keys; therefore, its length is also the key length. Additionally, the following job is about the calculation of $(p - 1) \times (q - 1)$. The expression is as follow:

$$\phi(n) = (p - 1) \times (q - 1)$$  \hspace{1cm} (2)$$

The value, $\phi n$, should be kept secure. After these, the next step is about the private key $d$ and public key $e$. At the beginning, an integer $e$ should be chosen for the public key; additionally, this $e$ must be in the range of $1 \leq e \leq \phi n$ and the $e$ should be co-prime with $\phi n$. After this, the public key, $d$, should be calculated and the formula of $d$ is shown as follow:

$$d \equiv e^{-1} (mod \phi n)$$  \hspace{1cm} (3)$$

The $d$ is the modular multiplicative inverse of the $e$. Until now, all RSA algorithm parameters are generated. The $e$ and $n$ are published by the system as the public parameters, but the other parameters, $d$, $p$ and $q$ are set as the private security information, only known by the owner [16]. After these, the RSA system operation will be introduced. At the beginning of the RSA system, the sender should obtain the public information $e$ and $n$ from the owner. Then, the sender can use the RSA system to communicate with the RSA system owner to protect the data transmission. For getting the easy understanding, the message coming from the sender will be defined $m$ and the ciphertext will be defined $c$. If the sender wishes to send the message to the owner, the encryption part is processed as the following step.

$$c \equiv m^e mod n$$  \hspace{1cm} (4)$
After this, the ciphertext $c$ will be sent to the RSA system owner. After the owner receives the ciphertext $c$, he/she will use the next formula to decrypt the ciphertext.

$$m \equiv c^d \mod n$$

(5)

Until now, the RSA algorithm introduction is finished. The next part is for the unbalanced RSA.

### 2.3.3 Unbalanced RSA Algorithm

In the parameters generation part, the steps and parameters are all the same with RSA algorithm. The difference is about the length of $p$ and $q$. In the [16], the length of $p$ is near ten times larger than the length of $p$. The reason will be shown in the next part. Therefore, these parameters in unbalanced RSA is using the same definitions with RSA algorithm. Additionally, the RSA and the unbalanced RSA encryption parts are the same. The different part, decryption part of unbalanced RSA is shown here:

$$m \equiv c^d \mod q$$

(6)

From the unbalanced RSA decryption part expression, it is obvious that the difference in decryption part of unbalanced RSA and RSA algorithm is the modular number; for RSA, it is the $n$, for unbalanced RSA, it is the $q$. The the proof of the unbalanced RSA is shown below, which uses the Chinese Remainder Theorem:

$$m \equiv c^d \mod n$$

(7)

$$m_1 \equiv c^d \mod p$$

(8)

$$m_2 \equiv c^d \mod q$$

(9)
\[ m_2 \equiv m \equiv m \mod q (m < q) \] (10)

\[ m \equiv m_2 \equiv c^d \mod q \] (11)

From the equations 7, 8, 9, 10 and 11, it is obvious that the unbalanced RSA can only be used while the message length is less than the length of \( q \). Because of this reason, the RSA algorithm can be used for the digital signature and other security service, but the unbalanced RSA algorithm is only used for the confidentiality. The detailed comparison summary is shown in the next subchapter.

### 2.4 Security Strength and Requirements

#### 2.4.1 Security Strength of Key Length

In the security area, the security level delegates the brute force time. Normally, different security algorithm has different security level. In the RSA and unbalanced RSA algorithms, the length of \( n \) delegates the security level. For getting an easy understanding of the relationship between security level with RSA algorithm, please see the next chart:

<table>
<thead>
<tr>
<th>Security Level (bits)</th>
<th>RSA modulus size (( n ))</th>
</tr>
</thead>
<tbody>
<tr>
<td>80</td>
<td>1024 bits</td>
</tr>
<tr>
<td>112</td>
<td>2048 bits</td>
</tr>
<tr>
<td>128</td>
<td>3072 bits</td>
</tr>
<tr>
<td>192</td>
<td>7680 bits</td>
</tr>
<tr>
<td>256</td>
<td>15360 bits</td>
</tr>
</tbody>
</table>

Table 2.1: Relationship of Security level and RSA modulus size

In the Tab. 2.1, the smallest security level is 80 bits, which is 1024 bits in the RSA and unbalanced RSA algorithm. In these five security levels, the 112 bits security level is security before 2030. Therefore, currently normal RSA and unbalanced RSA
algorithm is choosing 2048 bits as the \( n \) to set up the RSA or unbalanced RSA algorithm system.

### 2.4.2 Lightweight and Ultra-lightweight Security

Lightweight and ultra-lightweight security is used for some special areas to provide security services [18]. In these areas, because some special hardware reasons, the lightweight and ultra-lightweight security is applied. In this thesis, considering the ZigBee devices characteristics, small devices, minimal flash memory and low power consumption, the lightweight and ultra-lightweight security is suitable for ZigBee devices and network systems.

### 2.5 Comparison Between RSA and Unbalanced RSA

In current security area, the length of RSA is normally extended to 2048 bits to make sure the cryptographic system is secure. Then, the following table shows the difference between the RSA and unbalanced RSA.

<table>
<thead>
<tr>
<th></th>
<th>RSA ( e, n \text{ (i.e., } n \sim 2000 \text{ bits)} )</th>
<th>Unbalanced RSA ( p, q \text{ with } p \gg q ) (i.e., ( p \sim 1500)-bit, ( q \sim 500)-bit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Public parameters</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Private parameters</td>
<td>( p, q )</td>
<td>( p, q \text{ about same size) }</td>
</tr>
<tr>
<td>Encryption</td>
<td>( c = m^e \mod n \text{ (m: plaintext; c: ciphertext)} )</td>
<td></td>
</tr>
<tr>
<td>Decryption</td>
<td>( m = c^d \mod n )</td>
<td>( m = c^d \mod q )</td>
</tr>
<tr>
<td>Feature</td>
<td>Encryption and decryption have the similar complexity.</td>
<td>Decryption is much more efficient than encryption.</td>
</tr>
</tbody>
</table>

Table 2.2: Comparison between RSA and unbalanced RSA

The Tab. 2.2 has shown the differences between unbalanced RSA and RSA. It is obvious that the difference in the parameters is the private key length. In the RSA
algorithm, the lengths of private parameters $p$ and $q$ are similar. Contrary to this, the private parameter $p$ is longer than $q$. Additionally, the decryption part formula of unbalanced RSA is a special case of RSA since the RSA is modular $n$, but the unbalanced RSA decryption part is only modular $q$, which is the shorter parameter in the two private parameters. Because of this reason, the unbalanced RSA decryption part is more efficient than the RSA. However, the $n$ of RSA algorithm and unbalanced RSA algorithm should be the same since it determines the security level of these two algorithms; therefore, in the key generation part of unbalanced RSA, the length of $p$ is much larger than that of $q$.

2.5.1 Reasons of Choosing Unbalanced RSA

In the ZigBee system, there are FFD and RFD devices existing in the system. Normally, the end devices in the ZigBee system is only providing data transmission of small amount; therefore, the end devices normally are the RFD devices. Because of this reason, the calculation capacities between FFD and RFD are not the same, so the unbalanced RSA is suitable in the ZigBee system for the communication between FFD and RFD devices. Additionally, considering the decryption part of unbalanced RSA is more efficient than the encryption part, unbalanced RSA algorithm is a good choice in the ZigBee system. Furthermore, more efficient decryption part delegates low consumption and cost in the decryption part, so unbalanced RSA is suitable for lightweight and ultra-lightweight security, which is suitable for RFD devices.
3 AN OVERVIEW OF RELATED WORKS

3.1 Security Options for ZigBee and Bluetooth Standards

In now world wide ZigBee applications, the ZigBee security options have caused increasing attractions than before. Because of the development of ZigBee technology, three individual kinds of security options of ZigBee have been researched for many years and achieved some results[9]. The first one is called confidentiality, which is normally using symmetrical key systems to protect the data security in the transmission period. Specially, ZigBee technology runs the AES 128-bit algorithm in MAC layer to ensure system confidentiality[9]. Then, the next security option is authentication, which is used for checking the devices or messages identification to make sure there is no modification about the devices or the messages [19]. Additionally, the last part security question is about the key establishments, which is about these three kinds of keys that have been mentioned before. Then, in the related work part, the two important security options, authentication and key establishment, will be introduced in detailed and in the following parts, the basic knowledge of cryptographic technology will be given. In the Bluetooth security options, there are also three parts used for description of Bluetooth security, similar to the ZigBee security options. However, the authentication part of the Bluetooth technology is not the same because it is using the Challenge-response scheme to offer this service. The Key establishment method has three components: XOR algorithm, using XOR to protect the key establishment; Diffie-Hellman scheme using Diffie-Hellman scheme, a public-key cryptography system, to protect the key delivery; unbalanced RSA algorithm is using the unbalanced RSA algorithm, a public-key cryptography system, to protect the key transfer[9]. These details will be shown in this chapter on related work.
3.2 Key Structure in ZigBee and Bluetooth

There are three major keys existed in the ZigBee system. For understanding these three kinds of keys in ZigBee, the keys explanations are listed as follows [20]:

- **Link key**: Link key is a private key used for the communication between two devices, which is used for sending/receiving data individually. Because of this, the link key normally is a symmetric key. Likes the AES algorithm key, specially, the encrypt/decrypt keys are the same and the constructions are simple, so symmetric key is popularly used for large data encryption/decryption. However, since the encrypt and decrypt keys are the same, link key needs other kinds of keys to make sure the link key is secure [21].

- **Network key**: For protecting the network integration and security, network key is used widely in ZigBee system for authentication service for new joining devices. The network key is a common key shared among all nodes in a ZigBee system and is used for protecting the ZigBee network infrastructures and all nodes should be required to possess a valid Network Key in order to utilize the ZigBee network for transmitting and receiving data. Nodes without a valid Network Key should not be allowed to join (associate) or utilize a ZigBee network for transport. Routing nodes should validate ZigBee data packets based on the Network Key before processing and forwarding the packet [21].

- **Master key**: Master key is the base of network key and link key since these two kinds of keys are derived or generated from the master key. For the key protection, master key is normally stored in the core devices, like the coordinator or router. However, how to make sure the master key is secure is important since it is the foundation of these three kinds of keys.

The Bluetooth system is not similar with the ZigBee system because there is only one key existing in the system, which is defined-Link key. Because the Bluetooth
technology does not use the key to authenticate devices or key architectures, the link key is used for protecting the data transmission among devices; therefore, this key is important in the Bluetooth technology.

### 3.3 On Confidentiality in ZigBee and Bluetooth

As shown the Fig. 2.2, the security is the top hot topic in the ZigBee and Bluetooth systems. In the following subsections, the confidentiality will be explained. Confidentiality means the privacy in the security area, which delegates the protection of message security. It means although an attacker can obtain a message, he/she can not get the message information directly if the message has the confidentiality security. In the ZigBee system, the confidentiality is normally proposed for the data transmission between devices, from FFD to FFD or FFD to RFD [14]. The Bluetooth system is choosing this service to protect the data transmission, too. Because of this, the confidentiality service is using 128 bits AES to implement its security mechanisms. The security suite is 128 bits AES algorithm model.

The AES algorithm is applied in the ZigBee device. The message is encrypted and decrypted by the AES algorithm. Therefore, the 128 bits AES algorithm provides the confidentiality security service in ZigBee technology.

Additionally, because the AES algorithm is used in the Bluetooth technology, it only uses the link key to do the AES algorithm. From this section, the ZigBee and bluetooth system security will be introduced and the security service part in ZigBee stack is shown below [14]:
3.4 On ZigBee Authentication

Authentication is an important security service in the security area, which is used for providing the process of giving individuals access to system objects based on their identity. In the ZigBee technology, there are normally three different components in the authentication services: certificate authentication, message authentication code and network key. In the following subsections, these three methods will be shown [22].

3.4.1 Certificate Authenticaion

The certificate authentication in the security service is using the certificate to authenticate the device identity [22]. In the ZigBee network certificate authentication, these are five different roles in this kind of service: Global Virtual Certificate Authority (GVCA), Trust Center (TC), Manufacturer’s Certificate Authority (MCA), Manufacturer’s End Device (MED) and Manufacturer’s Virtual Certificate Authority (MVCA).
This method is illustrated with two authentication parts as follows:

**Fig. 3.2: Authentication protocol between TC and MCA**

1. Trust Center sends the authentication request to MCA before allowing the MCA to connect.
2. MCA verifies the certificate and sends a challenge to the TC.
   - TC decrypts the challenge and sends a response.
   - MCA verifies the response.
3. MCA sends an acknowledgment to TC.

**Fig. 3.3: Authentication protocol between TC and MED through MCA**

In the Fig. 3.2, during the first step of certificate authentication, the MCA should set up the connection with TC first. Before the protocol is started, all ZigBee devices are having the certificate from the GVCA to identify their identity. In this period, there are 11 steps. The steps are shown below:

1. Trust center sends the authentication request to MCA before allowing the MCA
to join this network.

2. MCA feeds back the TC asking for the certificate of TC to make sure the TC identity.

3. The TC feeds the certificate signed by GVCA to the MCA and ask the MCA certificate for the MCA identification.

4. After the MCA received the certificate from TC, the MCA verifies the certificate.

5. The MCA will ask the challenge response from TC to make sure the owner of the TC is true. Additionally, the MCA will send the MCA certificate signed by the GVCA back to the TC to help the TC to verify the MCA identification.

6. The TC will decrypt the challenge and send the response back to MCA to verify its identity.

7. The challenge from the TC to MCA is sent to MCA to help the TC to verify the MCA identity.

8. The MCA will verify the decryption result from TC and decrypted the challenge from the TC.

9. The MCA will feed the response back to TC.

10. The TC needs to verify the response from the MCA: if the response is the same with the stored one, the certificate authentication part is finished.

In the Fig. 3.3, there are eleven steps to set up the the authentication of MED:

1. The TC sends the authentication request to MED.

2. The MED sends the MCA authentication request to TC.

3. Because the TC and MCA have set up authentication before, the TC will send the MVCA certificate request to MCA.

4. The MCA will send the MVCA to TC because the TC and MCA have the authentication before.

5. After getting the MVCA from the MCA, the TC will send the MVCA certificate to MED.
6. After the MED authenticating the MCA, the MED will ask the certificate from the MCA to help the MED authenticate the TC.

7. The MCA sends the TC certificate to MED to help MED to set up authentication of TC.

8. After the MED authenticated the TC, the MED requests the MCA to help it join the existing network.

9. The MCA feeds back the Manufacturer’s End Device Certificate to MED.

10. The MED sends the Manufacturer’s End Device Certificate to TC to authenticate itself.

11. After the TC receiving the Manufacturer’s End Device Certificate from MCA, the TC should check this certificate. If the certificate authentication complete successfully, the MED will be allowed to join the existing network of TC.

12. The TC will tell the MCA that it is accepted to join the existing network.

After these two protocols, the MED device authentication is finished. This is the certificate authentication in the ZigBee technology. Then, other two kinds of authentication methods will be shown in the following sections.

3.4.2 Message Authentication Code

This message authentication code method is proposed by Suhas Kulkarni, Uttam Ghosh and Haribobu Pasupuleti. First the notation and abbreviation are shown as follows:
Table 3.1: Notation and Abbreviation in MAC

<table>
<thead>
<tr>
<th>Notations</th>
<th>Abbreviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>K</td>
<td>Common key</td>
</tr>
<tr>
<td>Ksd</td>
<td>Key shared between source and destination</td>
</tr>
<tr>
<td>I</td>
<td>Intermediate nodes between source and destination</td>
</tr>
<tr>
<td>RREQs</td>
<td>Route request of source</td>
</tr>
<tr>
<td>RREPs</td>
<td>Route reply of source</td>
</tr>
<tr>
<td>RREQi</td>
<td>Route request of intermediate node</td>
</tr>
<tr>
<td>REPi</td>
<td>Route reply of intermediate node</td>
</tr>
<tr>
<td>RREPD</td>
<td>Route reply of destination node</td>
</tr>
</tbody>
</table>

Then, the next graph shows the MAC authentication protocol in 14 steps.

The Fig. 3.4 is used for authentication between source node and destination node.
through some intermediate node in the route path. The 14 steps are shown in detail as below:

1. The source node should generate the $RREQs$, $MAC's$ and $MAC(sd)$. $MAC's$ and $MAC(sd)$ is generated as follows:

$$MAC's = H(RREQs, K)$$
$$MAC(sd) = H(RREQs, Ksd)$$

(12)

The $H$ denotes the hash function.

2. Then, after these three parameters are generated, they will be sent to the intermediate node from the source node.

3. After receiving these parameters, the intermediate node should generate $MACG$. If $MACG == MAC's$, the intermediate node should generate the $MAC'p$. Until now, the source node authentication is successful. Then, the $MACi$ is generated as following steps. These parameters are generated as next equations.

$$MACG = H(RREQs, K)$$
$$MAC'p = H(RREP, K)$$
$$MACi = H(RREQi, K)$$

(13)

4. Intermediate node feeds the $RREP$ and $MAC'p$ back to the source node.

5. Then, the source node needs to create the $RRCP$ and judge whether $MACP$ equals to $MAC'p$. If $MACP$ equals to $MAC'p$, the intermediate node authentication is successful.

6. The source node sends $RREQs$, route path, data, $MAC(sd)$ and $K(sd)$.

7. From this step, the intermediate node will set up the authentication to destination node. The intermediate node sends $RREQi, MACi$ and $MAC(sd)$.

8. After receiving these three parameters from step seven, the destination will
generate $MACGi$ to do the authentication, if authentication is successful, the $MAC’pi$ is created. Additionally, these two parameters is set as follow equations:

\[ MACGi = H(RREQi,K) \]
\[ MAC’pi = H(RREPi,K) \] (14)

9. Then, the destination will send these two parameters to intermediate node to do the destination node authentication.

10. The intermediate node needs to generate $MACPi$ in the equation 15 to judge whether $MACPi$ equals to $MAC’pi$ to authenticate the destination node.

\[ MACPi = H(RREPi,K) \] (15)

11. After the intermediate node finished the destination and intermediate nodes authentication, the $RREQs$, route path, data, $MAC(sd)$ and $K(sd)$ are sent from intermediate node to the destination node.

12. After the data transfer, the destination node generates the $MAC(sd’)$ to do the source node authentication. If this authentication is successful, the $MAC’D$ and $MACsd$ are generated to feed back to the source node to authenticate the destination. These three parameters are following the equation 17.

\[ MAC(sd’) = H(RREQs,Ksd) \]
\[ MAC’D = H(RREP_D,K) \] (16)
\[ MACsd = H(RREP_D,Ksd) \]

13. After the three parameters are generated at destination node, they are back to the source node.

14. After receiving these parameters coming from the destination node, the source node should create the $MAC’Dc$ to authentication the key($K$). If the key authenti-
cation is successful, the source node generates the $\text{MAC}_{\text{src}}$ to finish the destination node authentication. The generation parameters are calculated as follows [22]:

$$\text{MAC}'{D_c} = H(R\text{REPD}, K)$$

$$\text{MAC}_{\text{src}} = H(R\text{REPD}, K_{sd})$$

(17)

In this MAC protocol, the device and key authentication are involved. Additionally, this kind of MAC protocol is suitable for mesh network authentication because this protocol involves the route path and all working device authentication.

### 3.4.3 Network Key Authentication

This kind of network key authentication is much simpler than the two authentication protocols to set up the authentication between end device and trust center. In the fig. 3.5, the network key authentication is exhibited [22].

![Network key authentication](image)

Fig. 3.5: Network key authentication

There are four steps in the fig. 3.5.

1. The end device obtains the pre-installed Trust Center Link Key from the device Installation Code, $L Ki$ and derive the $TKi$ from the $LK_i$ (Key derivation function).

2. The end device will send some necessary information to trust center through the out of band method, such as web login and interaction through hand-held appliance.

3. After receiving these information coming from the end device, the trust center
will derive the $TK_i$ from the $LK_i$(Key derivation function).

4. The trust center will distribute the network key $NK$ encrypted by the $TK_i$.

After these four steps, the end device will use the decrypted $NK$ to join the existing network. Additionally, these three authentication methods are discussed in this chapter. In the next chapter, the key establishment methods will be shown.

3.5 Existing Works on ZigBee Key Management

Key establishment, also called key exchange, means the key transmission between two devices through an insecure communication channel [23]. This security service is allowing use of cryptographic algorithm. Because this security service is used for the key protection, the message is only the key; therefore, the message size is small and normally, for protecting its security, this service in ZigBee technology chooses the symmetric key establishment protocol for set up. In the ZigBee security, the key establishment has two types: group key protocols and symmetric key protocols. In the next parts, the two types are shown.

3.5.1 Group key protocols

This method is used for delivering individual keys to different devices to encrypt/decrypt the message, which consists of the key in the message [24]. In this method, the key establishment algorithm is using RSA algorithm in this protocol, which involves three kinds of devices: user, server and node. In the Fig. 3.6, the protocol steps is shown.
The special steps description are in the following:

1. The user, server and node are finishing their individual authentications and the server should generate the algorithm parameters for node key establishment and send the parameters to node. In this step, the node’s individual key is used for deriving the group key in the future steps.

2. The user designates the group about the group members.

3. The server will generate the group key, which is used for protecting the transmission of the delivered key.

4. After these three steps, the user needs to generate the delivered key as the message and send it to server, which is using the plaintext form.

5. The server will help the user to encrypt the message by using the node’s key establishment parameters.

6. After the step four, the server also needs to send the node sign to the special node to help the node to derive the individual node key.

7. The node will derive the group key through the node’s security key and node sign. After getting the group key, the node decrypts the message through the group key to get the delivered key. In the next section, the Symmetric Key establishments is shown.
3.5.2 Symmetric Key Establishments

This kind of key establishment is shown in the Fig.3.7 [25], like in previous sections, the description is shown after this graph. Additionally, the node’s symmetric key parameters has been shared with server before this key establishment protocol [25].

Fig. 3.7: Symmetric key protocol

1. Node should send the key establishment request to server for asking the key.
2. After receiving the request, the server generates the key as the data and uses the shared key known by node to encrypt the key.
3. The delivered key, message, is sent from the server to the node.
4. After receiving the message including the delivered key, the node will use the shared key with server to decrypt the message to obtain the key.
5. After getting the delivered key, the node needs to acknowledge the server that it has received the delivered key.

There are two methods on key establishments discussed in this chapter. Then, the next subchapters will show the master key establishments methods. The master key is the basic key of the ZigBee system key establishment. As explained before, there are two kinds of key establishments of the ZigBee system which is using the previously shared key. Therefore, the master key transmission is very important in the ZigBee system [26] [27]. In the ZigBee system, the existing master key transmission
in the ZigBee technology are realized in three methods. In the section before, the
two kinds of key establishments in ZigBee system are shown. However, these two
protocols are based on the shared keys acquired before the key establishments, so
these two protocols need shared keys to support these protocols working. Then, the
next chapter will talk about the three kinds of key transmissions of master key, which
are not introduced in this chapter because the master key is the basic key of ZigBee
system which does not have any pre-shared key before. Therefore, the three methods
of master key transports are introduced below: pre-installed, user-entered and key
transport.

3.5.3 Pre-installed Master Key

This method involves the manufacturer because the master key is installed by the
manufacturer in the manufacture period. The benefit is low cost for the ZigBee de-
vices [27]; therefore this method is popular in the master key transports. However,
because the master key is pre-installed, it is known by the manufacturer and stored
in the database of the manufacturer. Because of this, if the database of the manu-
facturer is cracked or broken by the attackers, all device master key will be known
by the attackers. Therefore, this method’s security has the high connection with
the manufacturer’s database security. Another disadvantage is the device delivery
period. Because the master key is installed into the device before the delivery period,
in the delivery period, it is the hands change during the delivery period. Because
of this, the master key can be modified or known in the delivery period [26]. After
considering these two disadvantages, the security level of this method, pre-installed,
is low [13] [21].
3.5.4 User-entered Master Key

The next method is user-entered. This method means the master key generation has a deep relationship with what the user entered into the system through the input devices. If only comparing with this method with pre-installed method, the security level of this one is much higher because this kind of master key is only known by the user of the ZigBee device and it is not eternal master key. In addition, this master key is initialized after the delivery period and is only known by the user [27]. However, this method is normally used for the FFD devices, like the coordinater or router because it needs the input device. Considering this reason, the end device is not suitable for using this method because the ZigBee devices typically need to be of small size and low cost [13]. The input devices restrict the size and cost of ZigBee devices; moreover, if the input devices are choosing the interfaces to reduce the size, the interfaces delegate the higher cost in the ZigBee devices. Therefore, this method is a great way to set up the master key but not the best choice.

3.5.5 Key-transport Master Key

After considering these two methods shown before, another suitable key establishment used in the ZigBee device is the key-transport. This method is based on the trust center, which is designed for the authentication or security service device. The benefit of this method is that transmitted master key is only known by the user or the owner; the disadvantage is this method involved the trust center, which is more complicated than the user-enter and the pre-installed methods [13]. This method is a great way to set up the master key; however, this approach has some disadvantages in the application period because the question is appearing in the transport period, on using the plaintext to transmit it or using the out-channel method to set up the master key. For the focuser, because the ZigBee technology appeared in a short time, for reducing the cost, the key transformation is currently choosing the plaintext. For the latter,
the out-channel method means this channel is unprotected and open [26]. Because of these existing master key transport methods having their own special disadvantages, the security level of master key establishment should be improved.

3.6 Recent Works on Bluetooth Security

For the Bluetooth link key key establishments, there are six existing methods and will be introduced in the following pages. XOR gate in Bluetooth technology is a popular method to protect the key establishment, but it is an easy way to protect the link key. This algorithm is used widely in Bluetooth technology and the functions will be shown [28].

3.6.1 Initialization Key Transmission

This transmission method is shown in the Fig.3.8 first and the details will be shown below [11]:

![Initialization key transmission protocol](image)

Fig. 3.8: Initialization key transmission protocol

1. The Bluetooth Device 1 and Bluetooth Device 2 need to enter PIN to set up the connection.

2. The Bluetooth Device 1 generates the $R_{init}$ randomly, then creates the $K_{init} = E_2(R_{init}, PIN, L)$. In this formula, the L delegates the length of PIN. In addition, it also generates $R_1$ randomly. After that, the $C_1$ is generated by the $R_1$ XOR
$K_{\text{init}}$. Additionally, the IDBD1- Identification of Bluetooth Device 1 is stored in the Bluetooth Device 1.

3. After these parameters generation, the $R_{\text{init}}$ is sent from the Bluetooth Device 1 to Bluetooth Device 2.

4. While the Bluetooth Device 2 receiving the $R_{\text{init}}$ from the step three, the Bluetooth Device 2 should create the $K_{\text{init}} = E_2(R_{\text{init}}, PIN, L)$ and $C_2 = R_2 \text{ XOR } K_{\text{init}}$, where $R_2$ is the random generation. Furthermore, the IDBD2- Identification of Bluetooth Device 2 is stored in the Bluetooth Device 2.

5. Additionally, the $C_1, C_2$, IDBD1 and IDBD2 will be exchanged between the Bluetooth Device 1 and Bluetooth Device 2.

6. Then, after these information exchanges, the Bluetooth Device 1 and Bluetooth Device 2 will figure out the following steps individually: for the Bluetooth Device 1, there are four steps existing in this step, $R_2 = C_2 \text{ XOR } K_{\text{init}}$, $K_1 = E_2(R_1, \text{IDBD1})$, $K_2 = E_2(R_2, \text{IDBD2})$ and $K_{1\cdot2} = K_1 \text{ XOR } K_2$. The next part in this step is for the Bluetooth Device 2. The steps are similar with Bluetooth Device 1, $R_2 = C_2 \text{ XOR } K_{\text{init}}$, $K_2 = E_2(R_2, \text{IDBD2})$, $K_1 = E_2(R_1, \text{IDBD1})$ and $K_{1\cdot2} = K_1 \text{ XOR } K_2$.

After these six steps, the shared keys are known by each other in the initialization key transmission and the $E_21$ and $E_22$ are custom key generator in the Bluetooth standard. The next key transmission protocol is the Unit key transmission protocol.

### 3.6.2 Unit Key Transmission

This key transmission protocol is using XOR algorithm to protect the key delivery by using the $K_{\text{init}}$ and this method is illustrated in the following graph. Additionally, the $K_{\text{init}}$ is shared before this transmission start with plaintext communication after PIN match between Bluetooth Device 1 and Bluetooth Device 2[11].
This is a simple key transmission. The security level is very low. Then, the next subsection is for the combination key transmission.

### 3.6.3 Combination Key Transmission

In this subsection, a new key transmission protocol is introduced, combination key transmission. As other key transmission protocols shown before, the Fig.3.10 describes this protocol first.

For a detail description, there are seven labels in used the Fig.3.10 and they will be shown first. $E_2$: Custom key generation algorithm in Bluetooth standard; $K_{	ext{init}}$: Initialization key; $K_{1,2}$: Combination key; $AK\_\text{RAND}_1$: Random number generated by Bluetooth Device 1; $AK\_\text{RAND}_2$: Random number generated by Bluetooth Device 2; $BD\_\text{ADDR}_1$: Address of Bluetooth Device 1; $BD\_\text{ADDR}_2$: Address of Bluetooth Device 2.
Device 2. Then, there are five steps in this protocol:

1. The Bluetooth Device 1 needs to generate the $AK_1 = E_{21}(AK_{RAND1}, BD_{ADD})$ and $C_1 = AK_{RAND1} \ XOR \ K_{INIT}$.

2. The Bluetooth Device 1 needs to send $C_1$ from step 1 to the Bluetooth Device 2.

3. After the Bluetooth Device 2 receiving the message from step 2, it will generate two parameters, $AK_2 = E_{21}(AK_{RAND2}, BD_{ADD})$ and $C_2 = AK_{RAND2} \ XOR \ K_{INIT}$.

4. The Bluetooth Device 2 will feed back the $C_2$ to Bluetooth Device 2.

5. This process has two parts, for the Bluetooth Device 1 and 2. These steps are processed at the same time. For the Bluetooth Device 1, the $AK_{RAND2} = C_2 \ XOR \ K_{INIT}$; $AK_2 = E_{21}(AK_{RAND2}, BD_{ADD})$ and $K_{1,2} = AK_1 \ XOR \ AK_2$. Additionally, for the Bluetooth Device 2, $AK_{RAND1} = C_1 \ XOR \ K_{INIT}$; $AK_1 = E_{21}(AK_{RAND1}, BD_{ADD})$ and $K_{1,2} = AK_1 \ XOR \ AK_2$.

After these five steps, the key has been known by each other, so the key delivery is finished.

### 3.6.4 Master Key Transmission

The simple key delivery protocol has three steps and this protocol is shown in the Fig.3.11.

![Fig. 3.11: Master key transmission protocol](image)
There are three steps to help the Bluetooth Device 1 to Bluetooth Device 2[11]:

1. The Bluetooth Device 1 should generate three parameters: \( K_{\text{master}} = E_{22}(\text{RAND}_1, \text{RAND}_2, L); \text{OVL}=E_{22}(K_{\text{link}}, \text{RAND}, L) \) and \( C = \text{OVL} \ XOR \ K_{\text{master}} \). In this step, the \( L \) delegates the length of PIN.

2. The Bluetooth Device 1 should send the \( \text{RAND} \) and \( C \) to the Bluetooth Device 2.

3. After these two steps, the Bluetooth Device 2 needs to generate \( \text{OVL}=E_{22}(K_{\text{link}}, \text{RAND}, L) \) to obtain the \( K_{\text{master}} \) through \( K_{\text{master}} = \text{OVL} \ XOR \ C \).

Through the step 1 to 3, the Bluetooth Device 2 can obtain the master key \( K_{\text{master}} \).

In the following subsection, the Diffie-Hellman scheme will be shown.

### 3.6.5 Diffie-Hellman Scheme

As the description before, the Diffie-Hellman scheme is shown in the Fig.3.12 with six processes.

Fig. 3.12: Diffie-Hellman scheme

The detailed steps in the Fig.3.12 will be shown below[11]:

1. Bluetooth Device 1 generates \( x = \text{DH Key and IDBD1} \). The IDBD1 is the Identification of Bluetooth device 1. Then, the Bluetooth Device 1 should computes the \( g^x \) as the public key. After these, the Bluetooth Device 1 also needs to generate the \( K \) randomly. Furthermore, the \( C \) is calculated through the function \( C = MAC(K, g^x) \)
in this step. After generating these parameters, the $x, g^x, C$ and IBD1 are stored in this device for the following steps.

2. The Bluetooth Device 1 should send these parameters in the step 1 to the Bluetooth Device 2.

3. The Bluetooth Device 2 needs to store the IBD2, Identification of Bluetooth Device 2, computes $C' = MAC(K, g^x)$. After these, if $C' = C$, the authentication of Bluetooth Device 1 is successful.

4. After the authentication of Bluetooth Device 1 is successful, the Bluetooth Device 2 should generate $y$ randomly as the DH key. Then, this device should computes the $S$ following the function $S = g^{xy} = DH$ shared. After these, the $K_{link} = KDF(PIN, S, C, K, IBD1, IBD2)$ and $E_k = KDF(K, IBD2)$ should be obtained in the Bluetooth Device 2.

5. In this step, the $g^y$, IBD2 and $E_k$ need to be sent to Bluetooth Device 1 from the Bluetooth Device 2.

6. After receiving these information from step 5, the Bluetooth Device 1 will compute $S = g^{xy} = DH$ shared, $K_{link} = KDF(PIN, S, C, K, IBD1, IBD2)$. Then, this device will decrypt the $E_k = KDF(S, IBD2)$. If the decrypted $K = K$, the $K_{link}$ will be accepted.

In this key establishment method, the KDF delegates a custom function of $K$ in the Bluetooth standard. Then, the last method, unbalanced RSA algorithm protocol is shown.

### 3.6.6 Unbalanced RSA Protocol

In the Fig.3.13, the unbalanced RSA algorithm in Bluetooth technology is shown.
In the Fig.3.13, there are seven steps displayed in the following descriptions between two Bluetooth devices.

1. The Bluetooth Device 1 should generate the $K$ randomly, then compute the $C = K^e \mod n$ and $H = \text{MAC}(K, C)$. After these computations, the $H, K, C, e$ and IDBD1 will be stored in the Bluetooth Device 1. At the same time, the Bluetooth Device 2 should store the IDBD2 itself for the next steps. Additionally, the Bluetooth Device 1 should broadcast public parameters of unbalanced RSA to the Bluetooth Device 2.

2. The Bluetooth Device 1 and 2 need to establish connection through the PIN entry: if the PIN authentication is successful, the connection is set up.

3. The Bluetooth Device 2 has to feed the IDBD2 back to Bluetooth Device 1.

4. After these three steps, the Bluetooth Device 1 will send $K, C, \text{IDBD1}$ and $H$ to Bluetooth Device 2.

5. After receiving these parameters from step 4, the Bluetooth Device 2 needs to decrypt the $C$ through $K = (C^d \mod p)$. Then, it needs to compute $H' = \text{MAC}(K, C)$. If $H' = H$, Bluetooth Device 1 authentication is successful. After these, the Bluetooth Device 2 will computes $K' = \text{Hash}(K)$, $H_R = (K' \text{ XOR } H' \text{ XOR IDBD2})$, $K_{\text{link}} = \text{HASH(PIN, K, H, IDBD1, IDBD2)}$.

6. The $H_R$ should be sent back to Bluetooth Device 1.
7. As the last step, the Bluetooth Device 1 will compute $K' = \text{HASH}(K)$, $K_{\text{link}} = \text{HASH} \left( \text{PIN}, K, H, \text{IDBD1}, \text{IDBD2} \right)$ and recover $K'$ from $H_R$. If the $K' = K'$, the $K_{\text{link}}$ transmission is successful.

After these six methods key establishments in the Bluetooth technology displayed, the more authentication and challenge-response scheme works will be shown in the following pages.

3.6.7 Authentication in Bluetooth

The authentication part in the Bluetooth technology has one method, Challenge-response scheme, which is described as following subsection [10].

3.6.8 Challenge-response Scheme

For the details of this scheme, the next figure, Fig.3.14 shows the protocol’s details.

![Fig. 3.14: Challenge-response scheme protocol](image)

In the Fig.3.14 [10], the link key is shared by Unit A and Unit B before the protocol is started. The Unit A is also defined Challenger and the Unit B is defined Claimant. Specially, there are six processes in the next part.

1. The Challenger, Unit A, needs to generate a random input, donated by the $AU_{\text{RAND}A}$.

2. The Challenger, Unit A, should send the $AU_{\text{RAND}A}$ to the Claimant, Unit B.
3. After receiving the $AU\_RAND_A$ from Challenger, Unit A, the Claimant, Unit B, should generate a random input, $AU\_RAND_B$.

4. The Claimant, Unit B, needs to create the $SRES_B$, by using $E1(AU\_RAND_A, AU\_RAND_B, LinkKey)$ to obtain the $SRES_B$.

5. The Claimant, Unit B, needs to feed back the $AU\_RAND_B$ and $SRES_B$ to the Challenger, Unit A.

6. The Challenger, Unit A, will generate the $SRES_A$ from $E1(AU\_RAND_A, AU\_RAND_B, LinkKey)$ and then the Challenger, Unit A, will compare with $SRES_B$. If the $SRES_A == SRES_B$, the authentication is successful.

Additionally, the E1 in the steps is a custom algorithm in the Bluetooth standard, which is used for the Bluetooth technology authentication.
4 PROPOSED WORKS

Because the ZigBee device needs the small size and low cost, the security requirements are not the same. Additionally, for making sure the master key establishment is secure, there are two individual methods of master key establishments explained in the following subsections [29].

4.1 New Master Key Establishment Protocol in ZigBee

In this part, one new method for master key establishment is proposed. For getting the details of this method, the Fig.4.1 indicates the special steps.

![Master key establishment protocol by unbalanced RSA algorithm](image)

Fig. 4.1: Master key establishment protocol by unbalanced RSA algorithm

This protocol is named the full version because it uses the unbalanced RSA algorithm to support the confidentiality service and the authentication service. there are nine steps existing in it.

1. End device provides its certificate or ID coming from the manufacturer and timestamps to help the coordinator/router to check its identity.

2. After receiving these messages coming from the end device, the coordinator/router check the certificate or ID and timestamps from the step one to authenticate the end device.
3. After authenticating the end device identity, the coordinator/router should feed back its certificate or ID and the timestamps to make sure the end device can identify the identity of the coordinator/router.

4. The end device should check the information from step three to make sure the authentication is successful. Until now, the mutual authentications are finished through the certificates and the timestamps.

5. End device share the individual unbalanced RSA public key parameters with coordinator or router.

6. After receiving these parameters, the coordinator/router needs to use these parameters to encrypt the master key to offer the confidentiality service.

7. After the encryption, the ciphertext will be sent back to end device by the coordinator/router.

8. The end device should decrypt the message coming from the coordinator/router to get the master key.

9. After the end device finishing the decryption to obtain the master key, the end device should acknowledge the coordinator/router that it has obtained the master key, which can help the device to join the existing network.

4.2 Simplify Master Key Establishment Security Protocol in ZigBee

After introducing the first version, which is involved in the authentication service, this simplified protocol is used for high security needs. As the same arrangements before, the Fig. 4.2 will indicate the steps in this protocol.
Fig. 4.2: Improvement master key establishment protocol by unbalanced RSA algorithm

From the Fig. 4.2, there are five steps in this protocol, which will be shown below. To simplify this protocol, the unbalanced RSA parameters are generated and stored by the end device.

1. End device shares the individual unbalanced RSA public key parameters with coordinator or router.

2. After the coordinator or router receiving these parameters, the coordinator/router needs to use these parameters to encrypt the master key to offer the confidentiality service.

3. After the encryption, the ciphertext will be sent back to end device by the coordinator/router.

4. The end device should decrypt the message coming from the coordinator/router to get the master key.

5. After the end device finished the decryption to obtain the master key, the end device should acknowledge the coordinator/router that it has obtained the master key, which can help the device to join the existing network.

In this simple version, the authentication part is not involved because this protocol is proposed to the ZigBee network system, where the authentication service has been
finished before. Additionally, the unbalanced RSA algorithm is helping for reducing
the calculation pressure to the ZigBee end device.

Until now, the two protocols are proposed and the difference between the first
one and the second one is the authentication steps. Therefore, for the master key
establishment parts, these two protocols are the same.

4.3 Simulation Results for RSA and Unbalanced RSA Algorithms

The simulation was performed to check and calculate how long it takes for the en-
cryption and decryption part for the master key establishment protocol. Simulation
results are performed on the unbalanced RSA algorithm and RSA algorithm. For
obtaining a clear and suitable simulation results, the simulation was performed for
modulus sizes of 1024 bits and 2048 bits [28]. In the unbalanced RSA algorithm, the
size of $p$ when $n$ is 1024 bits was 256 bits, making $q$ 768 bits. Additionally, while the
$n$ was 2048 bits, the size of $p$ was 512 bits, making $q$ 1536 bits. The parameters’ sizes
are decided by considering the security level. If any parameters’ sizes in the unbal-
anced RSA algorithm is small, it would be easy to crash the system. Therefore, the
parameters’ sizes should be in a suitable area. Moreover, the modular exponentiation
was performed using the square and multiply method to figure out this question as
it is one of the most popular methods to use. This simulation result is performed by
using JAVA on a laptop with an Intel(R) Core(TM)I5-6200U CPU @2.3GHz 2.4GHz.
For the details about this simulation, the code is attached in the appendices.

4.3.1 Simulation Results

The Tab.4.1 shows the simulation results of the RSA and the unbalanced RSA algo-
rithms. The particular running time is listed in the Tab. 4.1 will be discussed in the
following.
Table 4.1: Simulation Results for 1024 bits of the RSA and unbalanced RSA algorithms

<table>
<thead>
<tr>
<th></th>
<th>RSA</th>
<th>Unbalanced RSA</th>
</tr>
</thead>
<tbody>
<tr>
<td>(n(\text{bits}))</td>
<td>1024</td>
<td></td>
</tr>
<tr>
<td>(p(\text{bits}))</td>
<td>512</td>
<td>256</td>
</tr>
<tr>
<td>(q(\text{bits}))</td>
<td>512</td>
<td>768</td>
</tr>
<tr>
<td>Total time consumption (ms)</td>
<td>11.5</td>
<td>6.4</td>
</tr>
<tr>
<td>Time consumption for encryption (ms)</td>
<td>5.5</td>
<td>5.7</td>
</tr>
<tr>
<td>Time consumption for decryption (ms)</td>
<td>6.0</td>
<td>0.7</td>
</tr>
</tbody>
</table>

There are two difference tables for individual security needs because the ZigBee technology has a widely popular application in many different areas and the security needs are special in every area. Because the parameters in the Tab.4.1 and Tab.4.2 are not the same, the time consumptions are not the same so the analysis of the simulation results are in the subsection.
4.3.2 Analysis of Results

The simulation results in the Tab.4.1 show the time differences with Shamir’s discovery that the time delay of RSA computation grows with the size of the modulus. Additionally, if only focusing on the Tab.4.1, the total time consumption of unbalanced RSA algorithm is less than the RSA algorithm, it saves nearly 44.3% running time of the RSA algorithm. Additionally, the encryption part time consumption of RSA algorithm is similar with the unbalanced RSA algorithm, so the encryption part is following the A.Shamir’s theoretical results. Then, the decryption parts are significant that the unbalanced RSA algorithm is much faster than the RSA algorithm, which is only occupying nearly 10% decryption time of the RSA algorithm. Therefore, from the Tab.4.1, the A.Shamir’s theoretical results are obviously tested. Not only for the Tab.4.1, the Tab.4.2 is the same to show this aspect.

Then, after the comparison between the RSA and the unbalanced RSA algorithms simulation results in the two figures, more results can be obtained by comparing the two figures, Tab.4.1 and Tab.4.2. In these two tables, after increasing the parameters’ sizes from 1024 bits to 2048 bits, the time cost is not doubled with the doubled size. First, if we only focus on the RSA algorithm, the total time cost is from 11.5ms to 54.4ms: it increases nearly five times with the $n$ size increasing from 1024 to 2048 bits. However, for the unbalanced RSA algorithm parts, the addition all amount of time is the same, nearly five times, from 6.4ms to 32.3ms. Specially, for the encryption parts time cost, the RSA algorithm is rising from 5.5ms to 30.4ms. The unbalanced RSA of parts improvement is similar, five times, too, from 5.7ms to 30.2ms. Then, the last row of decryption parts are not the similar. The decryption time consumption of RSA algorithm is increasing four times, from 6.0ms to 24ms. Then, for the another column, unbalanced RSA algorithm, the decryption time cost is three times differences, from 0.7ms to 2.1ms. Therefore, if only comparing the last row of Tab.4.2, the decryption time cost for the difference between RSA and unbalanced RSA algorithm is also the
ten times, too; therefore, the unbalanced RSA algorithm is following the A. Shamir’s theory. Therefore, this algorithm can reduce the decryption computational capacity, so this algorithm is suitable for the two communication devices when these two devices do not have the same computational capacity.

4.4 Enhancement for the Unbalanced RSA Algorithm in Bluetooth

In the unbalanced RSA algorithm of Bluetooth technology, the step five has its special security weakness. In the step five, there is one function: \( H_R = (K' \ XOR \ H' \ XOR \ IDBD2) \). It is choosing the XOR algorithm to protect the \( K' \). The Vulnerability is appearing here. In this protocol, the \( H' == H' \) and the \( H \) and IDBD2 are plaintext transmission through the channel. Because of this, any attacker can obtain the \( H, H_R \) and IDBD2 in the transmission period. Then, while the connection is setting up successfully, this step shows the parameters are the correct and the \( H' == H' \) because only the \( H' == H' \) can make sure the communication success. Therefore, this XOR algorithm is redundancy in this protocol. However, if any attacker can obtain the \( K' \) in the transmission period, because the \( K' \) is the HASH function, the attacker can not recover the \( K \) from the \( K' \). Therefore, this computation step can be reduced. Then, the new protocol is proposed in the Fig.4.3.
Fig. 4.3: Enhancement key establishment protocol in Bluetooth
5 CONCLUSIONS AND FUTURE WORKS

5.1 Conclusions

In this thesis, the issue of secure and efficient key establishment in ZigBee and Bluetooth has been investigated. The proposed works include the following three new protocols.

- Two master key establishment protocols for ZigBee
- One master key establishment protocol for Bluetooth

The proposed protocols utilize the public key technology to provide highly secure key establishment for master keys and do not need the involvement of a third trusted party for key distribution. In order to overcome the barrier that public key technology usually requires communication devices equipped with high computational power, the new works take advantage of the unbalanced distribution of computation complexity of unbalanced RSA which make them suitable for the common scenarios in ZigBee and Bluetooth networks where one of the two communication parties (device) has relatively weaker computational capacity.

5.2 Possible Future Work

Based on the research results proposed in this thesis, the following two research directions are expected to have great potential for further investigation:

- Utilize unbalanced RSA to other wireless technologies where devices with limited computation capacity are involved.
- Investigate public key cryptographic technologies other than RSA for its suitability in the network where devices with limited computational capacity are used.
REFERENCES


VITA AUCTORIS

NAME: Chen Chen

PLACE OF BIRTH: Wuhu, China

YEAR OF BIRTH: 1990

EDUCATION: West Anhui University, Liuan, Anhui, China
Bachelor of Electrical Engineering, Communication Engineering 2009-2013

University of Windsor, Windsor ON, Canada
Master of Applied Science, Electrical and Computer Engineering 2015-2017