ENHANCING THE SECURITY OF RCIA ULTRA-LIGHTWEIGHT AUTHENTICATION PROTOCOL BY USING RANDOM NUMBER GENERATOR (RNG) TECHNIQUE

SHAYMAH AKRAM YASEAR

MASTER IN INFORMATION TECHNOLOGY (IT)
UNIVERSITI UTARA MALAYSIA
2015

Permission to Use

In presenting this thesis in fulfilment of the requirements for a postgraduate degree from Universiti Utara Malaysia, I agree that the Universiti Library may make it freely available for inspection. I further agree that permission for the copying of this thesis in any manner, in whole or in part, for scholarly purpose may be granted by my supervisor(s) or, in their absence, by the Dean of Awang Had Salleh Graduate School of Arts and Sciences. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to Universiti Utara Malaysia for any scholarly use which may be made of any material from my thesis.

Requests for permission to copy or to make other use of materials in this thesis, in whole or in part, should be addressed to:

Dean of Awang Had Salleh Graduate School of Arts and Sciences
UUM College of Arts and Sciences
Universiti Utara Malaysia
06010 UUM Sintok

Abstrak

Dengan permintaan yang semakin meningkat untuk sistem Pengenalpastian Frequensi Radio (RFID), terdapat keperluan untuk merekabentuk protokol pengesahan Pengenalpastian Frequensi Radio Ultra-ringan supaya ianya menjadi lebih serasi dengan sistem dan juga mampu bertahan terhadap kemungkinan serangan. Walaubagaimana pun, protokol pengesahan Pengenalpastian Frequensi Radio Ultra-ringan yang sedia ada amat terdedah kepada pelbagai serangan. Oleh itu, kajian ini adalah sebagai satu usaha untuk meningkatkan keselamatan protokol Kerahsiaan Teguh, Integriti, dan Pengesahan (RCIA) terutama yang berkaitan dengan isu-isu privasi. Dalam protokol RCIA, nilai *IDs* dihantar melalui pembaca dan *tag* sebagai nilai malar. Nilai malar ini membolehkan penyerang untuk mengesan lokasi *tag* yang akhirnya menceroboh privasi pengguna. Dalam usaha untuk meningkatkan keselamatan protokol RCIA, teknik Penjanaan Nombor Rawak (RNG) telah digunakan. Teknik ini bergantung kepada penjanaan nombor rawak di bahagian sebelah *tag*, menggunakan operasi *Bitwise*. Idea teknik ini adalah untuk menukar *IDs tag* pada setiap sesi pertanyaan supaya ia tidak akan kekal sebagai nilai malar. Pelaksanaan penambahbaikkan RCIA telah dilaksanakan dengan menggunakan teknik simulasi. Teknik simulasi ini menyediakan keupayaan untuk membandingkan operasi protokol RCIA sedia ada dengan inovasi RCIA yang baru. Hasilnya menunjukkan bahawa inovasi RCIA terbukti mampu mengatasi keupayaan sistem keselamatan yang sedia ada.

Kata kunci: RFID, ultra-ringan, protocol, nombor rawak, kemungkinan serangan

Abstract

With the growing demand for low-cost Radio Frequency Identification (RFID) system, there is a necessity to design RFID ultra-lightweight authentication protocols to be compatible with the system and also resistant against possible attacks. However, the existing ultra-lightweight authentication protocols are susceptible to wide range of attacks. This study is an attempt to enhance the security of Robust Confidentiality, Integrity, and Authentication (RCIA) ultra-lightweight authentication protocols especially with regard to privacy issue. In the RCIA protocol, IDs value is sent between reader and tag as a constant value. The constant value will enable attacker to trace the location of the tag which violates the privacy users. In order to enhance the security of RCIA protocol, Random Number Generator (RNG) technique has been used. This technique relies on generating random numbers in the tag side, based on Bitwise operations. The idea of this technique is to change the IDs of a tag on every query session so that it will not stay as a constant value. The implementation of Enhanced RCIA has been conducted by using a simulation. The simulation provided the ability to show that the operations of RCIA protocol as to compare with the enhanced RCIA. The outcome shows that the enhanced RCIA outperforms existing one in terms of privacy.

Keywords: RFID, ultra-lightweight, protocol, random number, traceability attack.

Acknowledgement

I would like to express my sincere thanks and appreciation to my supervisor Dr.Nur Haryani Zakaria for her guidance, support and the many helpful discussions throughout the course of this study.

I take this opportunity to express my gratitude to Dr.Mohd Nizam for his help in the preparation of this work.

I wish to thank my parents for their persistence encouragement and patience throughout. Special thank is also extended for my lovely sister Zainab for her encouragement, motivation and precious advices.

Shayma Akram

Table of Contents

List of Tables

List of Figures

LIST OF ABBREVIATIONS

AIDC	Automatic Identification and Data Capture
AVISPA	Automated Validation of Internet Security Protocols and Applications
CA	Certificate authority
CRC	Cyclic Redundancy Code
DoS	Denial-of-Service
EMAP	Efficient mutual authentication protocol
GA	Good approximations
IFF	Identify friend or foe
LMAP	Lightweight Mutual Authentication protocol
LSB	Least Significant Bit
MSB	Most Significant Bit
OCR	Optical Character Recognition
RAD	Rapid Application Development
RAPP	RFID authentication protocol with permutation
RCIA	Robust Confidentiality, Integrity, and Authentication
RFID	Radio Frequency Identification
RNG	Random Number Generator

Rn	Random Number
SASI	Strong Authentication and Strong Integrity
UHF	Ultra High Frequency
UMAP	Ultra-lightweight Mutual Authentication Protocol

CHAPTER ONE
INTRODUCTION

1.1. Introduction

This chapter will involve studying the background of Radio Frequency Identification (RFID) technology, additionally, the topics that will be covered are statement of the problem, research questions, research objectives, significance of research, and scope of the study.

1.2 Background of RFID Technology

The evolution of technology has contributed in reducing the gap between the physical and digital worlds [1]. One manifestation of this convergence is emerging a new technology that helps to identify objects automatically without the need for human intervention. This technology, called Automatic Identification and Data Capture (AIDC) or also known as "Auto-ID." This technology includes RFID, bar codes, magnetic stripes, Optical Character Recognition (OCR), voice recognition, biometrics, and smart cards. One of the most important relatively recent additions to Auto-ID technologies is RFID Technology. RFID is a communications technology that depends on radio waves to collect data automatically without the need for contact [2].

The origins of RFID technology dating back to the 19th century, which was during the Second World War when British Royal Air Force deployed "identify friend or foe" (IFF) system. This system was the first usage of RFID technology, which helped in distinguishing between the enemy and friendly aircraft [3]. In 1973, Mario Cardullo receives first U.S patent for an active RFID tag with rewritable memory. During the 1990s the emergence of standards and Ultra High Frequency (UHF) systems developed and patented [4].

The RFID system consist of tag or transponders, which wirelessly communicate with readers or transceivers [2], and back-end database stores items of tag. In RFID system the data is stored and retrieved remotely by using radio waves. The tag can be embedded in or attached to any object**,** and it has identification values, for instance a secret key and an identification number (ID) stored in its memory, and in the back-end database. The reader queries tag by sending radio frequency (RF) signals. The tag reflects the signal back to the reader using backscatter technology to transmit its identification values [5].

The use of RFID technology has become more widespread over the past decade. RFID technology has versatile applications, which the most prominent of these applications are in the area of manufacturing, supply chain management, smart card, bank note identification, pharmaceutics, and hospital equipment. In addition, it is used in different governmental and military institutions. Nevertheless, certain aspects of RFID technology have raised some privacy concerns, especially with regard to federal and commercial use [6]. The threats that are relevant to the RFID technology, represented by revealing personal information stored in the tag or in the associated database, and tracking a person or object location through a tag ID associated with that person or object. This information could be used to profile the preferences of the victim, movement, and social network [7].

The nature of communication in the RFID system makes it susceptible to a wide range of attacks. One of them is the attack that affects the communication channel between reader and tag. For example, an RFID tag cannot distinguish between authenticated reader and an illegal one. Therefore, the authentication protocols which were applied in this system are very important.

Depending on the level of complexity of the operations it carried out, the RFID classification of authentication protocols can be divided into four different classes. The first class full-fledged authentication protocol allows application classics cryptographic functions such as symmetric encryption or public and private key and one way hash functions. The second one, simple authentication protocol which supports the generation of random numbers and hash functions. The third category is a lightweight authentication protocol supports random number generator, simple functions such as Cyclic Redundancy Code (CRC) and simple bitwise operations (hash function is not included). The last one is ultra-lightweight authentication protocol can support simple bitwise operations (XOR, AND and OR) [8]**.**

The demand for produce low-cost RFID system caused limitations in its resources, especially in the RFID tag, which reflected negatively on the security of RFID system. Currently, several authentication protocols have been proposed most of these protocols are based on a hash function [1] [9]. However, with the limitation in RFD hardware resources (computing power and storage capacity) of wireless network environment, hash based authentication protocols cannot meet the requirements of practical application. This prompted researchers to develop new protocols commensurate with these limitations. This development included the RFID ultra-lightweight authentication protocol.

In this study, the problem under discussion is the privacy issue of ultra-lightweight authentication protocol. The cause of this issue is due to, the fixed messages that exchange between reader and tag. These fixed messages make it easy for any attacker to effect the RFID system by traceability attack.

In light of the above, this study will be centered to overcome this issue.

1.3. Problem Statement

Various RFID ultra-lightweight authentication protocols were proposed to enhance the security of RFID system, such as [7], [10], and [11]. The most recent ultra-lightweight authentication protocol is Robust Confidentiality, Integrity, and Authentication (RCIA) protocol, proposed by [12]. According to [12], RCIA protocol solved some of the weaknesses in the previous protocols such as desynchronization and full disclosure attack, by introducing and using a new ultra-lightweight primitive Recursive Hash function (Rh) [13]. Nevertheless, it still suffers from traceability attack which raised privacy issue.

In the RCIA protocol the authors claimed that RCIA resists against traceability attack since the messages (A, B, C, and D) combined with a random numbers (n_1 and n_2). In the RCIA protocol, the update operation for IDs value is performed only after each successful session. In this case, this update will prevent the attacker from tracking the tag with the assumption that the tag was read by legal reader.

Unfortunately, if the tag was read by illegal reader (i.e.: an attacker can pretend to be legitimate) traceability attack can happen in this scenario. In this case, when illegal reader sends a query to the tag, the attacker gets response from the tag by sending its IDs. In the next illegal query, the tag will send the same IDs which make it prone to traceability attack.

The main reason for this risk is when the illegal reader initiates a query to the tag, the responses of the tag each time are constant IDs. Therefore, RNG technique has been introduced to overcome this risk. This technique aims to make tag send different IDs value in each session. In this case the attacker will not be able to identify the same tag from query messages or interact with the tag to detect its location.

1.4. Research Questions

i. How to overcome the traceability attack on the RCIA ultra-lightweight authentication protocol?

ii. How to implement the RNG technique in the tag side?

iii. How to evaluate the RNG technique in preventing traceability attack?

1.5. Research Objectives

i. To generate variable IDs for the tag by adopting Random Number Generator (RNG) technique.

ii. To implement the RNG technique.

iii. To evaluate the RNG technique in preventing traceability attack.

1.6. Scope of the Study

The study would be limited to improving the security of a low-cost RFID system by enhancing ultra-lightweight authentication protocol, especially with regard to privacy issue. However, other categories of authentication protocols such as full-fledged, simple, and lightweight authentication protocol will not be covered in this study.

1.7. Significance of the Study

This study is an attempt to address the privacy issue of RFID ultra-lightweight authentication protocols, by introducing a technique called the Random Number Generator (RNG) to enhance the RCIA protocol. This in hope will contribute to enrich the research side on the effort of enhancing the security and overcome the threats that affected the RFID system.

Furthermore, nowadays RFID technology became a part and parcel of our life, whether at the level of individuals or institutions, where its application covered a wide array of different fields such as federal, commercial, health, education and many others. Preceding from the fact that the RFID system stores personal data, hence, the privacy of users is very important to be protected.

1.8. Summary

This chapter has presented a brief introduction about RFID technology. It highlights the problems of the research which is privacy issue of the latest ultra-lightweight RFID protocol known as RCIA. Furthermore, this chapter has entailed the objectives and scope of the research to be carried out. The significance of the study was also presented to highlight the contribution of this research. Next chapter will discuss in details the RFID ultra-lightweight authentication protocols and the attacks that affect these protocols.

CHAPTER TWO
LITERATURE REVIEW

2.1. Introduction

This chapter will provide a discussion of related work pertaining to RFID ultra-lightweight protocols. A brief explanation of RFID components will be provided to give a clear description about what RFID technology are and what are the limitations in the ultra-lightweight authentication protocol. Besides the protocols, attacks on RFID ultra-lightweight protocols were also presented and focus is given to the scope of the study that is the traceability attack.

2.2. Radio Frequency Identification (RFID) System

Radio frequency identification (RFID) is a non-contact automatic identification technology uses radio waves to achieve the object identification and data exchange. It is a great challenge to improve the security of RFID system, especially in providing secure and efficient, ultra-lightweight authentication protocols. As the name implies, these protocols cannot support strong cryptography algorithms with high level of requirements and specifications. For these reasons, these protocols are susceptible to security risks [8].

RFID system is consisting of three main components reader, tag, and back-end database. Figure 2.1 shows the components of RFID system.

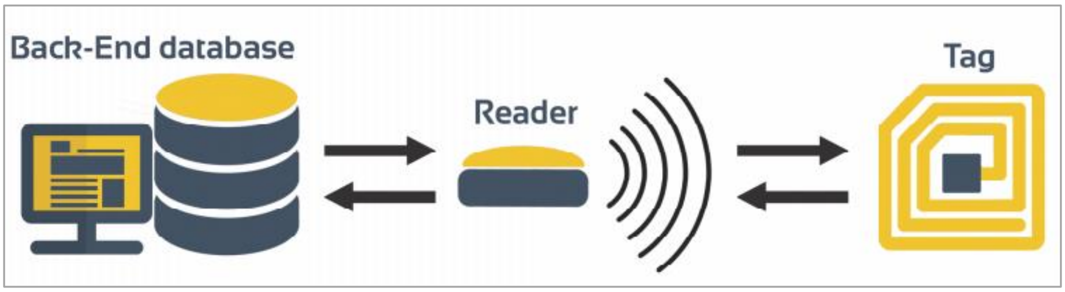


Figure 2.1. RFID System Components

All RFID items such as tag ID and secret keys are stored in the back-end database. The RFID reader, controls the communications (i.e.: read/write) with the tag. The RFID tag is an integrated chip connected to an antenna, and attached to the persons or objects. Generally the antenna coil is used for communication, and the microchip is used for storage and computation. Figure 2.2 illustrates the components of RFID tag.

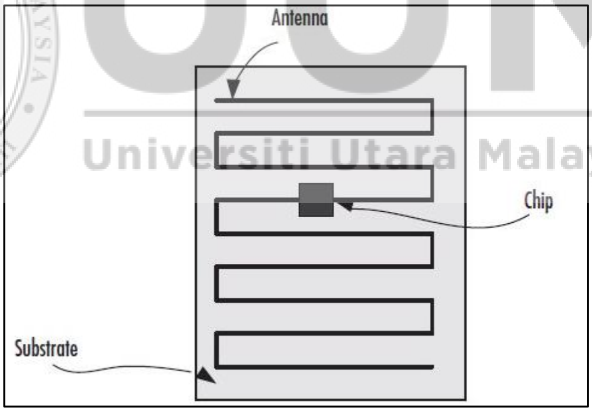


Figure 2.2. RFID tag components [2]

Since the RNG technique of this study will be implemented in the tag side, it is worth to provide more discussion on the types of tags to differentiate between them. In general, there are three types of tag which are, Passive tag, Semi-Passive tag and Active tag. For this study, passive tag will be used as it is commonly applied in ultra-lightweight category. Table 2.1 describes the types of tag.

Table 2.1

Type of RFID Tag [2]

	Passive tag	**Semi-Passive tag**	**Active tag**
Power source	Receives the power from the reader's signal	Has its own power source (battery)	Has its own power source (battery)
Communication	Initiated by the reader	Initiated by the reader	• Initiated by the reader • Initiate the communication by itself.

There are two factors that determine tag types, the communication initiation and the source of power. The passive tag does not have its own power, it receives the power from the reader's signal and the communication in this type initiated by the reader. While the semi-passive tag, has its own power source, but it cannot initiate the communication. Last type is active tag, this type has its own power source and it is able to initiate the communication and respond to the reader's signal.

This paper only focus on the passive tags, since the passive tags are totally powered by the signal of an interrogating reader and have become the mainstream of RFID applications [5].

2.3. Classification of RFID Authentication Protocols

In 2007, Chien [8] categorized RFID authentication protocols in four groups. Each group has its own characteristics. Figure 2.3 shows the classifications of RFID authentication protocols.

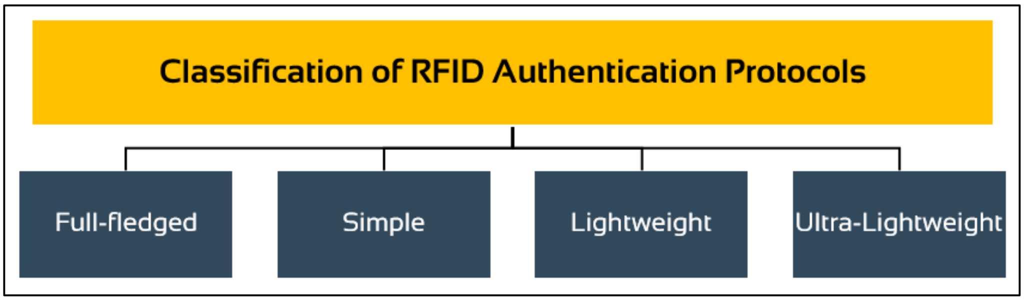


Figure 2.3. Classifications of RFID Authentication Protocols [8]

RFID authentication protocols classified according to their characteristics, as shown in Table 2.2.

Table 2.2

Characteristics of Ultra-Lightweight Authentication Protocols

Category	Characteristics
Full-Fledged Authentication Protocols	Support cryptographic functions like symmetric key encryption, one way hash functions and even public key cryptography, the protocols that fall under this category are [13], [14] and [15].
Simple Authentication Protocol	Refer to all protocols that uses a random number generator and one-way hash functions as an example for these protocols [16] and [17].
Lightweight Authentication Protocols	Refer to the protocols that only include simple function such as a CRC function instead of hash functions, for example [18], [19], [20] and [21] protocol.
Ultra-Lightweight Authentication Protocols	Include protocols that support simple bitwise operations like XOR, AND, OR, and rotation and modular addition, on the tag side. The protocols under this class are [7], [10] and [11].

The characteristics of RFID ultra-lightweight authentication protocols, as mentioned above, make it prone to several attacks. The following section will discuss the main attacks on these protocols.

2.5. Attacks on Ultra-Lightweight Authentication Protocols

In the RFID system, the communication channel between reader and tag is vulnerable to several types of attacks.

2.5.1. Traceability Attack

This attack closely related to privacy issue, where the attacker can threat the location of the tag's carrier. Since the tag uses constant values, when it communicates with the illegal reader, this will allow an attacker to track and detected the location of a person or an object for criminal reasons such as tagging of military assets [16]. For these reasons, it should be hard for any adversary to detect any tag that previously interacted with.

2.5.2. Impersonation Attack

In the impersonation attack, the attacker impersonates the identity of one of legitimate RFID system components (reader or tag) to gain its privileges.

2.5.3. Disclosure Attack

In this type of attack the attacker able to reveal the secret parameters of the tag and the reader, by using for example brute-force search. However, if the messages exchanged among reader and tag are not secure enough, then the attacker may uncover these secret parameters partially or completely.

2.5.4. Desynchronization Attack

Desynchronization happens when the reader and tag loss the synchronization between each other. In this attack, the attacker breaks the synchronization between the data stored in a reader and the tag that leads to the shared values update to new values in one side, while the others stay the same. In that case, the tag and the reader cannot recognize each other in future and tag becomes disabled [22]. Desynchronization attack can cause a tag to fall into a denial-of-service (DoS) attack. Where the tag becomes either temporarily or permanently incapacitated [23].

However, many authentication protocols proposed to increase the security of RFID system and to prevent these attacks. The next section will demonstrate these protocols.

2.6. Ultra-Lightweight Authentication Protocols

In this study, RNG technique aims to enhance the security of ultra-lightweight authentication protocol. Therefore, the related work introduced in this section, only focus on these protocols. Figure 2.4 illustrates ultra-lightweight authentication protocols.

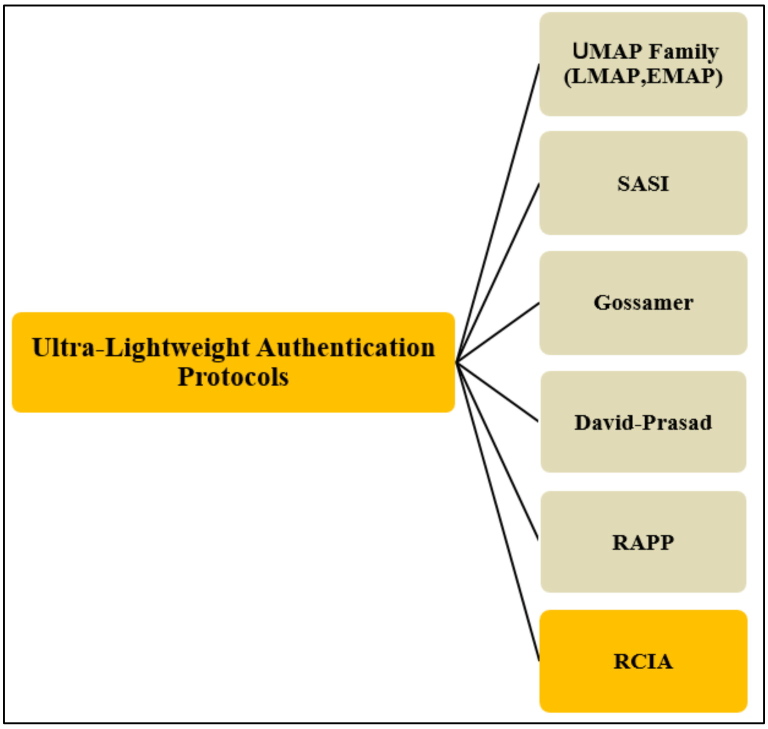


Figure 2.4. Ultra-Lightweight Authentication Protocols

Several studies reviewed and evaluated the security issues of RFID ultra-lightweight authentication protocols. In these studies, high level of vulnerabilities detected, these vulnerabilities include, common threats, for instance, desynchronization and DoS attack, in addition to, tracking the location of the tag.

Table 2.3

The Main Notation in the Ultra-lightweight Authenticaion Protocols

Notation	Description
K1, k2, k3, k4	Secret keys
ID	Tag ID
IDs	Pseudo IDs
n1, n2	random numbers
Per()	Permutation function

Rot()	Left Rlotation
⊕	Exclusive-OR
Rh()	Recursive hash function

In 2006, Peris-Lopez et al. proposed an Ultra-lightweight Mutual Authentication Protocol family (UMAP). This family includes two protocols which are: Lightweight Mutual Authentication Protocol (LMAP) [24] and Efficient Mutual Authentication Protocol (EMAP) [25].

LMAP protocol includes four main operations: Tag identification, mutual authentication, index-pseudonym updating and key updating. Tag contains a constant ID and five variable values IDs and four other keys K_1, K_2, K_3, K_4 that will be updated after each successful authentication session. The main operations of the protocol presented in Figure 2.5.

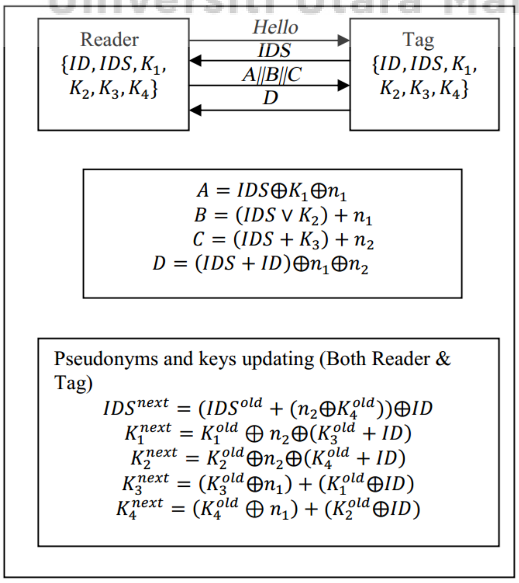

Reader
$\{ID, IDS, K_1, K_2, K_3, K_4\}$
Hello
IDS
A||B||C
D
Tag
$\{ID, IDS, K_1, K_2, K_3, K_4\}$
$A = IDS \oplus K_1 \oplus n_1$
$B = (IDS \vee K_2) + n_1$
$C = (IDS + K_3) + n_2$
$D = (IDS + ID) \oplus n_1 \oplus n_2$
Pseudonyms and keys updating (Both Reader & Tag)
$IDS^{next} = (IDS^{old} + (n_2 \oplus K_4^{old})) \oplus ID$
$K_1^{next} = K_1^{old} \oplus n_2 \oplus (K_3^{old} + ID)$
$K_2^{next} = K_2^{old} \oplus n_2 \oplus (K_4^{old} + ID)$
$K_3^{next} = (K_3^{old} \oplus n_1) + (K_1^{old} \oplus ID)$
$K_4^{next} = (K_4^{old} \oplus n_1) + (K_2^{old} \oplus ID)$

Figure 2.5. LMAP protocol [27]

In the LMAP protocol, the reader sends the 'Hello' message to the tag, and the tag responds with its IDS. The reader compares the received IDS with its IDs value if it matches with the IDs of tag, then the reader will generate two random numbers n_1 and n_2 and combines them with reader IDs and keys (K_1, K_2, K_3, K_4) to generate A, B and C messages. Reader concatenates and sends these messages to the tag. The tag will then extract $n_1$and n_2 from the messages and computes B using the same equation, and compares it with received B if a match occurs, it means the tag communicates with a legal reader and then tag will update its values (IDs, K_1, K_2, K_3 and K_4). After that, tag will calculate and send D message to the reader. The reader will generate and compare D with D messages sent by tag, if match found, the reader will authenticate the tag, and also both will update the values (IDs, K_1, K_2, K_3 and K_4).

The operations in the EMAP protocol are quite similar to LMAP protocol, except that, in the EMAP a new Parity function (Fp) was added, which is introduced, as vector built from the parity. EMAP protocol operation illustrated in Figure 2.6.

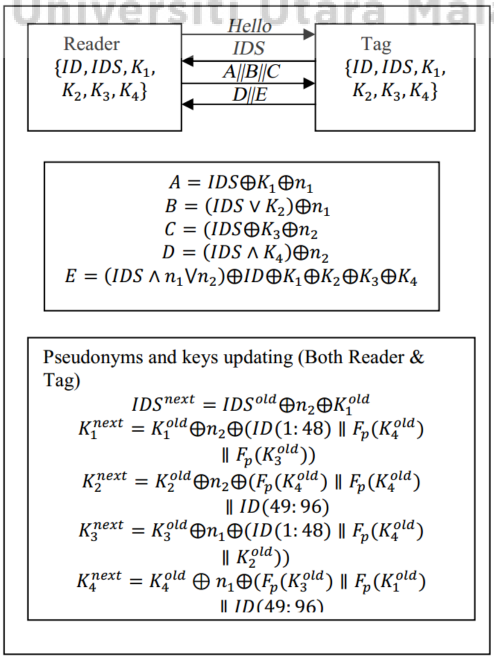

Reader
$\{ID, IDS, K_1, K_2, K_3, K_4\}$
Hello
IDS
A||B||C
D||E
Tag
$\{ID, IDS, K_1, K_2, K_3, K_4\}$
$A = IDS \oplus K_1 \oplus n_1$
$B = (IDS \vee K_2) \oplus n_1$
$C = (IDS \oplus K_3 \oplus n_2$
$D = (IDS \wedge K_4) \oplus n_2$
$E = (IDS \wedge n_1 \vee n_2) \oplus ID \oplus K_1 \oplus K_2 \oplus K_3 \oplus K_4$
Pseudonyms and keys updating (Both Reader & Tag)
$IDS^{next} = IDS^{old} \oplus n_2 \oplus K_1^{old}$
$K_1^{next} = K_1^{old} \oplus n_2 \oplus (ID(1:48) \parallel F_p(K_4^{old}) \parallel F_p(K_3^{old}))$
$K_2^{next} = K_2^{old} \oplus n_2 \oplus (F_p(K_4^{old}) \parallel F_p(K_4^{old}) \parallel ID(49:96)$
$K_3^{next} = K_3^{old} \oplus n_1 \oplus (ID(1:48) \parallel F_p(K_4^{old}) \parallel K_2^{old}))$
$K_4^{next} = K_4^{old} \oplus n_1 \oplus (F_p(K_3^{old}) \parallel F_p(K_1^{old}) \parallel ID(49:96)$

Figure 2.6. EMAP Protocol [27]

The analysis of UMAP family, pointed out that the protocols vulnerable to malicious attacks. In 2008, Li and Wang [26] proposed two attacks (desynchronization and full disclosure) on LMAP and EMAP and successfully refute security claims of both protocols. In these protocols, since the previous IDs value it does not store in the reader. If the attacker interrupts the communication among reader and tag, and block D message, the tag will update its values while the reader will not and it will remain use its previous values. In this case, in the next query from reader to tag, the tag will respond with its current IDs which is quite different from the IDs stored in the reader. As a result of that, the tag will become useless. And this mean, the UMAP protocols cannot prevent desynchronization and disclosure attack. Furthermore, UMAP can neither resist disclosure nor de-synchronization, cannot resist traceability attack. In UMAP, since the eavesdropper can pretend to be legitimate reader, when the reader sends a query to the tag, the eavesdropper gets the response with IDs. In the next query, when the legitimate reader sends a request, the tag will responds with same IDs, so that UMAP cannot resist traceability attacks.

In 2007, Chien [8] proposed the Strong Authentication and Strong Integrity (SASI) protocol. This protocol reported in [27], [28] and [29], their findings provide confirmatory evidence that SASI has many vulnerabilities such as desynchronization and secret disclosure attacks. In 2011 [29], a successful desynchronization attack was shown on SASI protocol. Thus, in 2013, Avoine et al. [30] propose a successful passive full-disclosure attack. Figure 2.7 summarized the operations of SASI protocol.

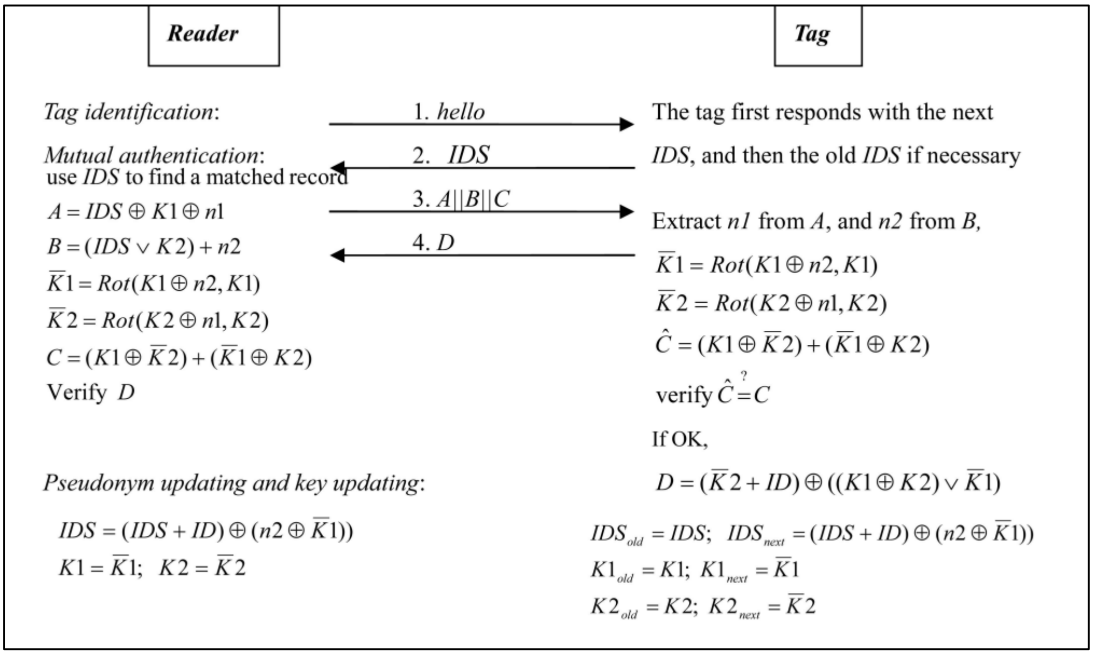

Reader
Tag
Tag identification:
1. hello
The tag first responds with the next
Mutual authentication:
use IDS to find a matched record
2. IDS
IDS, and then the old IDS if necessary
3. A||B||C
$A = IDS \oplus K1 \oplus n1$
Extract n1 from A, and n2 from B,
$B = (IDS \vee K2) + n2$
4. D
$\overline{K}1 = Rot(K1 \oplus n2, K1)$
$\overline{K}2 = Rot(K2 \oplus n1, K2)$
$C = (K1 \oplus \overline{K}2) + (\overline{K}1 \oplus K2)$
Verify D
$\overline{K}1 = Rot(K1 \oplus n2, K1)$
$\overline{K}2 = Rot(K2 \oplus n1, K2)$
$\hat{C} = (K1 \oplus \overline{K}2) + (\overline{K}1 \oplus K2)$
verify $\hat{C} \overset{?}{=} C$
If OK,
$D = (\overline{K}2 + ID) \oplus ((K1 \oplus K2) \vee \overline{K}1)$
Pseudonym updating and key updating:
$IDS = (IDS + ID) \oplus (n2 \oplus \overline{K}1))$
$K1 = \overline{K}1; \quad K2 = \overline{K}2$
$IDS_{old} = IDS; \quad IDS_{next} = (IDS + ID) \oplus (n2 \oplus \overline{K}1))$
$K1_{old} = K1; \; K1_{next} = \overline{K}1$
$K2_{old} = K2; \; K2_{next} = \overline{K}2$

Figure 2.7. SASI protocol [8]

In SASI protocol, the tag responds with IDs. When the reader receives IDs from the tag, it will calculate and send A, B and C messages. To attack a protocol, the attacker can block D message. In this case, the reader did not receive a D message so, it will not able to update its stored values but the tag will do.

In 2009, Peris-Lopez et al. [7] proposed a new ultra-lightweight authentication protocol called (Gossamer) protocol. This protocol proposed as an extension to SASI protocol to overcome its weakness. [11], [31]. Although this protocol shown resistance to a passive full disclosure attack, nevertheless, the desynchronization and denial of Service (DOS) attacks still exist in this protocol [32, 33]. The operations of Gossamer protocol is similar to other previously proposed protocols, except that, in Gossamer, they add two new functions; Double Rotation and MixBits [7]. In 2012, Zubair et al. [34] improved the performance of Gossamer protocol by proposed a counter based methodology. Combination this counter in Gossamer protocol makes it resilient against DOS and desynchronization attacks.

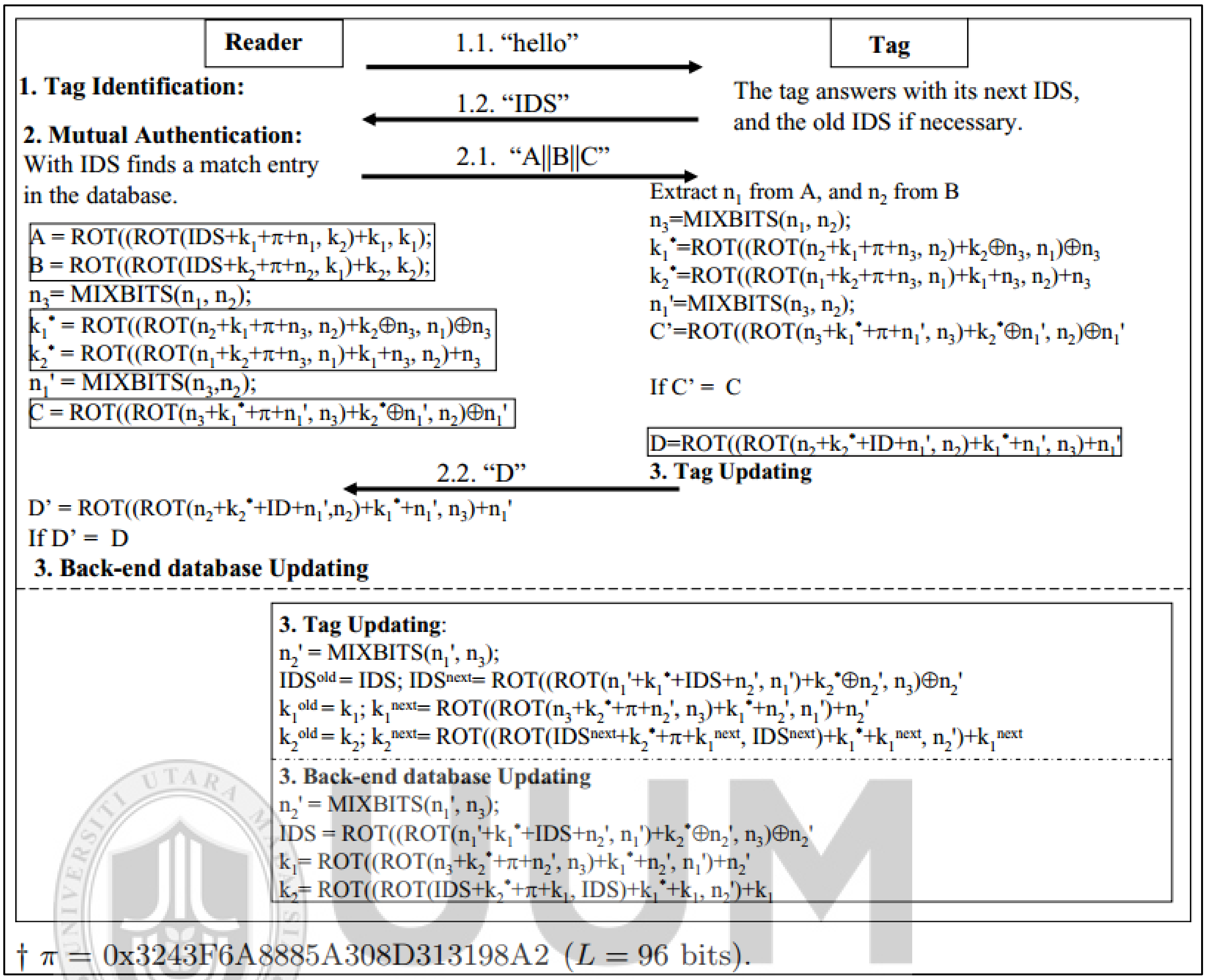

Reader
1.1. “hello”
Tag
1. Tag Identification:
1.2. “IDS”
The tag answers with its next IDS, and the old IDS if necessary.
2. Mutual Authentication:
With IDS finds a match entry in the database.
2.1. “A||B||C”
Extract n_1 from A, and n_2 from B
$A = ROT((ROT(IDS+k_1+\pi+n_1, k_2)+k_1, k_1);$
$B = ROT((ROT(IDS+k_2+\pi+n_2, k_1)+k_2, k_2);$
$n_3= MIXBITS(n_1, n_2);$
$k_1^* = ROT((ROT(n_2+k_1+\pi+n_3, n_2)+k_2\oplus n_3, n_1)\oplus n_3$
$k_2^* = ROT((ROT(n_1+k_2+\pi+n_3, n_1)+k_1+n_3, n_2)+n_3$
$n_1' = MIXBITS(n_3,n_2);$
$C = ROT((ROT(n_3+k_1^*+\pi+n_1', n_3)+k_2^*\oplus n_1', n_2)\oplus n_1'$
$n_3=MIXBITS(n_1, n_2);$
$k_1^*=ROT((ROT(n_2+k_1+\pi+n_3, n_2)+k_2\oplus n_3, n_1)\oplus n_3$
$k_2^*=ROT((ROT(n_1+k_2+\pi+n_3, n_1)+k_1+n_3, n_2)+n_3$
$n_1'=MIXBITS(n_3, n_2);$
$C'=ROT((ROT(n_3+k_1^*+\pi+n_1', n_3)+k_2^*\oplus n_1', n_2)\oplus n_1'$
If C’ = C
$D=ROT((ROT(n_2+k_2^*+ID+n_1', n_2)+k_1^*+n_1', n_3)+n_1'$
3. Tag Updating
2.2. “D”
$D' = ROT((ROT(n_2+k_2^*+ID+n_1',n_2)+k_1^*+n_1', n_3)+n_1'$
If D’ = D
3. Back-end database Updating
3. Tag Updating:
$n_2' = MIXBITS(n_1', n_3);$
$IDS^{old}= IDS; IDS^{next}= ROT((ROT(n_1'+k_1^*+IDS+n_2', n_1')+k_2^*\oplus n_2', n_3)\oplus n_2'$
$k_1^{old}= k_1; k_1^{next}= ROT((ROT(n_3+k_2^*+\pi+n_2', n_3)+k_1^*+n_2', n_1')+n_2'$
$k_2^{old}= k_2; k_2^{next}= ROT((ROT(IDS^{next}+k_2^*+\pi+k_1^{next}, IDS^{next})+k_1^*+k_1^{next}, n_2')+k_1^{next}$
3. Back-end database Updating
$n_2' = MIXBITS(n_1', n_3);$
$IDS = ROT((ROT(n_1'+k_1^*+IDS+n_2', n_1')+k_2^*\oplus n_2', n_3)\oplus n_2'$
$k_1= ROT((ROT(n_3+k_2^*+\pi+n_2', n_3)+k_1^*+n_2', n_1')+n_2'$
$k_2= ROT((ROT(IDS+k_2^*+\pi+k_1, IDS)+k_1^*+k_1, n_2')+k_1$
† $\pi = 0x3243F6A8885A308D313198A2$ ($L = 96$ bits).

Figure 2.8. Gossamer protocol [7]

In 2009, David and Prasad [10] presented a new ultra-lightweight authentication protocol based on bitwise operations. This protocol uses only two Bitwise logical operations AND and XOR, which contributed to reduce computational power at tag side. In David-Prasad protocol, reader needs to get one-day certificate from CA (Certificate authority) before inquiring the tag. Reader initiates the protocol by sending “Hello” message to the tag. Tag then responds with its current IDs, reader matches this IDs with IDs stored in the back-end database; if a match found, it will produce two random numbers (n1, n2), calculate and send (A, B and D) to the tag, the summary operations of David-Prasad Protocol are as in Figure 2.9.

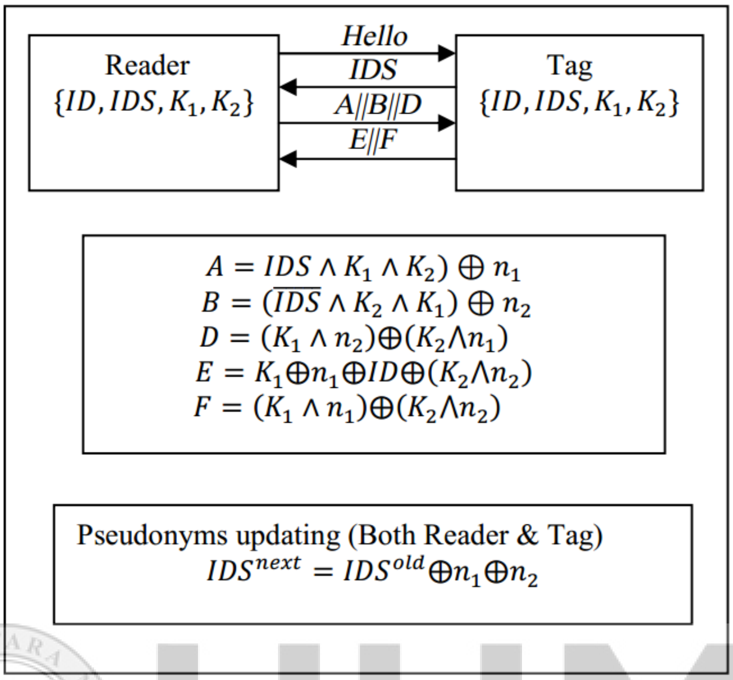

Reader
$\{ID, IDS, K_1, K_2\}$
Hello
IDS
$A||B||D$
$E||F$
Tag
$\{ID, IDS, K_1, K_2\}$
$A = IDS \wedge K_1 \wedge K_2) \oplus n_1$
$B = (\overline{IDS} \wedge K_2 \wedge K_1) \oplus n_2$
$D = (K_1 \wedge n_2) \oplus (K_2 \wedge n_1)$
$E = K_1 \oplus n_1 \oplus ID \oplus (K_2 \wedge n_2)$
$F = (K_1 \wedge n_1) \oplus (K_2 \wedge n_2)$
Pseudonyms updating (Both Reader & Tag)
$IDS^{next} = IDS^{old} \oplus n_1 \oplus n_2$

Figure 2.9. David-Prasad Protocol [36]

However, in 2010, Hernandez-Castro et al. [35] proposed full disclosure attack (Tango) on the David-Prasad protocol. Tango attack requires GA (good approximations) equations based on hamming distance with unknown variable. Later on, Barrero et al. in [36] presented genetic tango attack to improve Tango attack and later on, resolved the exhaustive searching of GA equations.

In 2012, a new ultra-lightweight authentication protocol called RFID authentication protocol with permutation (RAPP) proposed [11]. Unlike previous protocols, this protocol relied on the new technique. In this protocol the tag has the ability to perform three simple functions: Bitwise XOR operation, left rotation Rot(), and the [11] function Per(), and all these functions are cheap to implement in the tag. In order to compute new string from X and Y, using permutation function (Per(X, Y)), the bits position of second string will be shifted and copied based on the bits of the first string. Two pointers, (pX) and pY), are set to string X and string Y respectively. During the movement of the pointers, if the bit of (pY) is 1, the bit of (pX) will be copied into the new string. Since X and Y are of the same length, (pX) and (pY) will reach Least Significant Bit (LSB) at the same time. Then they will change the movement direction and go back to Most Significant Bit (MSB). During the return, if the bit (pY) is 0, the bit of (pX) will be copied into the third string. The permutation process will go to an end when (pX) and (pY) both return to MSB and the final result will be stored in the third string. Figure 2.10. Illustrates the process in the permutation function Per().

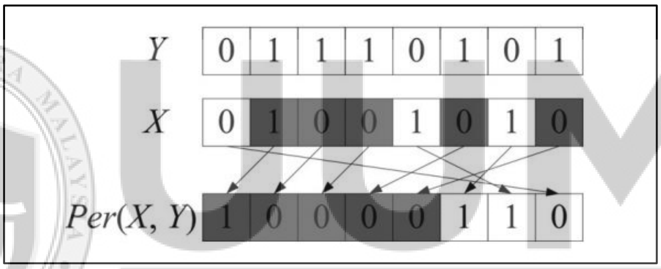


Figure 2.10. Permutation function [11]

In RAPP Protocol, each tag has an ID, secret keys (K_1, K_2 and K_3) and IDs shared with the backend database, and will be updated at the end of a successful protocol run. Figure 2.101 illustrates the details of the messages exchanged in this protocol.

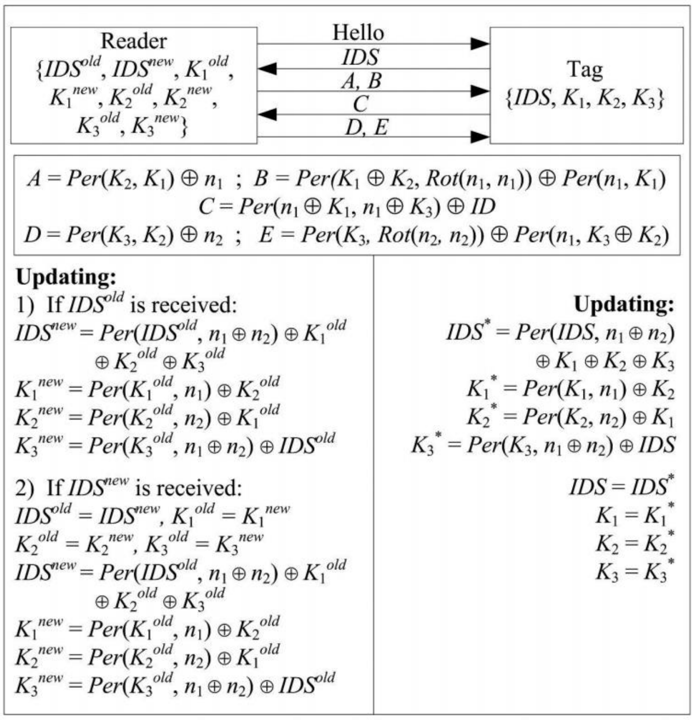

Reader
$\{IDS^{old}, IDS^{new}, K_1^{old}, K_1^{new}, K_2^{old}, K_2^{new}, K_3^{old}, K_3^{new}\}$
Hello
IDS
A, B
C
D, E
Tag
$\{IDS, K_1, K_2, K_3\}$
$A = Per(K_2, K_1) \oplus n_1$; $B = Per(K_1 \oplus K_2, Rot(n_1, n_1)) \oplus Per(n_1, K_1)$
$C = Per(n_1 \oplus K_1, n_1 \oplus K_3) \oplus ID$
$D = Per(K_3, K_2) \oplus n_2$; $E = Per(K_3, Rot(n_2, n_2)) \oplus Per(n_1, K_3 \oplus K_2)$
Updating:
1) If IDS^{old} is received:
$IDS^{new} = Per(IDS^{old}, n_1 \oplus n_2) \oplus K_1^{old} \oplus K_2^{old} \oplus K_3^{old}$
$K_1^{new} = Per(K_1^{old}, n_1) \oplus K_2^{old}$
$K_2^{new} = Per(K_2^{old}, n_2) \oplus K_1^{old}$
$K_3^{new} = Per(K_3^{old}, n_1 \oplus n_2) \oplus IDS^{old}$
2) If IDS^{new} is received:
$IDS^{old} = IDS^{new}, K_1^{old} = K_1^{new}$
$K_2^{old} = K_2^{new}, K_3^{old} = K_3^{new}$
$IDS^{new} = Per(IDS^{old}, n_1 \oplus n_2) \oplus K_1^{old} \oplus K_2^{old} \oplus K_3^{old}$
$K_1^{new} = Per(K_1^{old}, n_1) \oplus K_2^{old}$
$K_2^{new} = Per(K_2^{old}, n_2) \oplus K_1^{old}$
$K_3^{new} = Per(K_3^{old}, n_1 \oplus n_2) \oplus IDS^{old}$
Updating:
$IDS^* = Per(IDS, n_1 \oplus n_2) \oplus K_1 \oplus K_2 \oplus K_3$
$K_1^* = Per(K_1, n_1) \oplus K_2$
$K_2^* = Per(K_2, n_2) \oplus K_1$
$K_3^* = Per(K_3, n_1 \oplus n_2) \oplus IDS$
$IDS = IDS^*$
$K_1 = K_1^*$
$K_2 = K_2^*$
$K_3 = K_3^*$

Figure 2.11. RAPP Protocol [11]

However, in 2012, [37] and [38] highlighted two attacks on RAPP, desynchronization and traceability attack. Avoine [37] indicated that, the protocol RAPP- contrary to the claim of its designers -prone to desynchronization attack. In 2013, Ahmadian et al. [39], proposed a desynchronization attack on this protocol and highlighted the poor composition of RAPP messages. In the same year, Shao-hui et al. [38] highlighted some weaknesses of the newly proposed permutation function [32], which can be easily exploited to uncover secrets in the tag.

In 2015, robust confidentiality, integrity, and authentication (RCIA) protocol proposed by [12]. This protocol uses bitwise operations (AND, XOR) and left rotation Rot(A,B), in addition to a Recursive Hash function Rh(), which is used in this protocol for the first time.

The computation of a recursive hash of A, Rh(A), includes three steps.

(1) Decimate the string A into “K” number of chunks (memory blocks) with an equal number of bits “l” per memory block (K = n/l).

(2) Calculates the seed (index of the memory block) for recursive hash in following manner.

a. Compute R = n1$\oplus$ n2.

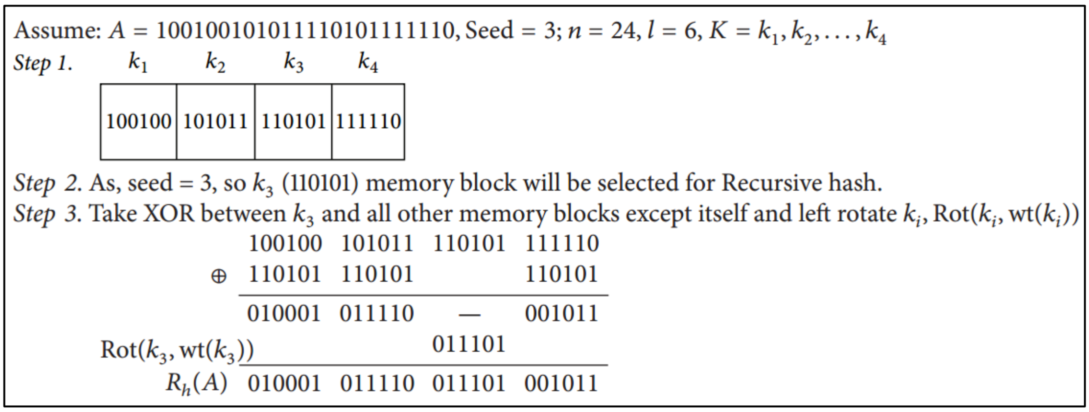
b. Seed (index of memory block) = wt(R) mod K, where wt(R) is the hamming weight of R.

(3) Use a seed to select the corresponding memory block (K_i) of decimated string A and perform the following operations to compute final recursive:

a. Take XOR between selected memory block (Ki) with all other blocks except the block itself.

b. Left rotate the K_i with itself: Rot(K_i, wt(K_i)).

Figure 2.12. shows the process of recursive hash function Rh().



Assume: $A = 100100101011110101111110$, Seed $= 3$; $n = 24, l = 6, K = k_1, k_2, \ldots, k_4$

Step 1.

k_1	k_2	k_3	k_4
100100	101011	110101	111110

Step 2. As, seed = 3, so k_3 (110101) memory block will be selected for Recursive hash.

Step 3. Take XOR between k_3 and all other memory blocks except itself and left rotate k_i, Rot(k_i, wt(k_i))

	100100	101011	110101	111110
$\oplus$	110101	110101		110101
	010001	011110	—	001011
Rot(k_3, wt(k_3))			011101	
$R_h(A)$	010001	011110	011101	001011

Figure 2.12. Recursive hash function Rh() [12]

Although RCIA protocol was able to solve some of the weaknesses in the previous protocols, but it still has a vulnerability that related to privacy issue. To clarify this issue, it has to be described the flow of operations the RCIA protocol. The main operations will be as following:

Step (1). Reader: sends a "Hello" message to the tag.

Step (2). Tag: responds with its "IDs"

Step (3). Reader: receives IDs, and compares it with the IDs in the database.

b. If it is an old one, IDs,$_{old}$, then the reader will use $K_{1,old}$ and $K_{2,old}$ for computation of messages (A, B, C).

c. If IDs is a new one (IDs,$_{new}$), then reader will use $K_{1,new}$ and $K_{2,new}$ to compute messages (A, B, C).

d. If IDs is not in the database, then the reader will immediately terminate the authentication session with the particular tag.

If a match is found in the database, reader will generate two random numbers (n_1 and n_2) and concatenate them with A and B. The reader will also compute ($R = n_1 \oplus n_2$), and seed for computation of recursive hash function. The seed is computed by taking hamming weight of R mod K(wt(R) mod K).

Step (4). Reader: Uses recursive hash function (R_h) of the variables (K_1, K_2, K_1^*, K_2^*, n_1, n_2) to compute "C" message.

Step (5). Tag: extracts random number (n_1 and n_2) from messages A and B. Then it computes the seed for recursive hash function using $R= n_1 \oplus n_2$ and wt(R) mod K. The tag further calculates K_1^* and K_2^* to compute the local value of C^* and compare it with the received C.

If both the values are equal, then the tag will perform two tasks:

a. Calculate and transmit D message towards reader.

b. Update (ID_S) and keys (K_1 and K_2).

Step (6). Reader: receives message D, and computes a local value of D and compares them together, if a match occurs, then the reader will also update (IDs) and keys (K_1 and K_2) for the tag. The operations of RCIA protocol described in Figure 2.13.

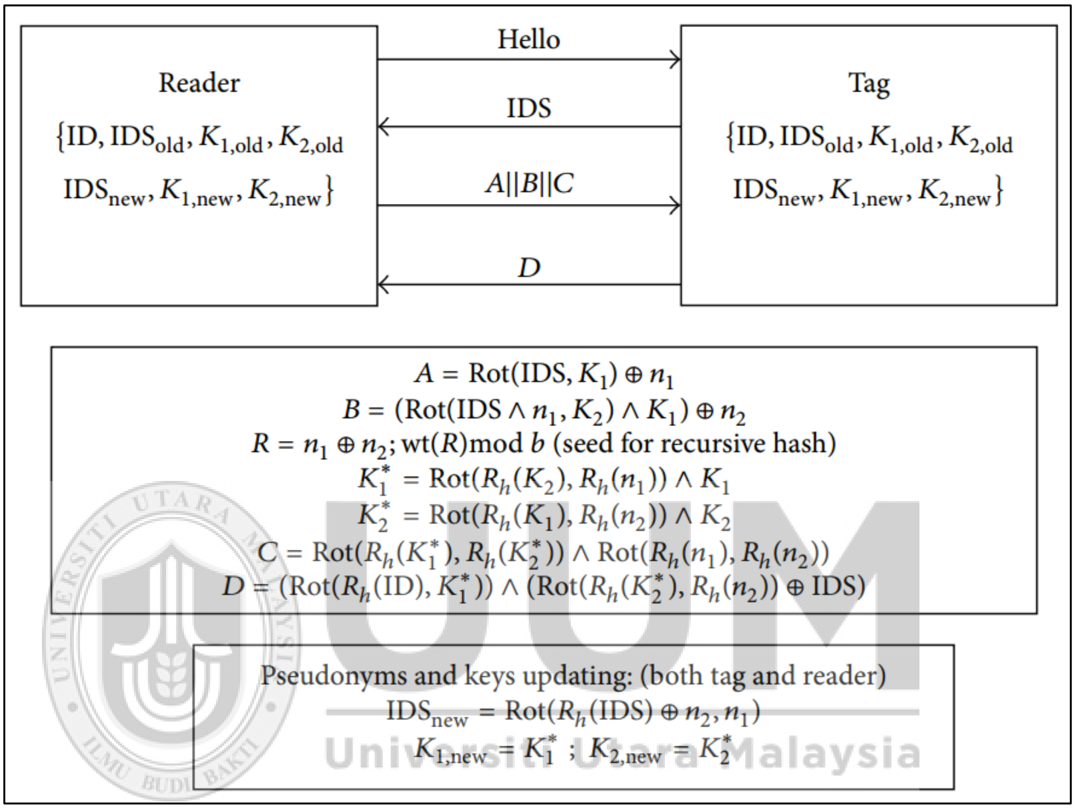

Reader
$\{ID, IDS_{old}, K_{1,old}, K_{2,old}$
$IDS_{new}, K_{1,new}, K_{2,new}\}$
Hello
IDS
$A||B||C$
D
Tag
$\{ID, IDS_{old}, K_{1,old}, K_{2,old}$
$IDS_{new}, K_{1,new}, K_{2,new}\}$
$A = \mathrm{Rot}(IDS, K_1) \oplus n_1$
$B = (\mathrm{Rot}(IDS \wedge n_1, K_2) \wedge K_1) \oplus n_2$
$R = n_1 \oplus n_2$; $\mathrm{wt}(R) \bmod b$ (seed for recursive hash)
$K_1^* = \mathrm{Rot}(R_h(K_2), R_h(n_1)) \wedge K_1$
$K_2^* = \mathrm{Rot}(R_h(K_1), R_h(n_2)) \wedge K_2$
$C = \mathrm{Rot}(R_h(K_1^*), R_h(K_2^*)) \wedge \mathrm{Rot}(R_h(n_1), R_h(n_2))$
$D = (\mathrm{Rot}(R_h(ID), K_1^*)) \wedge (\mathrm{Rot}(R_h(K_2^*), R_h(n_2)) \oplus IDS)$
Pseudonyms and keys updating: (both tag and reader)
$IDS_{new} = \mathrm{Rot}(R_h(IDS) \oplus n_2, n_1)$
$K_{1,new} = K_1^*$; $K_{2,new} = K_2^*$

Figure 2.13. RCIA Protocol [12]

In the second step of RICA protocol, it can be clearly seen that the (IDs) value of the tag sent as a fixed value, and this value cannot updated only by legitimate reader [12], as shown in the fifth step. As a result of that, the attacker can easily track the location of tag's carrier by sending a multi query to the tag and the tag will response by sending the same IDs to the illegitimate reader. These reasons pointed out that the RCIA protocol vulnerable to traceability attack. Figure 2.14 illustrate the operations of RCIA protocol.

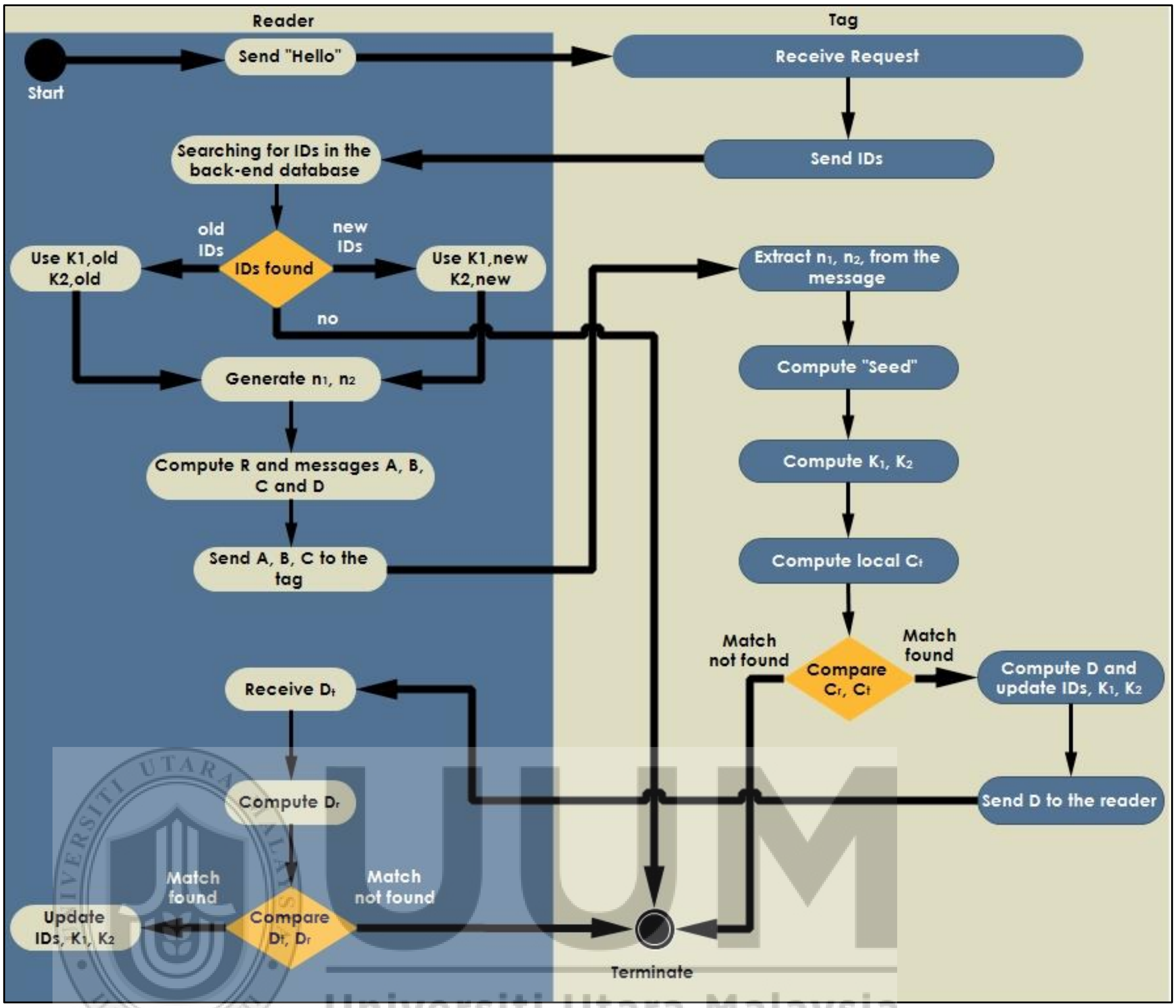


Figure 2.14. Operations of RCIA Protocol [12]

The next Table 2.5 shows a simple comparison of main attacks resistance, between the most recent ultra-lightweight authentication protocols.

Table 2.4
Attacks Resistance Comparison between ultra-lightweight authentication Protocols

	Traceability Attack	Desynchronization Attack	Disclosure Attack
UMAP family	X	X	X
SASI	X	X	X
Gossamer	X	X	√
David-Prasad	X	√	X
RAPP	X	X	√
RCIA	X	√	√

X : Susceptible to attack.
√: Resists such an attack.

While some of these protocols shows a resistance against desynchronization and disclosure attacks, nevertheless, these protocols unable to prevent traceability attack. Through studying the operations of previous protocols, it was observed that the IDs sent as a constant value from the tag to the reader. The value of IDs does not change until complete the session successfully. In this case, it is logically to say that the existing protocols are prone to traceability attack. Solving this issue can be done by using Random Number Generator (RNG). According to the Chien's classifications [8] the tag side in the ultra-lightweight authentication protocols, cannot generate random numbers using normal methods that used in the full-fledged and simple authentication protocols. Nevertheless, since the ultra-lightweight authentication protocols support Bitwise operations, it is possible to use them to generate random numbers and thus, prevent traceability attack.

2.7. Summary

This chapter discussed the existing ultra-lightweight authentication protocols focusing on the most recent one, RCIA: protocol. Studying the operations of RCIA protocol showed that it is vulnerable to traceability attack which leads to privacy issue. The next chapter will elaborate further on the methodology that will be carried out through this study.

CHAPTER THREE
RESEARCH METHODOLOGY

3.1. Introduction

This study aims to address the privacy issue in the ultra-lightweight authentication protocols. This chapter briefly describes the phases that had been performed to achieve the research objectives. In this context, to achieve these objectives, it is appropriate to organize this study into phases; each phase is allocated to each objective.

The first phase involved identifying the problem of the RCIA ultra-lightweight authentication protocol. This is followed by the second phase, which was allocated for conceptual design and implementation of the RNG technique using a prototype. The evaluation of the enhanced RCIA was conducted in the last phase. The following sub-sections will elaborate further on each phase. Next Figure 3.3 summaries all phases involved in this study.

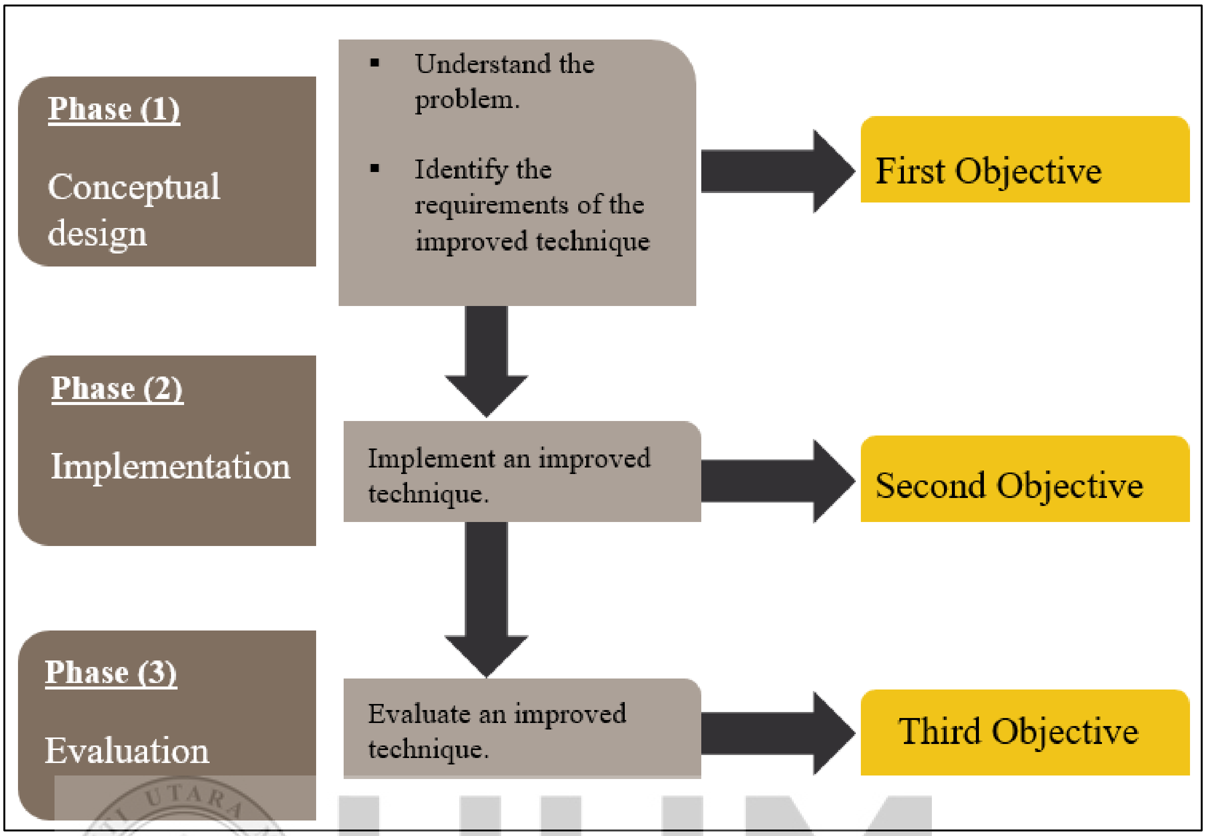


Figure 3.1. Phases of The Study

3.2. Phase 1 – Conceptual Design

The first phase was allocated for a conceptual design of the RNG technique. At this stage, understanding the problem was very important to make sure the RNG technique is able to be implemented.

The conceptual design, involved simple algorithms based on Bitwise operations such as XOR-Shift [40]. This algorithm takes into account constrained resources of ultra-lightweight authentication protocols and can be used with devices with limited resources. The algorithm was used to generate a random number in the tag side. In this case the messages that send by the tag to the reader will be inconstant. This will prevent the attacker from tracking the tag, thus, preventing traceability attack.

The completion of this phase achieved partial of the first objective. Figure 3.1 summaries the Enhanced RCIA.

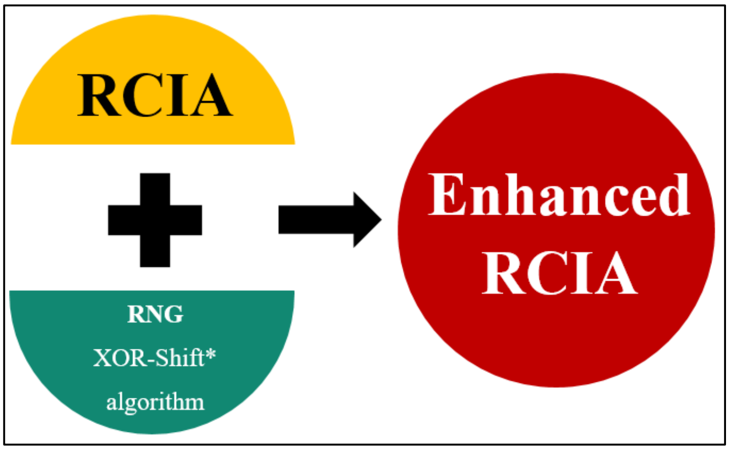


Figure 3.2. Summary of Enhanced RCIA

3.3. Phase 2 – Implementation of RNG

In this phase, RNG technique was implemented in a prototype form. The prototype was developed instead of using actual hardware components of an RFID system (reader and tag) due to limitation of financial and time constraint.

The methodology that has been used to develop the prototype is Rapid Application Development (RAD). This methodology uses rapid construction of prototype instead of large amounts of planning. The planning of software is interleaved with writing the software itself. This allows software to be written much faster, and makes it easier to change requirements. In the next stage, requirements are verified using prototyping, eventually to refine the data and process models.

The prototyping processes involved identify basic requirements of the RNG technique. The basic requirements include the input and output. Then, the initial prototype was developed which includes only user interfaces. The third step was the coding of all operations in RCIA protocol and Enhanced RCIA.

The prototype was developed using Microsoft Access 2013 which is used to design a back-end database, and Delphi XE8 environment developed by Embarcadero Inc. Delphi is a visual Pascal programming language, used to design and develop applications that run under different environment (Windows, Mac iOS, Android). Delphi can be used to develop applications quickly, so-called Rapid Application Development (RAD). This is achieved by using components and tools which are coordinated properly and programmed by writing several procedures associated with specific events. This type of programming called events programming.

Event programming is programming that depends on event occurs for a component in the application. In any occurrence of specific event such as click on a button or close the window, a specific code has already been written in the application executed. Each component can be connected to one or more events to do particular procedure. Delphi programming language allows the programmer to design the required application using several components placed on the Form, and then writes procedures for each event.

In this study, the prototype consists of two main parts, which illustrate the main components in the RFID system. First one represents the RFID tag and reader, while the second part represents the back-end database. The database contains all information that relates to the tag and reader, which is needed to accomplish authentication process. The expected outcome of this phase is a prototype of the RNG technique.

3.5. Phase 3 – Evaluation

After completing second phase, evaluation should be done to ensure that the RNG technique is able to solve the privacy issue. The RNG aims to enhance the security of RCIA protocol to prevent traceability attack. Based on that, the simulation of enhanced RCIA should demonstrate variable values for each message sent from tag to the reader, to prevent the eavesdropper from tracking the location of the tag. At the same time, the simulation will present the limitation in the existing protocol that relates to traceability attack. The expected outcome of this phase will be the validation of the enhanced RCIA which incorporates RNG technique. Figure 3.2 illustrates the structure of the simulation tool.

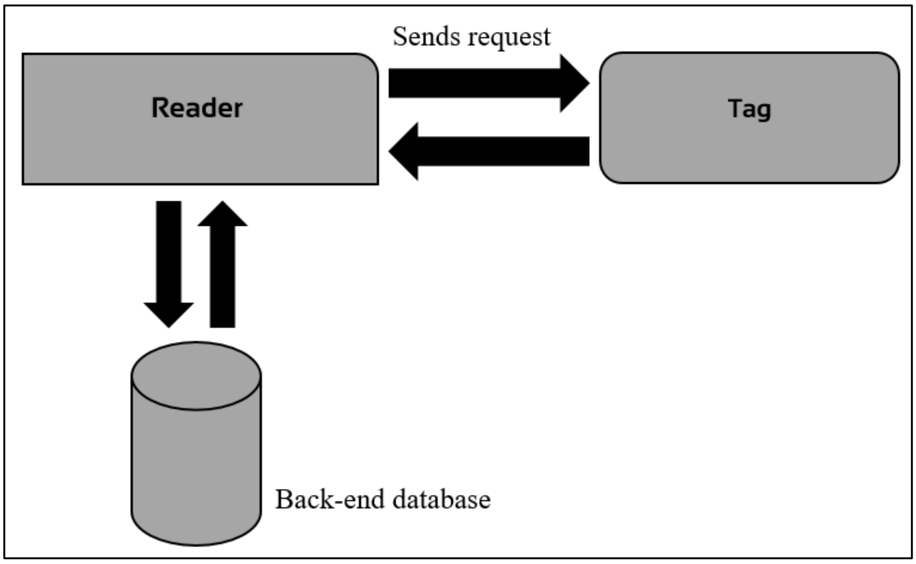


Figure 3.3. The Structure of The Simulation Tool

The above Figure 3.2 illustrates the general structure of the simulation tool. The structure includes, the RFID tag, reader, and back-end database.

3.6. Summary

This chapter has presented the methodology that has been carried out in this study. There were four main phases involved starting from identifying the problems, followed by the conceptual design of the enhanced RCIA. It is then followed by the implementation and finally the evaluation phase. Each phase was designed to achieve intended objectives. For example, phase 1 was designed specifically to achieve objective one. It was also shown that by having such methodology, all phases were able to be completed within the time frame given and all objectives were successfully achieved.

CHAPTER FOUR
RESULT & ANALYSIS

4.1. Introduction

The purpose of this chapter is to present the RNG technique to enhance the security of RCIA ultra-lightweight authentication protocol. Basically this chapter involves three main sections. The first provides explanation of the RNG technique, followed by its implementation and finally the evaluation procedures. Each section, will elaborate further on operations and algorithms involved including the simulation procedures carried out to evaluate the enhanced RCIA.

4.2. The Random Number Generator (RNG) Technique

Enhancing the security of ultra-lightweight authentication protocols in the RFID system is a challenge; due to it support only simple operations like Bitwise [8]. This is because ultra-lightweight protocols were designed for low cost RFID system and this makes it unable to have complex cryptographic methods (e.g.: one way hashed function). This is a distinct characteristic of ultra-lightweight protocols which also serves as a limitation to it. With this limitation, the RNG needs to consider the usage of Bitwise operations to generate random number (Rn) which can effectively be implemented in the tag side [8].

Random Number Generator (RNG) is an algorithm uses to produce a sequence of unpredictable random numbers. The RNG is very important to increase the security of any system due to using the same value for each session will lead to possible traceability attack. Figure 4.1 describes the RNG.

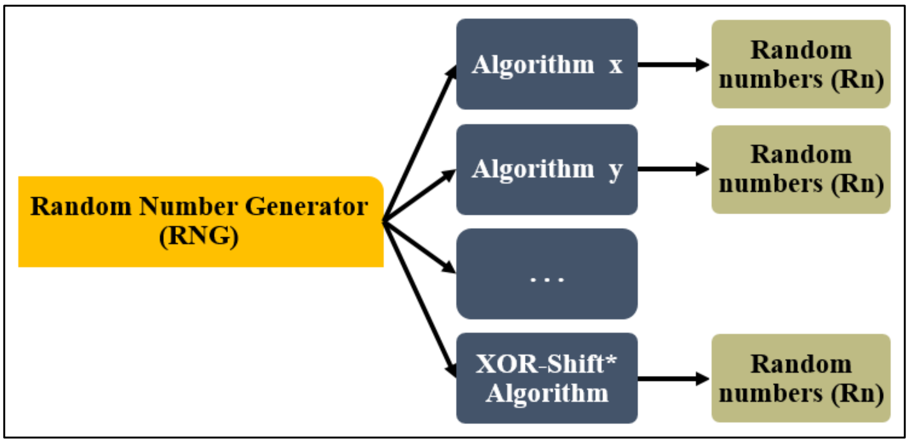


Figure 4.1. Random Number Generator (RNG)

Based on Figure 4.1, RNG can be generated using various algorithms in order to produce random numbers (Rn). For this study, the RNG involves an algorithm based on Bitwise operations. Table 4.1 shows some examples of Bitwise operations.

Table 4.1

Bitwise Operations

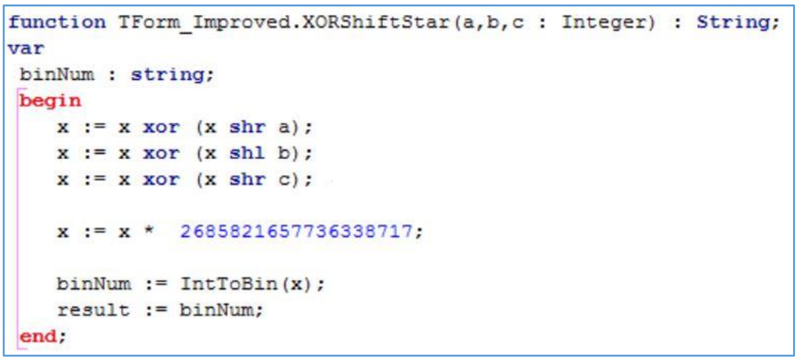
Operator	Symbol	Example
Right-Shift	>>	X >> 2
Left-Shift	<<	X<< 2
AND	**&**	X **&** Y
OR	|	X | Y
XOR	^	X ^ Y
Bit inversion	~	~ X

The RNG technique proposed in this study is based on Bitwise XOR and shifts (left and right). The following section will discuss further on the algorithm used in the RNG called the XOR-Shift* Algorithm.

4.2.1. XOR-Shift* Algorithm

In 2003, XOR-Shift algorithm has been proposed by Marsaglia [26], as a very fast and high quality random number generator. This algorithm is based on repeatedly applying exclusive-OR (XOR) and shift operations (left and right) [40]. However in 2014, Vigna [41], proposed XOR-Shift* algorithm following suggestion in Marsaglia's paper. The suggestion is multiplying the result of an XOR-shift generator by a suitable constant. This constant makes possible to generate a permutation of the sequence by the underlying XOR-Shift generator.

Based on rigorous experimental procedures, this XOR-Shift* generators successfully passed strong statistical test suites tool (i.e.: BigCrush and Dieharder) and was recognized as the fastest generator between all tested generators (i.e.: MT19937 , xorgens4096, WELL1024a and WELL19937a) [41]. Figure 4.2 shows the code of XOR-Shift* algorithm which acts as the main components of RNG. This algorithm takes into account the characteristics of ultra-lightweight authentication protocols and thus can be used in RCIA [41].



```
function TForm_Improved.XORShiftStar(a,b,c : Integer) : String;
var
 binNum : string;
 begin
    x := x xor (x shr a);
    x := x xor (x shl b);
    x := x xor (x shr c);

    x := x *  2685821657736338717;

    binNum := IntToBin(x);
    result := binNum;
 end;
```

Figure 4.2. XOR-Shift* Algorithm

Without RNG in the RCIA protocol, when the illegal reader send request to the tag, it will respond with same IDs in each query session. This makes RCIA protocol vulnerable to traceability attack, which leads to privacy issue. With the RNG, the random numbers (Rn) are generated by using XOR-Shift* algorithm and concatenates with IDs to produce a new one (i.e.: newIDs). This will enable the tag to send different IDs in each query session. Figure 4.3 below shows how RNG helps to prevent traceability attack. With the assumption that the query comes from illegal reader (i.e.: attacker), the tag will respond with different IDs in each query session. For example, in query session (1), the tag returns X as IDs while in query session (2), the tag return Y as IDs. In this case the attacker will not be able recognize whether the IDs belongs to which tag. Thus this prevents traceability attack and solves the privacy issue.

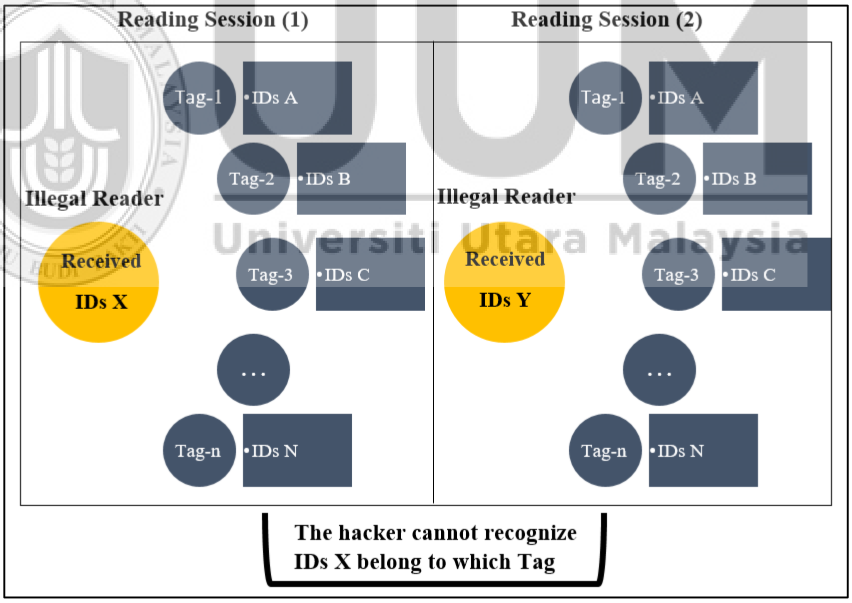


Figure 4.3. The Role of RNG in Enhanced RCIA

4.3. Implementation of the RNG

The implementation of the RNG has been conducted by developing a prototype, due to the lack of hardware components of an RFID system (reader and tag). The prototype consists of three parts, which illustrate the main components of the RFID system, reader, tag and back-end database. The database contains all information that relates to the tag and reader, which is needed to accomplish the authentication processes.

The operation of enhanced RCIA (i.e.: RCIA + RNG) is similar to the operations of existing RCIA protocol except that, in the enhanced RCIA, the RNG was used in the tag side. With this, updating IDs at the end of the query session is no longer necessary due to the randomization operations have been done by the RNG. In order to explain this further, the main operations of the enhanced RCIA will be as depicted in Table 4.2 below:

Table 4.2
The Main Operations of the Enhanced RCIA

Steps	**Operations**
1	Reader: sends a “Hello” message to the tag.
2	Tag: calculates the Rn by using XOR-Shift algorithm and newIDs where newIDs = ID $\oplus$ IDs | Rn. After that the tag sends the newIDs and Rn together to the reader.
3	Reader: receives newIDs and Rn, and compares it with the IDs $\oplus$ ID | Rn in the database.
	a. If a match is found in the database, and k_1, k_2 are old the reader will use $K_{1,old}$ and $K_{2,old}$ to calculate the messages A, B and C. If k_1, k_2 are new the reader will use $K_{1,new}$ and $K_{2,new}$ to compute messages A, B, C.
	b. The reader will generate two random numbers (n_1 and n_2) and concatenate them with A and B. The reader will also compute ($R = n_1 \oplus n_2$), and seed for computation of recursive hash function (R*h*). The seed is computed by taking hamming weight of R mod K (wt(R) mod K).
	c. If IDs is not in the database, then reader will immediately terminate the authentication session with the particular tag.
4	Reader: Uses (R*h*) of the variables K_1, K_2, K_1^*, K_2^*, n1, n_2) to compute “Cr” message.
5	Tag: extracts random number (n_1 and n_2) from messages A and B. Then it computes the seed for recursive hash function using $R= n_1 \oplus n_2$ and wt(R) mod K. The tag further calculates K_1^*, and K_2^* to compute the local value of Ct and compare it with the received Cr. If both values are equal, then the tag will perform two tasks:
	a. Calculate and transmit Dt message towards reader.
	b. Update keys (K_1 and K_2).
6	Reader: receives message Dt, and computes a local value of Dr and compares them together, if a match occurs, then reader will also update keys (K_1 and K_2).

The following Figure 4.4 illustrates all the steps above in the enhanced RCIA.

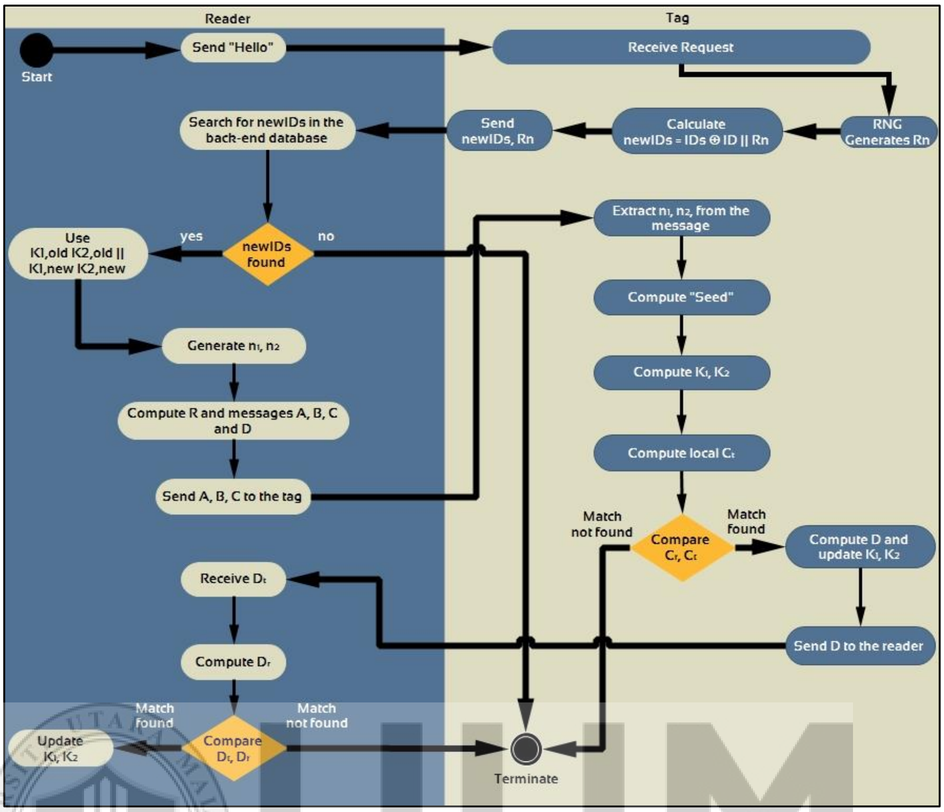


Figure 4.4. The processes of Enhanced RCIA

The implementation involved the processes of RCIA and enhanced RCIA. This is to provide comparison to promote better understanding on the implementation perspectives. Figure 4.5 shows the main interface of the prototype which includes two buttons to access the RCIA and enhanced RCIA.

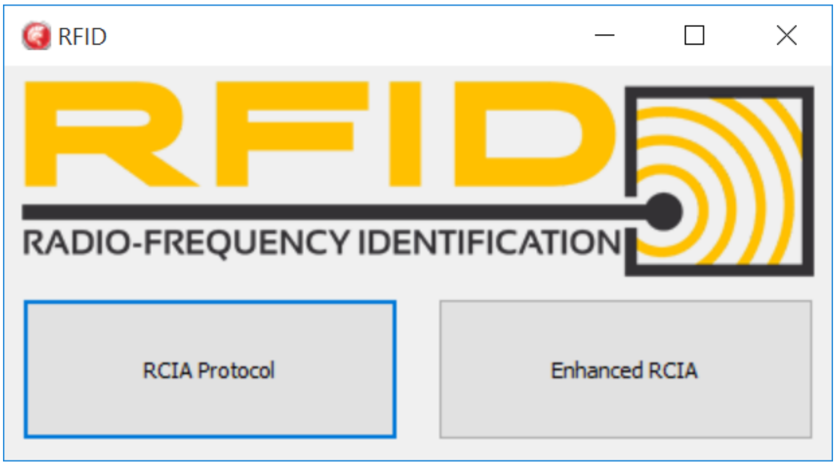
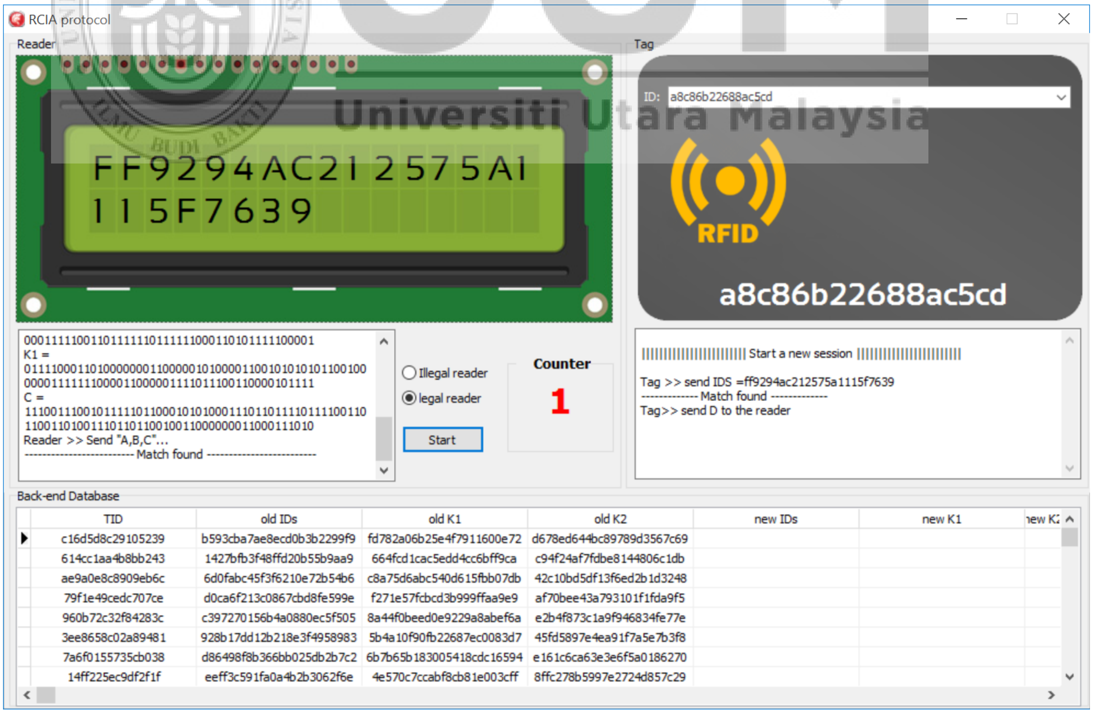


Figure 4.5. The Main Interface of the Prototype

The interface of the prototype consists of display that represents the reader, card that represents the tag and DB-grid which has the back-end database. Figure 4.6 shows the interface of RCIA while figure 4.7 shows enhanced RCIA.



```
FF9294AC212575A1
115F7639
```

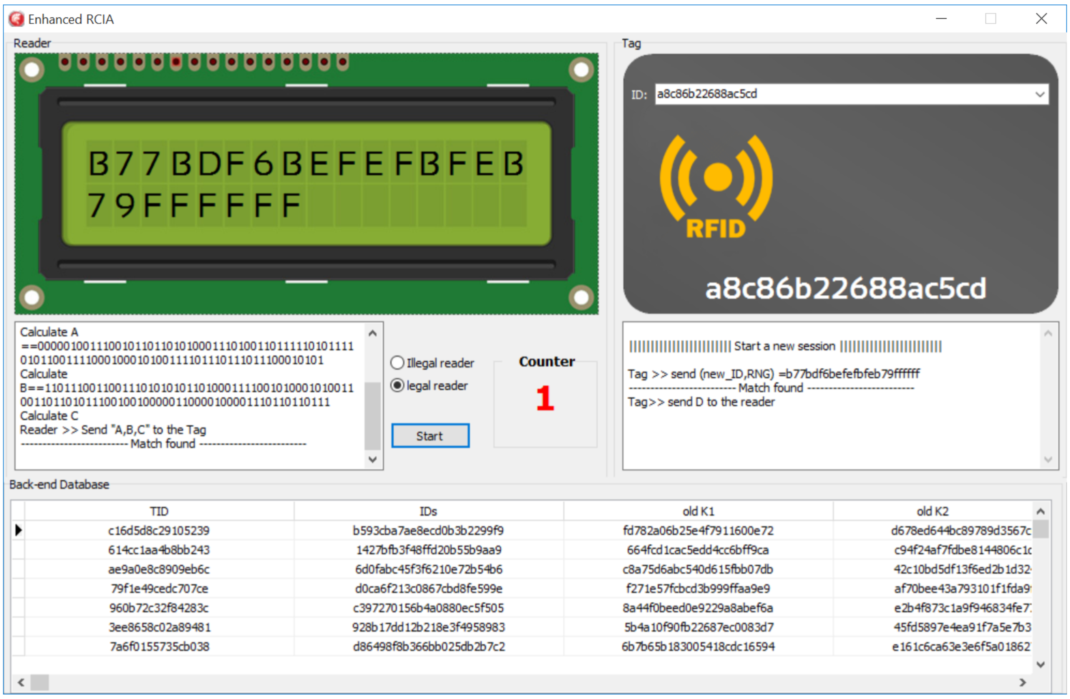
```
0001111100110111111011111100011010111110000 1
K1 =
0111100011010000000110000010100001100101010101100100
0000111111100001100000111101110011000010111 1
C =
1110011100101111101100010101000111011011110111100110
1100110100111011011001001100000000110001110 10
Reader >> Send "A,B,C"...
------------------------ Match found ------------------------
```

```
|||||||||||||||||||||||||| Start a new session ||||||||||||||||||||||||||

Tag >> send IDS =ff9294ac212575a1115f7639
------------- Match found -------------
Tag>> send D to the reader
```

Back-end Database

TID	old IDs	old K1	old K2	new IDs	new K1	new K2
c16d5d8c29105239	b593cba7ae8ecd0b3b2299f9	fd782a06b25e4f7911600e72	d678ed644bc89789d3567c69			
614cc1aa4b8bb243	1427bfb3f48ffd20b55b9aa9	664fcd1cac5edd4cc6bff9ca	c94f24af7fdbe8144806c1db			
ae9a0e8c8909eb6c	6d0fabc45f3f6210e72b54b6	c8a75d6abc540d615fbb07db	42c10bd5df13f6ed2b1d3248			
79f1e49cedc707ce	d0ca6f213c0867cbd8fe599e	f271e57fcbcd3b999ffaa9e9	af70bee43a793101f1fda9f5			
960b72c32f84283c	c397270156b4a0880ec5f505	8a44f0beed0e9229a8abef6a	e2b4f873c1a9f946834fe77e			
3ee8658c02a89481	928b17dd12b218e3f4958983	5b4a10f90fb22687ec0083d7	45fd5897e4ea91f7a5e7b3f8			
7a6f0155735cb038	d86498f8b366bb025db2b7c2	6b7b65b183005418cdc16594	e161c6ca63e3e6f5a0186270			
14ff225ec9df2f1f	eeff3c591fa0a4b2b3062f6e	4e570c7ccabf8cb81e003cff	8ffc278b5997e2724d857c29			

Figure 4.6. The Interface of RCIA



```
Calculate A
==00000100111001011011010100011101001101111101011110101100111100010001010011110111011101110001010 1
Calculate
B==1101110011001110101010110100011110010100010100110011011010111001001000001100001000011101101101111
Calculate C
Reader >> Send "A,B,C" to the Tag
------------------------- Match found -------------------------
```

```
|||||||||||||||||||||||||| Start a new session ||||||||||||||||||||||||||

Tag >> send (new_ID,RNG) =b77bdf6befefbfeb79ffffff
------------------------- Match found -------------------------
Tag>> send D to the reader
```

TID	IDs	old K1	old K2
c16d5d8c29105239	b593cba7ae8ecd0b3b2299f9	fd782a06b25e4f7911600e72	d678ed644bc89789d3567c
614cc1aa4b8bb243	1427bfb3f48ffd20b55b9aa9	664fcd1cac5edd4cc6bff9ca	c94f24af7fdbe8144806c1c
ae9a0e8c8909eb6c	6d0fabc45f3f6210e72b54b6	c8a75d6abc540d615fbb07db	42c10bd5df13f6ed2b1d32
79f1e49cedc707ce	d0ca6f213c0867cbd8fe599e	f271e57fcbcd3b999ffaa9e9	af70bee43a793101f1fda9
960b72c32f84283c	c397270156b4a0880ec5f505	8a44f0beed0e9229a8abef6a	e2b4f873c1a9f946834fe7
3ee8658c02a89481	928b17dd12b218e3f4958983	5b4a10f90fb22687ec0083d7	45fd5897e4ea91f7a5e7b3
7a6f0155735cb038	d86498f8b366bb025db2b7c2	6b7b65b183005418cdc16594	e161c6ca63e3e6f5a01862

*Figure 4.7.*The interface of The Enhanced RCIA

In order to run the prototype for both RCIA and enhanced RCIA, first the reader is chosen. There are options between legal and illegal reader (illegal reader selected by default). Legal reader refers to authenticate users while illegal reader represents attacker(s). Since the study is focusing on security issues, illegal reader is therefore relevant. The tag is selected from the list (i.e.: represented by list of ID in the combo box).

Once the reader and the tag are selected, the authentication processes can be activated with the start button. The counter refers to query session which will increase subsequently after each query. The following section will elaborate further on how the enhanced RCIA is evaluated based on the simulated authentication processes.

4.4. Evaluation

The aim of this evaluation phase is to show the RNG technique that has been embedded to the existing RCIA to ensure that the ID values generated will not be the same for each query session. This will help to solve the privacy issue.

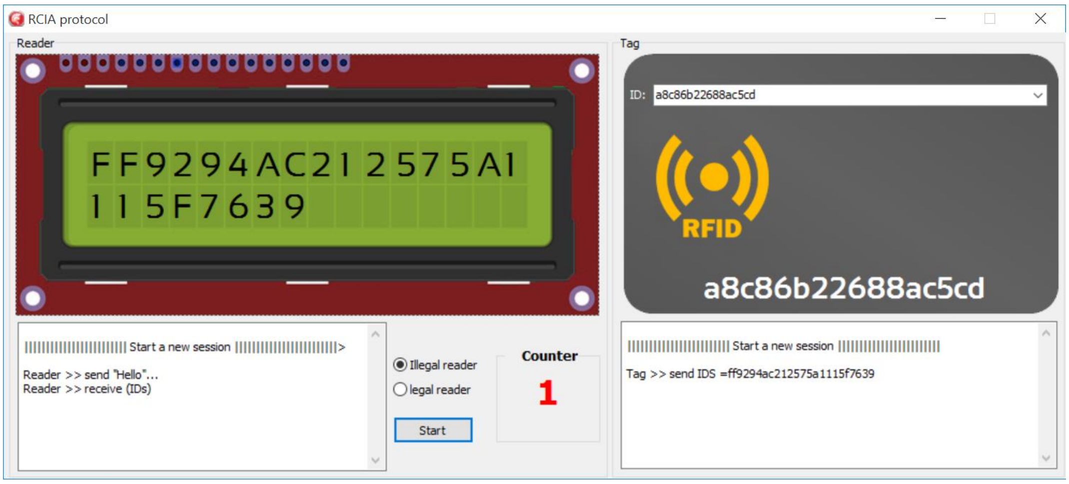
The evaluation scenario involved comparing between the existing RCIA and enhanced RCIA along with the traceability attack model adopted from Jules and Weis [21] .The following table 4.2 describes the processes of the traceability attack model.

Table 4.2
Evaluation scenario

EVALUATION SCENARIO	
Steps	**Attack processes**
1	The attacker takes two tags, e.g. T_0 and T_1 and the identifiers for each one is $(IDs)_0$ and $(IDs)_1$ respectively.
2	The attacker randomly chooses one of the tags (T_0 or T_1), let's say T_i with the identifier $(IDs)_i$
3	The attacker runs one query session with T_i and stores $(IDs)_i$ = X
4	The attacker runs the query session N times by using illegal reader, where N > 1. If $(IDs)_i$ in each time is not equal to X, in this case the attacker cannot track T_i . In other words, the attacker is unable to distinguish between T_0 and T_1. That means the enhanced RCIA successfully prevents the traceability attack.

	Otherwise, in each time, if the T_i responds with same $(IDs)_i$. In this case the attacker can easily track T_i on the basis that $(IDs)_i$ is fixed value.
	Solution – RNG Technique
1	The illegal reader sends query to the tag (T_i)
2	T_i , uses RNG technique to generate a random number Rn and produce newIDs = IDs $\oplus$ ID | Rn
3	The illegal reader received the newIDs
	The attacker runs the query session N times, where N > 1. In each time, Ti responds with different newIDs. In this case, the attacker is unable to distinguish between T_0 and T_1. That means the enhanced RCIA successfully prevents the traceability attack.

The simulation has been performed many times (n > 1) to demonstrate the dynamic values of IDs in each query session. In each session, the tag in the RCIA protocol sent the same IDs to the reader. In contrast, the enhanced RCIA sent different IDs (newIDs) to the reader. Figures 4.8(a) and 4.8(b) show two query sessions of RCIA. In Figure 4.8 (a) and 4.8 (b), it is evident that the tag sends the same IDs (e.g.: FF9294AC212575A1115F7639) in both queries.



FF9294AC212575A1
115F7639

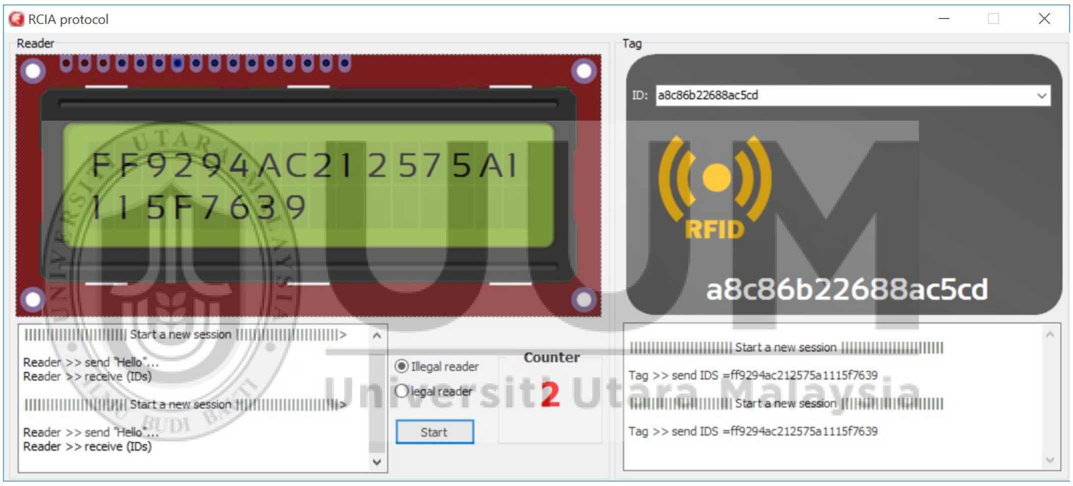
ID: a8c86b22688ac5cd

```
||||||||||||||||||||| Start a new session |||||||||||||||||||||>

Reader >> send "Hello"...
Reader >> receive (IDs)
```

● Illegal reader
○ legal reader
Start

```
||||||||||||||||||||| Start a new session |||||||||||||||||||||

Tag >> send IDS =ff9294ac212575a1115f7639
```

Figure 4.8(a). First Query Session of Simulated RCIA



FF9294AC212575A1
115F7639

ID: a8c86b22688ac5cd

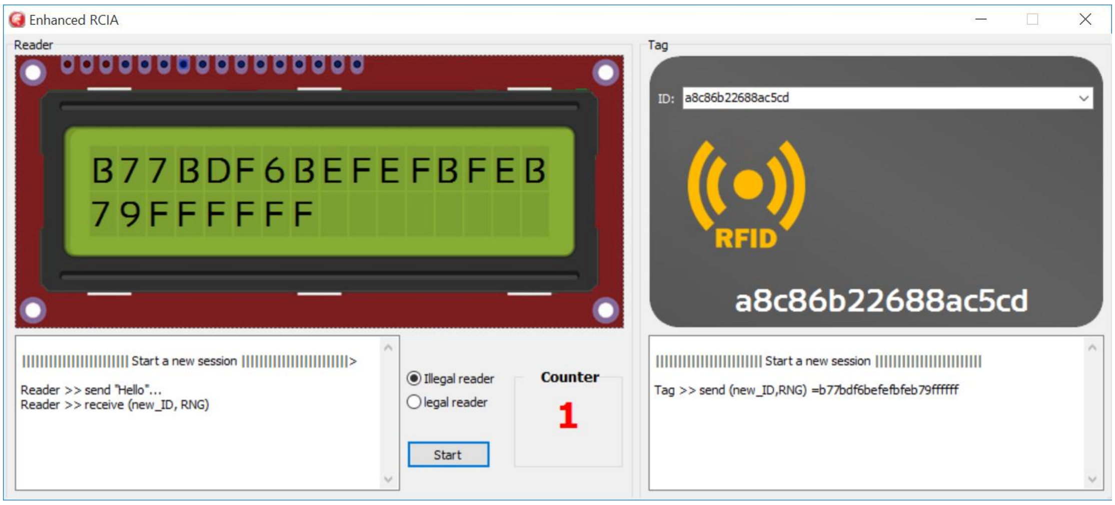
```
||||||||||||||||||||| Start a new session |||||||||||||||||||||>

Reader >> send "Hello"...
Reader >> receive (IDs)

||||||||||||||||||||| Start a new session |||||||||||||||||||||>

Reader >> send "Hello"...
Reader >> receive (IDs)
```

● Illegal reader
○ legal reader
Start

```
||||||||||||||||||||| Start a new session |||||||||||||||||||||

Tag >> send IDS =ff9294ac212575a1115f7639
||||||||||||||||||||| Start a new session |||||||||||||||||||||

Tag >> send IDS =ff9294ac212575a1115f7639
```

Figure 4.8(b). Second Query Session of Simulated RCIA

Meanwhile Figures 4.9(a) and 4.9(b) show two query sessions of enhanced RCIA. In Figure 4.9 (a) and 4.9 (b) shows that the tag sends different IDs (e.g.: Query session 1: B77BDF6BEFEFBFEB79FFFFFF and Query session 2: BF3AFBFBCBEF7F9F7FFFBBFF) in both queries.

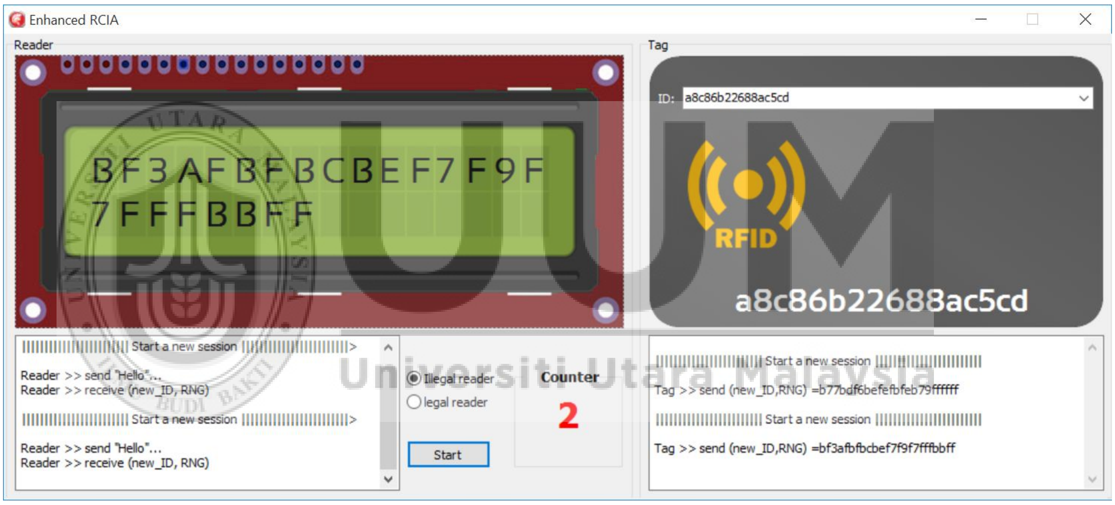


```
B77BDF6BEFEFBFEB
79FFFFFF
```

```
|||||||||||||||||||||| Start a new session ||||||||||||||||||||||>

Reader >> send "Hello"...
Reader >> receive (new_ID, RNG)
```

```
|||||||||||||||||||||| Start a new session ||||||||||||||||||||||

Tag >> send (new_ID,RNG) =b77bdf6befefbfeb79ffffff
```

Figure 4.9(a). First Query Session of Simulated Enhanced RCIA



```
BF3AFBFBCBEF7F9F
7FFFBBFF
```

```
|||||||||||||||||||||| Start a new session ||||||||||||||||||||||>

Reader >> send "Hello"...
Reader >> receive (new_ID, RNG)

|||||||||||||||||||||| Start a new session ||||||||||||||||||||||>

Reader >> send "Hello"...
Reader >> receive (new_ID, RNG)
```

```
|||||||||||||||||||||| Start a new session ||||||||||||||||||||||

Tag >> send (new_ID,RNG) =b77bdf6befefbfeb79ffffff

|||||||||||||||||||||| Start a new session ||||||||||||||||||||||

Tag >> send (new_ID,RNG) =bf3afbfbcbef7f9f7fffbbff
```

Figure 4.9(b). Second Query Session of Simulated Enhanced RCIA

The following Table 4.2 depicts the comparison between query sessions the tag using RCIA and enhanced RCIA.

Table 4.3

The Query Results of the Tag Using RCIA and enhanced RCIA

Query Session	**RCIA**	**Enhanced RCIA**

1	FF9294AC212575A1115F7639	B77BDF6BEFEFBFEB79FFFFFF
2	FF9294AC212575A1115F7639	BF3AFBFBCBEF7F9F7FFFBBFF
3	FF9294AC212575A1115F7639	BFBF55F5AFEF9FDFFFFFF7F5
4	FF9294AC212575A1115F7639	FF9D5F71FFFFDECBF9DDFFF4
5	FF9294AC212575A1115F7639	97DF717DD9ED3F937DDFF3FC
6	FF9294AC212575A1115F7639	9F19FF7989FF9E9FFDD5FFFC
7	FF9294AC212575A1115F7639	DF9CDFFD99ED9F8779D5BFFD
8	FF9294AC212575A1115F7639	9799FD67DDFD5FB379D5BFF7
9	FF9294AC212575A1115F7639	D7BD53E1FBFF9FD3FDF5B3F5
10	FF9294AC212575A1115F7639	DF5EDFF79FEFBFE37BDFFFF6
11	FF9294AC212575A1115F7639	F71D7D7D9BEFDEFBF9D5FFFD
12	FF9294AC212575A1115F7639	F7DF79758BFF9EE779DFBBF5
13	FF9294AC212575A1115F7639	97DEDD798BFD9E83FFD7BFFD
14	FF9294AC212575A1115F7639	97DB5F75EBEF3E9BFDDFBFF5
15	FF9294AC212575A1115F7639	FFBF7771DBED3EE77FFFB7F5
16	FF9294AC212575A1115F7639	DF9E5F718FEF1F937FD7BFF4
17	FF9294AC212575A1115F7639	F738736F9FED7FDFF9F5B3FE
18	FF9294AC212575A1115F7639	B77BD777FDFDFEB37FF7F7F6
19	FF9294AC212575A1115F7639	D73C796FBDFFDFCF7BFDFBFF
20	FF9294AC212575A1115F7639	97DBF9F5AFFD1FD7FDDFBBF5

With this simulated procedures, the enhanced RCIA has able to counter the problem of traceability attack by generating Rn. The different IDs values indicate that the attackers are now unable to trace the origin of the end users and thus prevent privacy violation issue.

1.5 Summary

This chapter has presented the RNG technique which generated the random numbers (Rn) in order to produce dynamic IDs for each query session. This is in general to prevent traceability attack and helps to solve privacy issue. The implementation of the RNG has been done in prototypical-based due to the limitation of hardware resources. The implementation has successfully been done producing clear interfaces showing necessary processes of the enhanced RCIA. The evaluation also has been completed with simulation technique and the results support the idea of the enhanced RCIA.

CHAPTER FIVE
DISCUSSION & CONCLUSION

5.1. Introduction

This chapter discusses and concludes what have been done in this study. Essentially, this chapter includes three main sections. The discussion section will provide brief explanation on how each objective was successfully achieved. Then it is followed by highlighting the contribution of the research work. The last section will present the future work that can be continued further in other research work.

5.2. Discussion

The research started with the identification of traceability attack issue on RCIA protocol. It was noted that the issue was due to the fix value of IDs transmitted during the communication of query sessions. Based from the literature, RCIA authors claimed that the traceability attack would not cause any problems due to the updating of the IDs was done as part of the protocol. However, this scenario would only fit with legal reader since the updating IDs process will be done at the end of each successful query session.

Given another scenario whereby the tag is illegal in the sense that it comes from any possible attackers, the traceability attack is deem possible. This is logically true since illegal readers are not connected to the database and hence the updating of the IDs would never happened considering that the process is done at the end of the session. Since the problem has clearly been identified, based on the reviews conducted on existing ultra-lightweight RFID protocols, it is considered that objective one is achieved.

In order to address the issue of traceability attack in the RCIA, this study adopt a technique known as the Random Number Generator (RNG). This technique was previously used in schemes such as in Hash-Chain scheme [9] and Randomized Hash-Lock [42]. In those schemes, random numbers (Rn) were generated by algorithms which may not be suitable for ultra-lightweight protocols [8].

In this study, the RNG is XOR-Shift* Algorithm. This algorithm was used to produce a random numbers; based on simple operation (i.e.: XOR and shifts) which suits to be implemented in the RCIA. This protocol has the characteristics that do not permits strong cryptographic functions and thus becomes a big challenge to provide good security. As mentioned previously, the XOR-Shift* algorithm used only three XOR and three shifts (left and right) which suits the requirement of ultra-lightweight protocols (i.e.: RCIA) [41]. With the development of the conceptual design for the RNG, it is considered that objective two was achieved.

The implementation of RNG has been conducted by using a simulation. The simulation was developed by Delphi programming language and Microsoft Access. The interface of simulation illustrated main components in the RFID system that is reader, tag and back-end database. The simulation has successfully demonstrated both operations of RCIA and enhanced RCIA protocol. Therefore, it is considered that objective three was achieved.

Based on this simulation, the enhanced RCIA was evaluated. In order to apply the traceability attack, the simulator can be set to act like illegal reader (i.e.: by selecting the “illegal reader” option). In the existing RCIA, if “illegal reader” option is selected, the illegal reader will prevent updating IDs process and force the tag to send the same

IDs in the next query session. This can be clearly seen in Figure 4.8 (a) and 4.8 (b). On the other hand, in the enhanced RCIA, with the same "illegal reader" option, the tag will send variable IDs even if there is no connection to the back-end database. This is because dynamic IDs (i.e.: newIDs) will be generated in the tag itself and not pushed towards the end of the query session (as done is the existing RCIA). This helps to prevent traceability attacks and considered as a solution to privacy issue. By producing this solution, it is considered that the forth objective was achieved.

This study has proven that although RCIA protocol was equipped with preventive mechanism (e.g.: updating IDs), however attacks can happen in various unsuspected ways. Not only that, with advances of technology it is possible that many other attacks can happen and threating the current RCIA protocol.

5.3 Research Contribution

This research contributes to the domain of information security specifically related with RFID security. The contributions can be viewed from several perspectives. First, adopted RNG in the tag side of ultra-lightweight protocol which was not considered in any previous ultra-lightweight protocols. This was probably due to the challenges of applying technique that would suit its characteristics. Many of the cryptographic functions are found not suitable to be implemented in ultra-lightweight protocols [8]. With XOR-Shifts* Algorithm embedded in the RNG, this helps to produce the random numbers (Rn) and provide the dynamicity elements which is important for an IDs.

The second contribution is from the adversarial perspective. In the existing protocol, the traceability attack was discussed with the assumption that the tag will be query only by legal reader and thus traceability attack might not be an issue. However, this study is taking another turn looking at the potential attack generated by illegal reader which can be done by any irresponsible parties. This can also be seen as pro-active preventive mechanism as security researchers should not wait until attacks happened and only then respond. Our enhanced RCIA also contributed to provide better resistance against traceability attack.

5.4. Limitation

In this study the operation of simulation tool limited on demonstrate that the RNG technique has successfully achieved its objective in producing the dynamic IDs. In other words, the simulation tool may not operate exactly like an actual device. Therefore, it only shows how the enhanced RCIA preventing the traceability attack by producing the dynamic IDs, using RNG technique.

5.5. Future Work

In the near future, the ultra-lightweight protocol specifically RCIA can consider other techniques or algorithms that probably may generate better results in enhancing the security. For instance, the RNG can consider another algorithm which may be more efficient. Additionally, other interested researcher can consider hardware implementation to expand the evaluation covering the performance and cost analysis perspective. It would also be useful to work on the adversarial platform, as to come up with several possible attacks so that preventive mechanisms can introduced even before the attacks were identified.

5.6. Conclusion

This study aimed to enhance the security of RCIA ultra-lightweight authentication protocol. This objective has achieved by adopting random number generator (RNG) technique. The RNG produced based on XOR-Shift* algorithm and used to provide a variable values for IDs. The RNG technique helped in preventing a traceability attack and as a result, solves a privacy issue.

The implementation of RNG technique has been conducted by using simulation technique. In order to provide a comparison between RCIA and enhanced RCIA, the simulation included simulating the operations of both protocols. Furthermore, the simulation used to evaluate the enhanced RCIA. The result of simulated enhanced RCIA, showed that the RNG technique has successfully prevented the traceability attack.

5.7. Summary

This chapter has presented the overall discussion of the study. Along the discussion was done, the objectives of the study was also addressed to indicate that all were successfully carried out. Also included in this chapter was the contribution of this study. This chapter end with a highlight on future work as well some a brief conclusion to end the chapter.

REFERENCES

[1] S. A. Weis, "Security and privacy in radio-frequency identification devices," Massachusetts Institute of Technology, 2003.

[2] F. Thornton and P. Sanghera, *How to Cheat at Deploying and Securing RFID*: Syngress, 2011.

[3] S. A. Weis, "RFID (Radio Frequency Identification): Principles and applications," *Retrived from www. eecs. harvard. edu/rfid-article. pd f on,* vol. 1, 2011.

[4] P. R. Agarwal and P. R. Agarwal, "RFID (Radio Frequency Identification) growth in daily life," *International Journal of Scientific Engineering and Technology*, vol. 1, pp. 71-78, 2012.

[5] A. Alqarni, M. Alabdulhafith, and S. Sampalli, "A Proposed RFID Authentication Protocol based on Two Stages of Authentication," *Procedia Computer Science,* vol. 37, pp. 503-510, 2014.

[6] United States Government Accountability Office, "Information Security: Radio Frequency Identification Technology in the Federal Government," Washington GAO-05-551, May 2005.

[7] P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Tapiador, and A. Ribagorda, "Advances in ultralightweight cryptography for low-cost RFID tags: Gossamer protocol," in *Information security applications*, ed: Springer, 2009, pp. 56-68.

[8] H.-Y. Chien, "SASI: A new ultralightweight RFID authentication protocol providing strong authentication and strong integrity," *Dependable and Secure Computing, IEEE Transactions on,* vol. 4, pp. 337-340, 2007.

[9] M. Ohkubo, K. Suzuki, and S. Kinoshita, "Cryptographic approach to “privacy-friendly” tags," in *RFID privacy workshop*, 2003.

[10] M. David and N. R. Prasad, "Providing strong security and high privacy in low-cost RFID networks," in *Security and privacy in mobile information and communication systems*, ed: Springer, 2009, pp. 172-179.

[11] Y. Tian, G. Chen, and J. Li, "A new ultralightweight RFID authentication protocol with permutation," *Communications Letters, IEEE,* vol. 16, pp. 702-705, 2012.

[12] U. Mujahid, M. Najam-ul-Islam, and M. A. Shami, "RCIA: A New Ultralightweight RFID Authentication Protocol Using Recursive Hash," *International Journal of Distributed Sensor Networks,* vol. 2015, 2015.

[13] A. Juels and S. A. Weis, "Authenticating pervasive devices with human protocols," in *Advances in Cryptology–CRYPTO 2005*, 2005, pp. 293-308.

[14] S. Kinoshita, M. Ohkubo, F. Hoshino, G. Morohashi, O. Shionoiri, and A. Kanai, "Privacy enhanced active RFID tag," *Cognitive Science Research Paper-University of Sussex CSRP*, vol. 577, p. 100, 2005.

[15] S. Kumar and C. Paar, "Are standards compliant elliptic curve cryptosystems feasible on RFID," in *Workshop on RFID Security-RFIDSec*, 2006.

[16] K. Rhee, J. Kwak, S. Kim, and D. Won, "Challenge-response based RFID authentication protocol for distributed database environment," in *Security in Pervasive Computing*, ed: Springer, 2005, pp. 70-84.

[17] H.-Y. Chien, "Secure access control schemes for RFID systems with anonymity," in *null*, 2006, p. 96.

[18] A. Juels, D. Molnar, and D. Wagner, "Security and Privacy Issues in E-passports," in *Security and Privacy for Emerging Areas in Communications Networks, 2005. SecureComm 2005. First International Conference on*, 2005, pp. 74-88.

[19] J. Bringer, H. Chabanne, and E. Dottax, "HB^+^+: a Lightweight Authentication Protocol Secure against Some Attacks," in *Security, Privacy and Trust in Pervasive and Ubiquitous Computing, 2006. SecPerU 2006. Second International Workshop on*, 2006, pp. 28-33.

[20] J. Munilla and A. Peinado, "HB-MP: A further step in the HB-family of lightweight authentication protocols," *Computer Networks*, vol. 51, pp. 2262-2267, 2007.

[21] H.-Y. Chien and C.-H. Chen, "Mutual authentication protocol for RFID conforming to EPC Class 1 Generation 2 standards," *Computer Standards & Interfaces*, vol. 29, pp. 254-259, 2007.

[22] M. Hutter and J.-M. Schmidt, *Radio Frequency Identification: Security and Privacy Issues 9th International Workshop, RFIDsec 2013, Graz, Austria, July 9-11, 2013, Revised Selected Papers* vol. 8262: Springer, 2013.

[23] X. Zhuang, Z.-H. Wang, C.-C. Chang, and Y. Zhu, "Security analysis of a new ultra-lightweight RFID protocol and its improvement," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 4, pp. 166-177, 2013.

[24] P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estévez-Tapiador, and A. Ribagorda, "LMAP: A real lightweight mutual authentication protocol for low-cost RFID tags," in *Workshop on RFID security*, 2006, pp. 12-14.

[25] P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estevez-Tapiador, and A. Ribagorda, "EMAP: An efficient mutual-authentication protocol for low-cost

RFID tags," in *On the move to meaningful internet systems 2006: Otm 2006 Workshops*, 2006, pp. 352-361.

[26] T. Li, G. Wang, and R. H. Deng, "Security Analysis on a Family of Ultra-lightweight RFID Authentication Protocols," *JSW*, vol. 3, pp. 1-10, 2008.

[27] T. Cao, E. Bertino, and H. Lei, "Security analysis of the SASI protocol," *Dependable and Secure Computing, IEEE Transactions on*, vol. 6, pp. 73-77, 2009.

[28] G. Avoine, X. Carpent, and B. Martin, "Strong authentication and strong integrity (SASI) is not that strong," in *Radio Frequency Identification: Security and Privacy Issues*, ed: Springer, 2010, pp. 50-64.

[29] H.-M. Sun, W.-C. Ting, and K.-H. Wang, "On the security of Chien's ultralightweight RFID authentication protocol," *IEEE Transactions on Dependable and Secure Computing*, pp. 315-317, 2009.

[30] G. Avoine, X. Carpent, and B. Martin, "Privacy-friendly synchronized ultralightweight authentication protocols in the storm," *Journal of Network and Computer Applications*, vol. 35, pp. 826-843, 2012.

[31] E. Taqieddin and J. Sarangapani, "Vulnerability analysis of two ultra-lightweight RFID authentication protocols: RAPP and gossamer," in *Internet Technology And Secured Transactions, 2012 International Conference for*, 2012, pp. 80-86.

[32] K.-H. Yeh and N. Lo, "Improvement of two lightweight RFID authentication protocols," *Information Assurance and Security Letters*, vol. 1, pp. 6-11, 2010.

[33] Z. Bilal, A. Masood, and F. Kausar, "Security analysis of ultra-lightweight cryptographic protocol for low-cost RFID tags: Gossamer protocol," in *Network-Based Information Systems, 2009. NBIS'09. International Conference on*, 2009, pp. 260-267.

[34] M. Zubair, E. U. Mujahid, and J. Ahmed, "Cryptanalysis of RFID Ultra-lightweight Protocols and Comparison between its Solutions Approaches," *Bahria University Journal of Information & Communication Technologies*, vol. 5, pp. 58-63, 2012.

[35] J. C. Hernandez-Castro, P. Peris-Lopez, R. C.-W. Phan, and J. M. Tapiador, "Cryptanalysis of the David-Prasad RFID ultralightweight authentication protocol," in *Radio Frequency Identification: Security and Privacy Issues*, ed: Springer, 2010, pp. 22-34.

[36] D. F. Barrero, J. C. Hernández-Castro, P. Peris-Lopez, and D. Camacho, "A genetic tango attack against the David–Prasad RFID ultra-lightweight authentication protocol," *Expert Systems*, vol. 31, pp. 9-19, 2014.

[37] G. Avoine and X. Carpent, "Yet another ultralightweight authentication protocol that is broken," in *Radio Frequency Identification. Security and Privacy Issues*, ed: Springer, 2013, pp. 20-30.

[38] W. Shao-hui, H. Zhijie, L. Sujuan, and C. Dan-wei, "Security analysis of RAPP an RFID authentication protocol based on permutation," *College of computer, Nanjing University of Posts and Telecommunications, Nanjing*, vol. 210046, 2012.

[39] Z. Ahmadian, M. Salmasizadeh, and M. R. Aref, "Desynchronization attack on RAPP ultralightweight authentication protocol," *Information processing letters*, vol. 113, pp. 205-209, 2013.

[40] G. Marsaglia, "Xorshift rngs," *Journal of Statistical Software*, vol. 8, pp. 1-6, 2003.

[41] S. Vigna, "An experimental exploration of Marsaglia's xorshift generators, scrambled," *arXiv preprint arXiv:1402.6246*, 2014.

[42] S. A. Weis, S. E. Sarma, R. L. Rivest, and D. W. Engels, "Security and privacy aspects of low-cost radio frequency identification systems," in *Security in pervasive computing*, ed: Springer, 2004, pp. 201-212.