2018

A Secure Quorum Based Multi-Tag RFID System

Ayad Al-Adhami

PHD

A Secure Quorum Based Multi-Tag RFID System

Al-Adhami, Ayad

Award date:
2018

Awarding institution:
University of Plymouth

UNIVERSITY OF
PLYMOUTH

A Secure Quorum Based Multi-Tag RFID System

By

Ayad Al-Adhami

A thesis submitted to the University of Plymouth in partial fulfilment for the degree of

Doctor of Philosophy

School of computing,Electronics and mathematics

August 2018

Acknowledgments

Firstly, thank God for providing me with the power and strength to complete the PhD study.

My deepest gratitude is due to my supervisor Prof Martin Tomlinson, for his guidance and excellence support during my PhD study. He has generously provided me helpful experience and goodness through my studies. I really appreciate his persistent help, tireless effort and massive information through the study.

My gratefully thanks is to the director of studies Dr Marcel Ambrose and my supervisor Dr Ingo Stengel for their support, encouragement throughout this thesis.

Many thanks to the University of Technology, and the Ministry of Higher Education and Scientific Research (MOHESR) in Iraq for providing the financial support for this doctoral study at Plymouth University.

I would like to thank Plymouth University and all my colleagues in Centre of Security and Network (CSCAN) especially my friends Hussam, Abdulwahid, Saad, Hiba and Yaseen for their time and effort in a way of helping me during the study.

My particular thanks go to my friends Omar and Anmar for their general help and support during the study.

Many thanks to my brother Ziyad, my sister Dina and my uncle Mutaz for their love, support and encouragement.

Finally, I dedicated this thesis to the memory of my father, to my mum, to my wife Rasha, and to my daughters Lyan and Lina for their eternal love and support.

Author Declaration

At no time during the registration for the degree of Doctor of Philosophy has the author been registered for any other University award without prior agreement of the Doctoral College Quality Sub-Committee.

Work submitted for this research degree at the University of Plymouth has not formed part of any other degree either at the University of Plymouth or at another establishment.

This study was financed with the aid of a scholarship from the Iraqi Government.

Relevant seminars and conferences were attended at which work was often presented and several papers prepared for publication.

1- SRI Security congress 2015, The 8th Australian Security and Intelegence Conference, Perth, Australia, November, 2015.
2- IEEE (UIC/ATC/ScalCom/CBDCom/IoP/SmartWorld) 2016, The 13th international conference on Advanced and Trusted computing, Toulouse, France, July, 2016.
3- IEEE FiCloud 2016, The 4th International Conference on Future Internet of Things and Cloud, Viena, Austria, August 2016.
4- IEEE ICUFN 2017, The Nineth International Conference on Ubiquitous and Future Networks, Milan, Italy, July 2017.

Word count of main body of thesis: 37,053 words

Signed .....

Date .....

Abstract

Radio Frequency Identification (RFID) technology has been expanded to be used in different fields that need automatic identifying and verifying of tagged objects without human intervention. RFID technology offers a great advantage in comparison with barcodes by providing accurate information, ease of use and reducing of labour cost. These advantages have been utilised by using passive RFID tags. Although RFID technology can enhance the efficiency of different RFID applications systems, researchers have reported issues regarding the use of RFID technology. These issues are making the technology vulnerable to many threats in terms of security and privacy.

Different RFID solutions, based on different cryptography primitives, have been developed. Most of these protocols focus on the use of passive RFID tags. However, due to the computation feasibility in passive RFID tags, these tags might be vulnerable to some of the security and privacy threats. , e.g. unauthorised reader can read the information inside tags, illegitimate tags or cloned tags can be accessed by a reader. Moreover, most consideration of reserchers is focus on single tag authentication and mostly do not consider scenarios that need multi-tag such as supply chain management and healthcare management. Secret sharing schemes have been also proposed to overcome the key management problem in supply chain management. However, secret sharing schemes have some scalability limitations when applied with high numbers of RFID tags.

This work is mainly focused on solving the problem of the security and privacy in multi-tag RFID based system. In this work firstly, we studied different RFID protocols such as symmetric key authentication protocols, authentication

protocols based on elliptic curve cryptography, secret sharing schemes and multi-tag authentication protocols. Secondly, we consider the significant research into the mutual authentication of passive RFID tags. Therefore, a mutual authentication scheme that is based on zero-knowledge proof have been proposed . The main object of this work is to develop an ECC- RFID based system that enables multi-RFID tags to be authenticated with one reader by using different versions of ECC public key encryption schemes. The protocol are relied on using threshold cryptosystems that operate ECC to generate secret keys then distribute and stored secret keys among multi RFID tags. Finally, we provide performance measurement for the implementation of the proposed protocols.

TABLE OF CONTENTS

List of Figures

List of Tables

1. Introduction

1.1 Introduction

Radio frequency identification system (RFID) is an auto-identification technology that enables the identification of objects automatically over a wide range of distance without any contact with tagged objects. The idea of the RFID system was first introduced in 1940 to identify aircrafts of friend or enemy and was called Identify Friend or Foe (IFF) system. The IFF is a radar system that uses a transponders and integrator to identify planes by transmitting a radio signal and detecting the reflected signal (Rieback et al., 2006). After decades, RFID system emerged to become one of the most promising technologies. RFID technology has expanded to be used in various fields of application that need an automatic identification and easy in tracking objects. These fields are needed to store important data and communicate wirelessly with other objects over a wide range without any contact with tagged objects.

The typical parts of RFID technology contain RFID tag, RFID reader, and a back-end server. Depending on the power supply, there are three types of RFID tags; passive, semi-passive and active RFID tag. Passive RFID tags are considered most popular because of the low-cost RFID tags' production. Identification of RFID tagged objects is done without human intervention by sending radio frequency signal from a reader to activate RFID tags. Subsequently, the RFID reader collects and reads RFID tag's data and sends it to a back-end server to verify and analyse RFID data. Practically, some RFID

system often work offline, so verifying and analysing RFID data can be done through the reader.

Nevertheless, the transformation of data through RF has to be secured due to important information that mostly contains secret information in the RFID tag. RFID tag communicates with RFID reader via wireless communication while RFID reader communicates with a back-end server via wire of wireless communication. Because of the wireless communication between RFID tags and RFID reader, the security vulnerability can be increased; many threats such as eavesdropping or interception of the message exchange between RFID tag and RFID reader which increases the demand for security and privacy.

Authentication is considered as the first line of defence against the wireless attacks due to its ability to trust and validate an identity to a reader for verification (Malek, 2012). This process is usually divided into two types: forward authentication and backward authentication. The forward authentication performs when an RFID tag proves its identification number to a verifier (an RFID reader). In contrst, backward authentication performs when a tag works as a verifier to a reader and identifies the access of reader to get the tag's information. The term of mutual authentication protocol refres to the term when both parties need to authenticate each other.

The performance of authentication protocols mainly depends on the time complexity of computations of the identification process. The classification of performance in terms of time complexity classes can be divided into three classes constant time O (1), logarithmic time O (log (N)) and linear time protocols O (N), where N denotes the number of tag in the server (Nance & Naps, 1992; Korsh, 1986)

In order to achieve the security demands, cryptographic primitives have been introduced in authentication protocols to decrease the security and privacy issues of RFID system. Cryptographic mechanisms for traditional authentication can be either one-way hash function or symmetric key encryption or asymmetric key encryption. This type of authentication is called Full-fledged protocols. Simple type protocols require one-way hash function and random number generator and called lightweight protocols. Lightweight protocols use random number generator and simple function such as Cyclic Redundancy Code (CRC), and checksum. In comparison, ultra-lightweight protocols use a simple bitwise operation on tags (Chen, 2007).

Even though cryptographic authentication protocols have been proposed to strengthen the security of RFID systems, most of these protocols still have some limitation in terms of satisfying enough security requirements against tampering with RFID tags or eavesdropping through communication. Despite the fact that the passive RFID systems improves the commercial usage, the security perspective has been affected due to low computation feasibility. Moreover, most of these protocols still focus on a single tag to single reader authentication and do not consider scenarios of using multi-tag authentication. For instance, multi-tag authentication protocols are used in practical applications scenarios that need to distribute RFID tags such as supply chain management. In supply chain management, some number of goods are packed into boxes at a manufacturer, shipped to warehouses, and then sent to retailers and distributors. As an RFID tagged box leaves the manufacturer a quick identification is needed to generate an evidence of verification and scanning at every stage from manufacturers until distributors. Therefore, the primary aim of

the thesis is to improve the security and privacy of RFID system by designing different cryptographic techniques that can be used in single RFID tag authentication and multi RFID tag authentication. This thesis presents firstly the design of a mutual authentication protocol that supports the authentication of single tag to single reader. Secondly, it designs and implements protocols that allow a package of RFID tags to be authenticated within a reader in an RFID system.

1.2 Thesis Structure and Organisation

The sequence of this thesis is structured as follows:

- Chapter two goes through defining backgrounds on cryptographic primitives that are used in this thesis. It reviews basic concepts to provable security and elliptical curve cryptography.
- Chapter three presents comprehensive background of RFID technology, how it does work, essential components of the system and its applications. Moreover, it presents the related security and privacy attacks that can affect the utility of the system.
- Chapter four surveys different cryptographic authentication techniques for RFID system. These protocols are involved with using symmetric key techniques, asymmetric key, and secret sharing techniques. In addition to related research that involves using group of authentication protocols for RFID system in both symmetric key and asymmetric key techniques.
- Chapter five addresses the use of zero-knowledge proof as an authentication protocol for RFID system. It also proposes a mutual authentication protocol that is based on using elliptical curve with zero-

knowledge and hash function. The main contributions of this chapter are as follows.

1. Identifies and formalises possible RFID threats that can influence the system.
2. Proposes mutual authentication protocol that relies on adopting Keccak hash function with zero knowledge proof.

- Chapter six involves designing and presenting two RFID systems that can be used in supply chain management. Both protocols rely on the idea of allowing three or more tags to be authenticated within a reader. These approaches are based on using threshold cryptosystem in association with an elliptical curve. The threshold cryptosystems are used to distribute shared keys amongst needed RFID tags and stored only sharing information. The main contribution of this chapter are depicted as follows:
    1. An efficient encryption method for storing data in each RFID tag.
    2. Tags are not required to perform any computation in the first approach and perform symmetric key technique in the second approach.
    3. Ensuring security and privacy requirement for the system.
- Chapter seven presents the conclusion derived in the thesis, concludes the contribution of the thesis and discusses direction for further research.

1.3 Contribution to Knowledge

The following lists summarise the main contributions of the thesis which are published in conferences.

1- Al-Adhami, A., Ambroze, M., Cristopher, C., Stengel, I. and Tomlinson, M., 2015. A secure sharing design for multi-tag RFID authentication protocol. Proceedings of the 8th Australian Security and Intelegence Conference, Perth, Australia, 30 Nov-2 Dec, pp 87-93, ISBN: 0-7298-0735-5, 2015.

2- Al-Adhami, A., Ambroze, M., Stengel, I. and Tomlinson, M., 2016, July. A Quorum System for Distributing RFID Tags. In *Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCom/IoP/SmartWorld), 2016 Intl IEEE Conferences* (pp. 510-517). IEEE.

3- Al-Adhami, A., Ambroze, M., Stengel, I. and Tomlinson, M., 2016, August. A Quorum RFID System Using Threshold Cryptosystem. In *Future Internet of Things and Cloud (FiCloud), 2016 IEEE 4th International Conference on* (pp. 107-113). IEEE.

4- Al-Adhami, A., Ambroze, M., Stenget, I. and Tomlinson, M., 2017, July. A 256 bit implementation of ECC-RFID based system using Shamir secret sharing scheme and Keccak hash function. In *Ubiquitous and Future Networks (ICUFN), 2017 Ninth International Conference on* (pp. 165-171). IEEE.

2. Security Definitions and Cryptography Primitive Concepts

2.1 Introduction

The main aim of this thesis is to design a provable secure RFID based system. Therefore, this chapter will introduce basic concepts of cryptographic protocols that will be used further in the design and implementations of the RFID based systems. In particular, this chapter will start by defining some basic concepts of cryptography protocols.

2.2 Provable security

The term cryptography refers to the use of a method for securely storing and transmitting data between parties. Historically, the term cryptography refers to using a mathematical algorithm to hide information. Nowadays, the term of cryptography has been associated with mathematical models and computer algorithms. The objective of using cryptography is to enable parties to communicate over secure channels. Any cryptographic protocol is considered as a provable secure protocol unless there is a vulnerability on the protocol that can violate the security of information. This means that in a polynomial-time algorithm a legitimate user can execute a cryptographic protocol with no dangers of breaking the protocol by an adversary. There are many techniques in cryptography which are involved with algorithms such as symmetric key encryption and public key encryption, hash function and Pseudo-random number generator.

2.2.1 Pseudo- Random Bit Generator

A pseudo-random bit generator (PRBG) is an algorithm that maps a bit sequence of input k to a bit string sequence of lengths n where $n \geq k$. The input k is called the length's seed and the output is called bit sequence. The PRBG is commutable and negligible in a polynomial time.

Thus, the security depends on the probability distribution and the period of the output procedure (Menezes, 1996).

2.2.2 Message Authentication Codes

A message authentication code (MAC) is a cryptographic checksum that is used to ensure the authenticity and integrity of data information. MAC allows parties to compute a match function by using a secret key k and a message m and output an authentication function. The MAC algorithms produce fixed lengths of bits by taking an input message, and a secret key then sends the message to the other parties. Upon receiving the MAC message, parties recomputed the MAC message in order to complete the matching process. A MAC algorithm is resistant to existential forgery if and only if there is no advantage in an adversary ability to compute a valid MAC of the data information message (Menzence, 1996).

2.2.3 Cryptographic Hash Function

A cryptographic hash function is a deterministic algorithm that takes a string-length input into a string fixed length output. Hash functions output an arbitrary length called hash value by inputting a string-length called preimage such that

any change into the input produces a different output. Hash function can be used in cryptography applications for its property to reduce the size of value and ensure the data integrity protection. Moreover, the hash function must have the one-way property that can be used as pseudorandom generator to generate several keys with a fixed size value. The one-way property isolates different parts of the system and ensuring that if an attacker knows one value, he cannot know the other values (Ferguson and Schneier, 2003).

A hash function $h: \{0,1\}^* \rightarrow \{0,1\}^l$ is an efficient function that require the following properties:

- Preimage resistant: For all output $y \in \{0,1\}^l$, its is not feasible to find an element $x \in \{0,1\}^*$ such that $y = h(x)$.
- 2nd preimage resistant: For all $x \in \{0,1\}^*$, it is not computationally feasible to find another input $x' \neq x$ such that $h(x) = h(x')$.
- Collision resistant: it is not computationally feasible to find two different input message messages $x \neq x' \in \{0,1\}^*$ such that $h(x) = h(x')$.

2.2.4 Encryption

An encryption scheme consists of three processes, key generation *KeyGen*, encryption *Enc* and decryption *Dec.* The key generation process is a probabilistic expected polynomial time algorithm that is responsible for generating public parameters and used to output encryption key K_e and decryption key K_d. The encryption process is a deterministic polynomial time algorithm which involves a message M and the encryption key K_e and outputs cipher message. The decryption process is a probabilistic polynomial time algorithm that is used to extract the message M from the cipher message C by

using the decryption key K_d. The encryption process can be illustrated as $C = Enc_{k_e}(M) \equiv M = Dec_{k_d}(C)$. Depending on the key types, there are two types of encryption symmetric key encryption and public key encryption. The symmetric key encryption uses a shared key between the sender and receiver while the public key encryption uses a pair of keys one for the encryption called a public key and one private key for decryption.

2.2.4.1. Symmetric key encryption

A symmetric key encryption is an encryption scheme that uses the same key for encryption and decryption such that $K_e = k_d$. The symmetric key encryption algorithm takes a plain text as an input with a key and a message from the plain text then output a cipher message such that $Enc(K, M) = C$, the encryption process executes difference substitution and transformation on the plain text in order to produce a cipher message. A decryption process inputs the shared key and the cipher message in order to output the message M such that $Dec(K, C) = M$.

According to different encryption methods that are used, symmetric key encryption, symmetric key encryption can be classified into two ciphers; stream cipher and block cipher. Stream cipher algorithm is based on combining a bit of plaintext with a stream of pseudo random bits as key encryption in each time in order to encrypt a plain text. Block cipher is an algorithm that take an input of block plain text and secret key for encryption process and output blocks of cipher text as a decryption process. The principal role of symmetric key encryption is to provide confidentiality for the broadcasted data and commonly used by security protocols as session keys for confidential online. Regarding

efficiency symmetric key encryption performs a high level of efficiency. However, there is a limitation in terms of the difficulty in distributing keys securely especially when a large number of parties communicate privately at the same time.

2.2.4.2. Public key encryption

A public key encryption scheme is an encryption scheme that uses different keys in the encryption process and decryption process such that $K_e \neq k_d$.

The term public key encryption or asymmetric encryption was first introduced by Whitfield Diffie and Martin Hellman in 1976 (Menezes, 1999). The idea is to use two different keys in the encryption process, and decryption process with the property that knowing the Public key does not permit some to conclude the private key. The public key and private key are different, and the way of generating keys are mathematically related. In public key encryption, key exchange is simpler than symmetric key encryption and can be broadcasted publically especially with network communication. Moreover, the public key encryption is based on a hard mathematical problem thus provides a high level of security in term of confidentiality and integrity but due to the complex computation of mathematical function, public key encryption needs a memory size for complex computations and high size of keys (Staling, 2011).

Public key encryption involves three probabilistic polynomial time algorithm such as key generation algorithm, encryption algorithm, and decryption algorithm. The key generation process inputs a parameter to generate a public key for encryption and a private key for decryption. The encryption process is used to transforms plain text into cipher message by using a public key and

outputs a cipher message such as $Enc(K_e, M) = C$. Upon receiving the cipher message, the decryption operation starts by converting the cipher message into the original message by using the private key K_d such that $Dec(K_d, C) = M$. All public key encryption algorithms satisfy the property of $C = Enc_{k_e}(M) \equiv M = Dec_{k_d}(C)$.

2.2.4.3. Security concepts of encryption

Any cryptosystem is a goal to an adversary to reveal the cipher message and extract the plain-text information. These challenges depend on the adversary goals and types of attack that involve with the adversary ability (Bellare & Rogway, 1998).

There are two goals for an adversary, One-Wayness (OW) and Indistinguishability (IND).

One-Wayness (OW) is a cryptosystem's designer goal to prevent an adversary from having access to the decryption key without a knowledge of the cipher. On the other hand, Indistinguishability is a cryptosystem's designer goal to prevent an adversary from having a knowledge about the probability of encrypting two messages is larger than one half. For instance, suppose that an adversary has the challenged cipher message C and there are two messages m_0 and m_1 in a message space M. The adversary can learn if the challenged cipher message encrypt a chosen message m_0 or m_1. In other word, the adversary can learn any information about the plain message from the challenged cipher message. In indistinguishability, the adversary aim is to predict given ciphertext from the plaintext corresponding. However, the ability of an adversary is restricted to decrypt only with the above information, but the

adversary ability is to modify a valid ciphertext from the original ciphertext by computing some operations. Therefore, the modified cipher-text for some other plain text, becomes a valid ciphertext. This type of forgery refers to the non-mealleability notion .

Furthermore, in relation to the adversary goal, there are two attacks considered in any cryptosystem that depends on the adversary ability to have the information knowledge. There are two types of attacks related to an adversary which are Chosen Plaintext Attack CPA and Chosen Cipher message Attack CCA. In CPA a chosen message from an adversary can be encrypted and generated an attacker cipher message. In CCA an attacker generates a cipher message query to the decryption oracle in order to reveal plain text information gradually by the decryption process. There are two cases in CCA called non-adaptive chosen cipher text attack (CCA1) and adaptive chosen cipher text attack CCA2. The difference between CCA1 and CCA2 is if the adversary obtains the challenge cipher message by using the decryption oracle this leads to CCA1 else leads to CCA2. A cryptosystem is called semantic secured, or indistinguishability (IND) secured if there is no information that is revealed from the cipher message by an adversary. This includes the knowledge of the cipher message and the length of the cipher message with a probability of distinguishing the chosen cipher message more than one half. Subsequently, there are security models that are related to these attacks. Depending on the adversary power, such models are OW-CPA, IND-CPA, IND-CCA1, and IND-CCA2. The OW-CPA security model refers to the ability of an adversary with given key parameter and cipher message to output his own message m^*. A cryptosystem is called OW-CPA if the probability of winning any value of security parameter is negligible.

The IND-CPA can experiment when the investigator generates a public key and private K_e, K_d and send the public key k_d to an adversary. Upon receiving the public key, the adversary generates the cipher message c' by choosing two plain text messages m_0, m_1 of the same length then selecting randomly $b \in \{0,1\}$ and returning the encryption messagem_b. The adversary tries to guess whether $b' \in \{0,1\}$ and wins if the $b = b'$.

An encryption scheme is secuue against IND-CPA if the probability of an adversary to wil the final challenge is negligable.

A similar notion is called indistinguishability under chosen ciphertext attack (IND-CCA1). The adversary goal starts when requesting queries to the decryption oracle and decrypts a chosen cipher message. After that, the adversary also chooses two plain text messages of the same lengthsm_0, m_1. Sends the chosen message to the encryption oracle to encrypt the message m_b by choosing a random $b \in \{0,1\}$. The adversary has an access to the decryption oracle to decrypt the chosen challenge cipher message C' and guess if $b' \in \{0,1\} = b$. The additional capability of the adversary is that he can encrypt or decrypt messages before the challenge cipher message. Furthermore, in relation to the capability of IND-CCA1, the adversary can have access to the encryption or decryption oracles after receiving the challenge cipher message C' without the ability to send the challenge cipher message to the decryption oracle.

A cryptosystem is called semantic secured, or indistinguishability (IND) secured if there is no information is revealed from the cipher message by an adversary. This includes the knowledge of the cipher message and the length of the cipher message with a probability of distinguishing the chosen cipher message more

than one half. An encryption scheme is called semantically secure against chosen cipher text attack (IND-CCA2) if the adversary has no negligible advantage to perform the following processes. The adversary goal starts when requesting queries to the decryption oracle and decrypts a chosen cipher message. After that, the adversary also chooses two plain text messages of the same lengthsm_0, m_1. Sends the chosen message to the encryption oracle to encrypt the message m_b by choosing a random $b \in \{0,1\}$. The adversary has an access to the decryption oracle to decrypt the chosen challenge cipher message C'. After receiving C', the adversary can ask again for the polynomial bound numbers of quires on the decryption oracle adaptively in the same way of the previous process except that the adversary cannot ask for decrypt query that involve the challenge C'. After that, the adversary outputs b' and win if $b = b'$. The notion refers to an encryption scheme if it is IND-CCA2 is the highest level of security compared to IND-CCA1 and IND-CPA. The IND-CPA notion is the lowest level of security.

2.2.5 Zero-Knowledge proof

Goldwasser and Micali introduced zero-knowledge proof in 1989. The idea of the zero-knowledge proof is based on proving a statement without revealing any information including the information of proving a statement (Goldwasser & Micali, 1989). The Zero-knowledge proof uses an intractable computation process between two entities: a claimant A and a verifier B (Goldreich, 2003; Blahut, 2013). A claimant controls a part of secret knowledge while revealing no information to the verifier. Moreover, zero-knowledge proof allows proof of confirmation while conveying no information about the initial confirmation. A claimant A sends a random challenge to the verifier B, then the verifier B replies

with a random challenge and sends it to A. A secretly calculates the result which is then sent to B. B verifies the challenge response, if the verification was successful then it will be accepted otherwise the process will stop. The verification process can be repeated N times depending on the requirement for the verification. The procedure of zero-knowledge proof can be shown in figure 2.1.

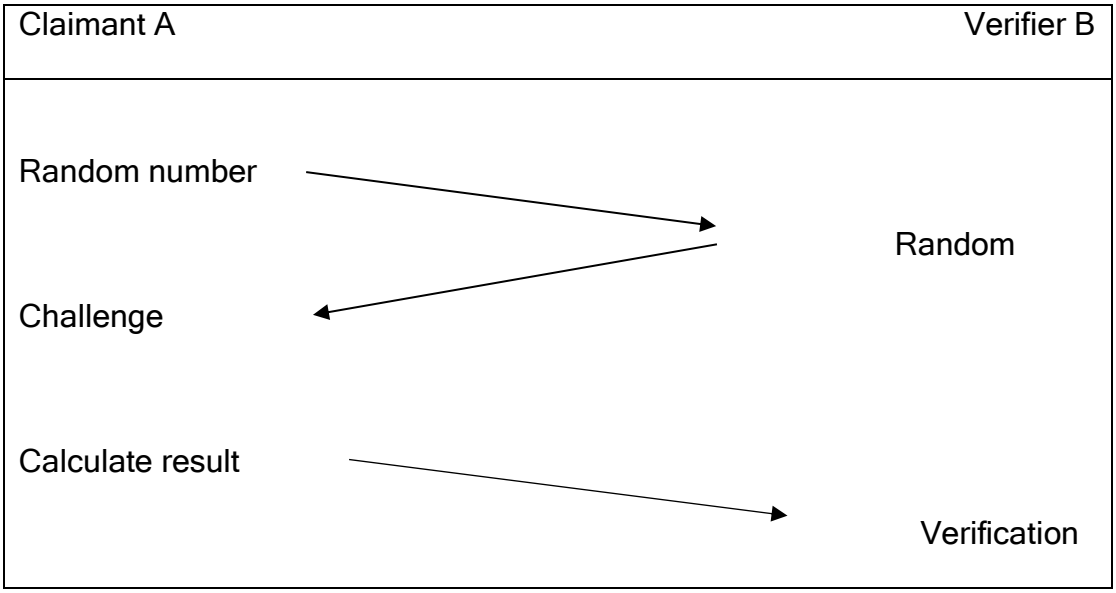

Figure 2 1: Zero-knowledge proof procedure

The zero-knowledge proof has three properties: completeness, soundness, and zero-knowledge. The completeness means if the process of the claimant and the verifier are complete and the statement is true then the completeness was proven. Soundness means that if the statement of verification is false, then the verifier cannot be cheated. The zero-knowledge means if the statement is true then the verifier cannot obtain other information that can affect the privacy of the process.

The idea of the zero-knowledge proof is that the secret key is must be held by the claimant as a proof of ownership. Similar to public key cryptosystem, zero-knowledge proof protocols are based on several mathematical problems such

as the square root problem, the large integer factoring and discrete logarithm problem. The feature of using the zero knowledge proof technique it does not suffer from retreating of security by repeated use (Blahut, 2013).

2.2.6 Secret sharing

Secret sharing and key splitting is a technique that is used to share secret information among parties in a manner way that keep the confidentiality of data, such that there is no single share to reveal secret information. The idea behind this technique is that the secret information is to be hidden and divided into shared keys such that these keys are a subset of the original keys then distribute shares among some number of parties. For example, consider a function that split keys to *n* parties by inputting a value s such that the shares will be $s_1, s_2, \ldots s_n$. The reconstruction of the secret key *s is* combined by defining a set of shares as the shared keys made as a part of the secret key.

The main of aim of using secret sharing techniques is hide and split information among parties such that it's difficult to gain the secret information without authorised participation. The secret sharing techniques are used for protecting the confidentiality of secret information that cannot be randomly notable. Therefore, shared keys require to be computationally indistinguishable for an adversary to collect different information and reconstruct the secret key. Splitting secret keys to parties is a simple way of distributing keys. However, an adversary can easily construct the secret key from shared keys. Thus, a secret sharing scheme needs to split and distribute keys to *n* parties in a certain method such that secret key can only be reconstructed when a number of needed shares *k* is less than the number of *n* shares. Secret shares techniques

is a useful method to distribute shares among at least three parties because computations two parties can easily be revealed (Blakley, 1979).

2.2.6.1. Additive secret sharing

An additive secret sharing scheme is a way where the secret share is distributed as a sum of shares among parties. The requirement for reconstructing the secret shares is the knowledge of all shares, such that the method is used for creating shared keys and reconstructs the secret key from the shared keys. The additive secret sharing algorithm can be used over a finite field $\mathbb{Z}_p$ where p is a prime number and starts when the secret key s is distributed among n parties such that $s_n = s - \sum_{i=1}^{n-1} s_i \, mod \, p$. For reconstruction, all shared keys are known and the secret key is computed by completing the sum of shared keys such that $s = \sum_{i=1}^{n} s_i \, mod \, p$.

The additive secret sharing is a powerful scheme unless there is no change in the shared key information (Trappe & Lawrence, 2006). However, if there is a messing in one shared key, the reconstruction will be failed. Therefore threshold secret sharing scheme is being used to solve this problem.

2.2.6.2. Threshold secret sharing

Threshold secret sharing is a technique that is used to overcome the problem of missing one or more shared key in the additive secret sharing scheme. The threshold idea or *(k,n)* secret sharing scheme is to use a polynomial of degree *k-1* that creates numbers of needed shared keys *n* from secret key *s* and recombs shared keys *k* such that $n \neq k$. In the reconstruction procedure, the parties do not know any information about other shared keys; therefore a

process of interpolation computation is done to reveal the secret key (Shamir, 1979).

2.2.7 Elliptical curve cryptography

For cryptography applications, elliptic curve cryptography (ECC) was first introduced by Neal Koblitz and Victor Miller in 1985 a public key cryptosystem. ECC required a small key size in comparison with other public key cryptosystem which leads to having higher efficiency and speed in generating keys. For example, using small parameters in ECC leads to having the same security with a big parameter in other public key cryptosystems. For that reason, ECC can be used with small devices such as RFID tags.

Elliptic curves can be defined on different algebraic fields. For cryptography, elliptic curves use coordinates that are from a prime field $\mathbb{F}_p$ or a power of 2, $\mathbb{F}_{2^m}$ (Hankerson et al., 2006).

The elliptic curve is given by:

$$y^2 = x^3 + ax + b \; modulo \; p \tag{2.1}$$

Where $(x, y) \in \mathbb{F}_{2^m}$. For some $a, b \in \mathbb{F}_{2^m}$ there is a constraintt that the determinant Δ should be non-zero where $\Delta = -16(4a^3 + 27b^2) \; modulo \; p \neq 0$. As the coordinate y is given by $(x^3 + ax + b)^{0.5} \; modulo \; p$ then $x^3 + ax + b \; modulo \; p$ must be a quadratic residue of $\mathbb{F}_p$. If (x_1, y_1) is a point on the curve then $(x_1, -y_1) = (x_1, p - y_1)$ is also on the curve.

The point of the curve forms a group $\mathbb{E}(\mathbb{F})_p$ and the total number of points on the curve is $\left|\mathbb{E}(\mathbb{F})_p\right| \leq p + 1 + 2p^{0.5}$ (Washignton, 2008). The total number of points $P(x, y)$ satisfies equation 2.1 with infinite order O that should be also

prime. By defining the order O, The set of points under the operation is an Abelian group and has the following properties (Shanmugam et al., 2001):

(1) A point P is called base point such that $nP = O$.

(2) Base point $p(x, y)$ has an inverse denoted as $-P(x, -y)$.

(3) Adding two points together $P + Q$ result in a third point (x_3, y_3) as follows:

$$x_3 = \lambda^2 - x_1 - x_2 \tag{2.2}$$

$$y_3 = \lambda(x_1 - x_2) - y_1 \tag{2.3}$$

Where

$$\begin{Bmatrix} \lambda = \frac{y_2 - y_1}{x_2 - x_1} \; if \; P \neq Q \\ \lambda = \frac{3x_1^2 + a}{2y_1} \; if \; P = Q \end{Bmatrix} \tag{2.4}$$

(4) For all points in the Abelian group, there is a scalar k or a point multiplication of P by k such that $if \; k \geq 1, \; then \; kP = P + P \ldots P$.

The scalar multiplication is an important operation in elliptic curve that is given an integer n then find the value point value of nP on the elliptic curve. On the other side, the discrete logarithm problem (DLP) on the elliptic curve is to find an integer n by giving points P and Q such that $= Q$. Elliptic curves, in general, possess the discrete logarithm property that is so useful in cryptography Blake et al., 1999). Points on the curve may be defined by a scalar multiple of a base point. This leads to the elliptic curve version of Diffie-Hellman.

2.2.8 Diffie-Hellman Key exchange

Diffie and Hellman (1976), introduced a key exchanged protocol that is based on using discrete logarithm problem. The Diffie-Hellman introduces the first idea of public key encryption. Their protocol allows two parties to set up a shared secret by exchanging messages through a public key authentication channel. The protocol starts when both parties choose a prime $p \in \mathbb{F}_p$ and a generator $g \in \mathbb{F}_p$. The first party selects a random $a \in \mathbb{F}_p$ then computes and sends $g^a \ mod \ p$. Upon receiving the g^a, the second party chooses $b \in \mathbb{F}_p$, then computes and sends $g^b \ mod \ p$. The secret key is now $g^{ab} \ mod \ p$. Both parties need to compute $g^{ab} \ mod \ p$ and $g^{ba} \ mod \ p$ and send these values to each other. If these values are correct then the protocol is completed otherwise the process will be rejected (Diffie & Hellman, 1976).

There are two types of Diffie-Hellman problem, Computational Diffie-Hellman problem (CDH) and the decisional Diffie-Hellman problem (DDH). In the CDH problem, a randomly points P, P^a, P^b in the Abelian group is given then computes P^{ab}. In the DDH problem, random points P, P^a, P^b and P^z are given then decide if z= $ab \ modulo$ the group order of point P.

2.2.9 Bilinear Pairing

A pairing is a function that uses two points on the elliptical curve as input and output a valid element in the cyclic group (Boneh & Franklin, 2001). Let G_1, G_2 and G_T be cyclic groups of prime order p and g be a generator of G. A map e is called bilinear map if the following properties are satisfied:

- **Bilinearity** For $g_1, g_2 \in G_1, G_2$, and $a, b \in \mathbb{F}_p$ there is a map e such that $e(g_1^a, g_2^b) = e(g_1, g_2)^{ab}$
- **Non-Degenracy** If $g_1 \in G_1$ and $g_2 \in G_2$ then $e(g_1, g_2) \neq 1$ is consider as a generator for G_T
- **Computationality** There is an efficient algorithm to compute $e(g_1, g_2)$ for all $g_1 \in G_1$ and $g_2 \in G_2$

The bilinear pairing is called symmetric if $G_1 = G_2$ otherwise its called asymmetric.

The bilinear pairings were introduced to construct fast algorithm to solve the Discrete logarithm problem and the DDH problem in elliptical curves. Thus, new assumptions are based on the bilinear pairing to be used in cryptography. Such assumptions are Bilinear computational Diffie-Hellman problems and Bilinear Decisional Diffie-Hellman problem (Boneh & Franklin, 2001).

Given $(g, g^a, g^b, g^c) \in G$ for unknown $a, b, c \in \mathbb{F}_p$, the Bilinear Diffie-Hellman (BDH) problem is to compute that for any probabilistic polinoyal time algorithm, the advantage to guess the probability of $(g, g^a, g^b, g^c) = e(g_1, g_2)^{abc}$ is nigigable.

The Decisional Bilinear Diffie-Hellman problem (DBDH) is Given $(g, g^a, g^b, g^c) \in G$ for unknown $a, b, c \in \mathbb{F}_p$, the DBDH problem is to decide whether $z = (g, g)^{abc}$ so the advantage of any probabilistic polynomial time algorithm in solving DBDH is negligible. That means the probability of $(g, g^a, g^b, g^c, e(g, g)^{abc} = 1$ minus the probability of $(g, g^a, g^b, g^c, Z = 1)$ and is computationally infeasible for any polynomial time algorithm to distinguish between them.

2.3 Summary

In this chapter, we surveyed some of the concepts of provable security, together with the cryptographic primitives that will be used in the rest of this thesis. We also provided an overview of secret sharing techniques alongside with elliptic curve cryptography and key exchanged protocol s which are used to design security models.

3. Overview of RFID technology

3.1 Introduction

Automatic Identification and Data Capture technology (AIDC) is one of the effective methods which are used to identify objects automatically without human intervention. AIDC technology is often used in many applications due to its services which include recognition of objects and obtaining information from objects, then enter information into a database system without human intervention (Finkenzeller, 2010). Moreover, AIDC technology is used by decreasing errors during the data entry process, ensures time and ensures lower labour costs. Over the past decades, AIDC technologies have been increasingly used in supply chain management to improve the efficiency of the supply chain by providing information and identifying products, people, goods and animals. Data capture technologies consist of many technologies, such as the barcode, Radio Frequency Identification (RFID), biometric, magnetic stripes and smart cards. The cheapest and appropriate technologies that are used in supply chain management are barcodes and RFIDs. In this chapter, a brief overview of RFID system will be introduced. The application of using RFID and RFID security and privacy problem will also be introduced.

3.2 Barcode technology

Barcode is one of the common AIDC technologies which have been used over the past 30 years. It consists of sequential black and white labels that can be read by an optical scanner. Barcode has been developed in the early 1970's to accomplish the requirement of the organisations in supply chain management

that needs labelling each product around the world. There are many types of barcode standards, but the barcode mainly used is the Universal Product Code (UPC), which is a 12-digit barcode that is used extensively for retail packaging in the United States. EAN (European Article Number) is a barcode standard that consists of either 12 or 13-digit product identification code. ISBN is the International Standard Book Number (ISBN), which is a unique commercial book identifier barcode that contains either 10 or 13 digits. UPC and EAN are the leading providers that make available barcode over goods. Typically, a unique serial number is stored in a database; The database provides the information of objects by their serial numbers. Barcodes are divided into three types: linear barcode and 2-D barcode and 3-D barcode. Linear barcode uses linear serial numbers which are scanned by an optical barcode scanner.

The linear barcode is very common and used everywhere due to its low costs. Most items in shops are labelled by using linear barcode. However, data cannot be stored in the linear barcode due to the small barcode size of the linear barcode. Usually, barcode contains numbers and characters, these numbers are unique, so the data of the barcode is read from the barcode, sent to a computer, then the computer returns the information about the item. The two-dimensional barcodes are barcodes that need to be read in two dimensions. The advantage of 2-D barcodes over the linear barcodes is the storage capacity of a small amount of data. However, these types of barcodes need special scanners that can scan the barcode, which in effect involve more expense than linear barcode. Typically, there are over 20 types of 2-D barcodes such as QR Code, PDF417, Data Matrix, Semacode, and MaxiCode, etc. The three dimensional barcode is the same linear barcode but embossed on the surface.

Barcodes are reliable to use as means of supply chain management in order to improve information accuracy, sharing information, tracking and controlling products (Manthon & Vlachopoul, 2001). Moreover, barcodes can be attached in particular to all products (Schutzberg, 2004). However, data cannot be reused or modified after reading or writing the barcode. Also, data storage is limited and reading distance is limited.

3.3 RFID technology

RFID technology is a technology that identify items, animals, and people by using radio frequency which communicate the ID of small devices that attached to the items. The attached devices are used to store information and details about each item. RFID technology, being similar to the barcode, can also be read at a distance.This particular property gives RFID system the priority to be used in many applications rather than the barcode, which will be explained in details in this chapter.

The essential parts of RFID technology includes three components, RFID tag, RFID reader and the back-end server. The function of RFID technology uses RFID reader to read the attached RFID tags by using radio frequency, and then send information via wire or wireless means to a back-end server for verifying or updating information of items (Wies, 2007).

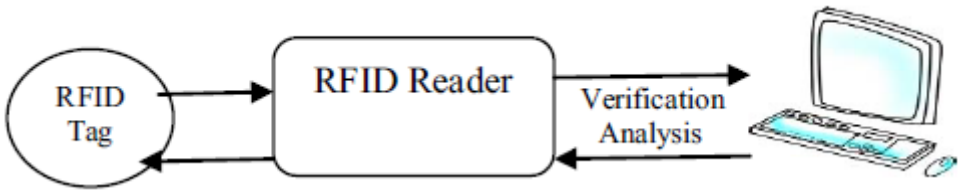

Figure 3. 1:RFID system

3.3.1 RFID tag

RFID tag or transponder is similar to barcode in identifying devices which are attached to an object by using RFID reader. Basically, each tag consists of an antenna, a microchip and encapsulating material. Some of an RFID tag antenna is structured from a small coil of wire or dipoles and used to connect with RFID reader through radio frequency signals. In some cases, RFID tag has some sensors for the measurement of temperature or humidity value. RFID microchip is used to store object information that RFID can read through radio frequency signal. RFID tags have different memory sizes and each tag contains a unique identifier number.

Depending on the power supply, there are different types of tags, which use different types of power, such as passive tags, semi-passive tags, and active tag. RFID reader powers a passive RFID. Usually, passive RFID tags are inactive, however, as soon as they receive the emission from a reader they wake up and start the process and transmit data. The majority of RFID tags produced are passive RFID tags, due to their low-cost. The distance for reading passive RFID tags is limited due to power limitation (Burmester.,et al 2008). Active RFID tags contain a battery that powers radio communication. The power of active tags allows the transmission of data to the RFID reader or to other RFID tags. In addition, active RFID tags have a more extended reading range. Also, the size of RFID tag is bigger than passive tags due to the capacity of storing processing data. Semi-Passive RFID tags are based on integrated power sources to run the tag chip circuit and extract the communication energy from the reader for regular radio communication. Additionally, semi-passive tags have a more extended reading range than passive tag but not longer than

active tags. Table 3.1 describes the difference types of RFID tags (Finkenzeller, 2010).

RFID tag	Advantage	Disadvantage
Passive	Longer life time Lowest cost more flexible in devices	Limited distance 3 Meter or less
Semi-Passive	More extended range for communication 100 Meter Can be used as a communicator	Expensive Cannot determine battery status
Active		

Table 3. 1:Comparison between RFID tags

3.3.2 RFID reader

The RFID reader is controlled by a digital signal processor, which can communicate with RFID tags. Reader consists of two parts: an antenna and electronic module. The antenna is used to wirelessly connect with RFID tags through radio frequency signal, then capture data, while the electronic module is networked to the backend server through a wired or wireless network. The functionality of the RFID reader is to read / write information into or from the RFID tag. In the passive systems, RFID readers transmit energy through electromagnetic fields to tags and provide the energy necessary for the tag to respond to reader operation queries. In the active systems, a reader is used to read and write information into the tag and the battery is used to enhance the query range between the reader and the active tag. There are different types of

RFID readers such as wire reader or wireless reader or integrated into a mobile computer.

3.3.3 Back-end server

The back-end server is responsible for the reception of data from the reader. It filters and processes data. Moreover, the back-end is for the reception of data from the reader, records objects, tracks movement, verifies identifiers and gives authorization for tags and for the control system database.

3.3.4 RFID communication

The communication between RFID reader and RFID tag can be categorised into two communication channels, the forward communication channel and the backward communication channel. The forward channel provides energy to the RFID tags and transfer of data from the reader to the tags, while the backward channel is used for sending data from the RFID tag to the reader (Bolic et al., 2010). The forward channel is more extended than the backward channel. The power of forwarding communication channel can be increased to expand the network domain as it determined by the transmission signal strength of the reader. The backward communication channel works with proximity requirement between RFID reader and RFID tags. Taking into consideration the communication, it can be run using several radio frequency bands, such as low frequency (LF), high frequency (HF), ultra-high frequency (UHF) and microwave frequency (MF) (Rieback et al., 2006), (Wies, 2007)

The Low-Frequency signal operates between 125 kHz and 142 kHz. HF tags can be used over a short range for less than 1 meter with the lowest data

transfer rate among other frequency rates. Usually, this type of tag is passive, and its ability to read on metal is the best among other frequency ranges. Most of the passive tags cannot be readable when they are attached to metal.

The High-Frequency signal operates at 13.56 MHz; the operation is similar to LF but offers better range than LF extending up to 1 meter with larger memory size. Usually, this type of tags is passive, and its ability to read on metal is not as good as at the LF.

The Ultra High-Frequency signal operates at 433 MHz and 860- to 960- MHZ, using UHF in reading tags within a distance up to 20 Meter. Also, data processing is faster than LF and HF signals with a smaller size of the tag. Usually, UHF tags are used with products that do not contain metal or water. Use of UHF signal over 860-960 MHz is used for passive tags and 430 MHz is used for active tags.

The Microwave Frequency (MF) operates at 2.45 -GHz and can be used for active and passive tags. The MF is used for higher data rate and uses a real-time location system; also the size of tag is smaller than those at other frequencies, and the distance is up to 10 meters.

3.3.5 RFID standards

There are international standards for RFID technology and depending on the allowed frequencies in different countries. Significant organizations develop standards for RFID technology such as International standards organization (ISO) and EPC Global. Moreover, there are specific industry groups who work on RFID standards, such as the European Telecommunication Standards Institutes (ETSI), Federal Communication Commission (FCC), American

Trucking Association in the transport industry, Near Field Communication (NFC) in electronics and mobile devices industries, Asian Group in the automotive industry. An RFID ISO standard covers different areas of technology, such as proximity cards, air interface, animal identification and supply chain management. The EPC standard mainly covers the area of supply chain management (Korkmaz and Ustandag, 2007; Huang, 2009).

3.3.5.1. EPC Standards

In 1999, Uniform Code Council (UCC) and EAN international joined a project to make the final EPC (Electronic Product Code) standard as an official global standard and to research RFID technologies aimed at establishing the standards for the RFID technologies. The former AUTO-ID society carries this mission at Massachusetts Institute of Technology centre including numbers of partners, such as Wal-Mart, Procter and Gamble, and Gillette. EPC standards work at UHF frequency and mainly work with data management, supply chain management, and inventory management. EPC enables identification of objects over the world by providing a unique identifier number for each object. The EPC code is a unique 96 bit number that allows each manufactured object to be identified, instead of a type of product (Hutto and Atkinson, 2004). EPC can be classified into five types; these types are different in each class used in an application. The classification of EPC standard depends on the ability of reading or writing and the type of tag (Violino, 2005), as shown in table 3.2.

EPC Class Type	Properties	Type of tag
Class 0	Read Only	Passive (64 bit only)
Class 1	Write Once, Read Many	Passive (96 bit min)
Class 2 (Gen 2)	Read/Write	Passive (96 bit min)
Class 3	Read/Write with battery	Semi-Active
Class 4	Read/Write active transmitter with battery	Active

Table 3. 2: Type of EPC classes (Violino, 2005)

3.3.5.2. ISO standard

The International Organization for Standardization (ISO) is a non-government organisation which focuses on creating standards for products. ISO published over 13000 international standards for products (Bhuptani and Moradpour, 2005). Moreover, ISO has developed RFID standards which are used in different frequency bands and for different RFID applications.

ISO 18000 to ISO 18000-6 series especially used for the air interface, ISO 14443 is used for proximity cards, and ISO 17358, ISO 17363, ISO 17364, ISO

17365 ISO 17366, ISO 17367, and ISO 17374.2 are used for supply chain management (Bhuptani and Moradpour, 2005).

3.3.6 RFID Applications

RFID is a promising technology that can be used in many applications, such as supply chain management, building access, human and animal implantation, libraries, transportation, health care management and e-passports (Henrici, 2008; Liu et al., 2010; Karmakar, 2010).

As the RFID system has different type of tags such as passive RFID tags, semi-passive RFID tags and active, the applications of using these tags are varied. Passive RFID tags do not require long scan ranges and apply multiple radio frequencies. LF passive RFID tags have a very short scan ranges and it can be used in tracking animals and embedded keys. High frequency passive RFID tags have also a short ranges for scanning and its been used for access control, library books pharrmaceuticals. In addition, NFC is another type of High-Frequency passive RFID tags wich is widely use with contactless payment. UHF passive RFID tags have the longest scan ranges and mostly use in inventory system, supply chain management, health care management, assest management personal tracking and patient tracking. In addition it can be used for anti-counterfeiting such as casion token and expensive goods.

MF passive RFID tags are the faster in data transfer and least common tags that are used in passive RFID system. They have used in applications that require fast transfer and less scanning range of items such as inventory tracking. Semi-passive RFID tags are use UHF and similar to passive RFID tags. As they have more computation power and memory, semi-passive RFID tags are mostly use in applications that require long scan range and sensors.

Such applications are tracking temperature in food, chemicals, medecin and other industrial products.

Active RFID tags provide the longest range in scanning, reliable in different environments and much more computations power and memory. Therefore , the applications of using active RFID tags are mainly used in military assist, logistic tracking and ocian containers and work on the frequency of 433 MHz. other application such as real time locations is worked on 2.45 GHz. Some other active RFID tags are carried sensors and can be used in temperature tracking, In addition, active RFID tags can work on UHF and can be used in logistic and container tracking and work on the frequency 915 MHz.

3.3.7 RFID in Supply Chain Management (SCM)

The term SCM means the management of goods flows; it consists of all parties that deal with customer request. The term SCM management was first developed in 1980 to integrate the key business process from supplier until customer. The main goal of SCM is to have a holistic view depending on effective cooperation between enterprises about the customer needs and get the best result of cost, time and efficiency (Liu et al., 2010).

The process of SCM system starts with orders of objects from the supplier then manufactured object or product assembly. The products are transported to the distributor then to the retailer and finally to the users. Suppliers are responsible for receiving correct quantity, quality, and prices of the material. Manufacturers are responsible for the inventory of raw material, the progress of work and the finished works. Distributors are responsible for transportation, shipping costs, vendor requirement. Retailers are responsible for goods distribution and

delivery; finally, customers are concerned with delivery time, return policy and the quality of goods (Chopra and Meindl, 2007)

The SCM parties may be national or global, although the level of technologies and security are different. Sharing information among SCM parties can be vulnerable to the SCM security systems. Information between SCM passes through all parties including supplier, manufacturer, distributor, retailer, and customer. Information needs to have security characteristics such as confidentiality, integrity, and availability. Information between parties must be accurate and the system is accessible in a timely manner and satisfied (Knorr et al., 2001), (Alkattan and Allkhudairi, 2008).

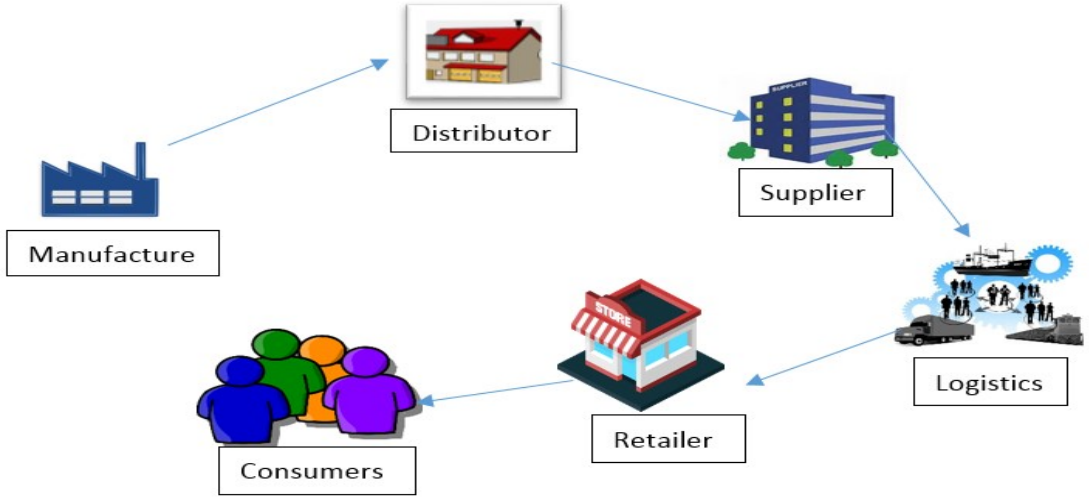

Figure 3. 2: RFID in supply chain management (Vaidya et al., 2012)

In recent years, RFID technology has been increasingly used in SCM due to its high offer in providing accurate information, visibility control in SCM and reducing labor cost. These advantages have taken place by using RFID in SCM. In 2003, Wal-Mart Company adopted RFID technology and ordered their supplier to implement RFID technology in their product. Moreover, the US Department of Defense (DoD) carried out efforts to develop RFID technology

and compel their suppliers over the world to tag their services by using RFID tag. Subsequently, large numerous organizations and suppliers have been launched RFID technology and thereby changed their SCM process. RFID technology has grown significantly in SCM, according to Bagchi et al. (2007), the prediction of RFID growth will be 20 billion Dollars while a billion Dollar in 2003 and 4 billion in 2008. (Masum et al., 2013)

The SCM plays a significant role in the success of company's strategy and depending on the adopted technology that can be used in their services. Technologies are used to achieve the purpose of SCM needs that information from supplier to distribute should be accurate in the identification and tracking of the object. In addition, information about the inventory should be accurate about the number, types, and conditions. Furthermore, this information is very critical for an organization because of the demands and pressures of factories, distributor, and retailer to maximize work process and minimize cost while providing good value for customers. One of the most popular of AIDC technology, which is still used in SCM is barcode technology. Barcode also supports automated data capture and is still being used in many applications of SCM due to the cost of the services in comparison with other services such as RFID technology.

RFID technology offers several advantages in traceability and identification due to its unique identification of project with the property of communication at distance and providing real-time information (Sayging et al., 2007; Michael and McCathie, 2005; Masum et al., 2013). Moreover, it can improve the accuracy of inventory system and visibility of the information flow in SCM process. Despite the cost, RFID offers many more advantages than barcode in the process of the SCM. (Alkattan and Alkudair, 2008; Liu et al., 2010) as shown in table 3.3

	RFID	Barcode
Line of sight	Not required	Required
Read range	Passive RFID 400 feet for fixed reader 20 feet for handle reader Active RFID up to 100 feet	Several inches up to several feet
Read data	10's, 100,s or 1000's significantly	Only one time
Identification	Can uniquely identify each item	Most barcodes only identify the type of item (UPC Code) but not item uniquely
Read/Write	Read/Write in most Tags	Read only
Technology	Radio frequency	Laser
Automation	Collect data automatically	Human operator

Table 3. 3: Comparison between RFID and barcode technology (Alkattan and Alkudair, 2008)

RFID has gained considerable attention from researcher to discuss the technology impact on inventory control management. According to Zeiimpekis et al. (2007), in supplying chain management RFID offers the ability to manage the identification of products in real-time information. This can correctly affect

and prevent error during the process of transportation management, inventory management, and order management. In addition, RFIDs will also help in taking advantage of automatic identification of objects, which reduce labour costs. Also, RFIDs can be used to reduce the problem of inventory inaccuracy (Kang and Gershwin, 2004). In comparison with the barcode, RFID technology offers more advantageous services than barcode technology. RFID is faster than barcode by 15-20 times, reduction in labor cost and reduction in time (Rekik, 2010). Alkattan and Alkudairi (2008) compared with two systems, one uses RFID technology in supply chain management, and one uses barcodes to compare the inventory level between the two technologies. The inventory data becomes accurate and easier to share with supply chain management, which helps to control the whole SCM system.

3.4 Security and Privacy Threats

The main issues emerging from the use of RFID systems are security and privacy. The needs of protecting data privacy from an attacker during the authentication process has increased because of the use of wireless communication between the RFID tag and RFID reader which is not secured. Moreover, in some cases, the wireless communication is also between the RFID reader and back-end server, which increases the demand for data security and privacy (Lee and Yi, 2011). Some possible threats and risks can affect RFID systems. The possible threats and risks to the system need to be determined, and then threats can be used to set the security requirements for the system (Jules et al., 2006; Peris-Lopes et al., 2006). In this section, two main types of threats which can affect the privacy and security of RFID system are

investigated. Moreover, possible solutions for solving the problem of these threats are investigated. The possible solutions which will be investigated are physical solutions and solutions in the authentication process.

3.4.1 Security needs

For providing security to information systems, there are three main aspects of security which are called CIA triad. The CIA triad is applicable across the whole system that subject to security analysis. CIA triad contains three main objects for security confidentiality, integrity, and availability. If penetration is done for one of these aspects then the system may have serious consequences for the parties concerned (Layton, 2006; Menezes, 1996).

Confidentiality: Confidentiality means the ability to ensure the protection of information from unauthorised access that can disclose the information. This service is the most part of any security system, but it is also the most aspect that can be attacked. Cryptographic methods can be used to ensure confidentiality of information.

Integrity: Integrity means the assurance of data accuracy and unchanged representation of the original secure information. Integrity ensures information so that it is not changed, deleted and copied.

Availability: Availability means the assurance of providing, storing and processing information from the system. A type of attacks to this service is denying access to the appropriate user.

3.4.2 Security Services and Characteristics

In addition to these three aspects of security, there are also some services that need to be provided for ensuring enough information security which are

authentication, access control, non-repudiation, scalability, and performance (ISO7498-2, 1989); Menezes, 1996)

Authentication: Authentication is the service that provides identification and validation of users during a communication session. Mutual authentication means the proof of authentication between two parties during a communication session.

Access control: Access control is the service that controls the flow of information also it provides protection against unauthorised access to so as to change read, write or delete information.

Non-Repudiation: Non-repudiation is a service that provides assurance of received or sent message. Parties can claim that the message has not been sent or received

Scalability: A network protocol is said to be scalable if the number of nodes can increase without forcing an unacceptable workload on any entity in the network.

3.4.3 RFID privacy Threats

User privacy is one of the most important concerns of the users of RFID systems. Tag's information can be disclosed due to unprotected communication through the wireless channel between a tag and a reader or a tag and a server (Jules, 2006). The most vital type of attacks RFID privacy occur due to tag information leakage and tag tracking.

- **Information leakage**

Typically, tag information leakage happens when an adversary gets information about the tag identifier through queries between a tag and a server. Unauthorised entities can obtain private information from the tag or from the

server database. This happens when there are no encryption protocols for the message exchange between the tag and the server (Okubo et al., 2003). For example, RFID tags can be attached to some passports or ID cards to store user's information. Using RFID system for labelling objects without a specific security mechanism makes RFID system vulnerable to disclosure. RFID system must have the ability to resist against tag information leakage threat by using encryption protocols, and protecting tag's data against unauthorized access (Gafinkel et al., 2005). Moreover, RFID system needs to have the ability to control tag's information by using protocols that allow only authorized access to access to the tag's information (Garfinkel and Rosenberg, 2006).

- Tracking

Tracking or traceability is the most commonly discussed privacy leaking property of RFID; (Okubo et al., 2003). Simply it can disclose a tag's location. Tracking happens when an attacker can join at the same time multiple tags' interactions in one point or more than the tag's location, which can be tracked by participating unauthorized entities (Najera and Lopez, 2008). For example, in a market when a person gets an object with an attached RFID tag and walks with a number of RFID readers controlled by the attacker. The attacker can deduce the person location at various points by linking reading protocols on different reader that is produced from the same tag. Thus, the susceptibility and the deduction of the person's location in time at various points can be done by the attacker. RFID system needs to have the ability to resist against the tag tracking attack by making protocols that provide an anonymous message from tags (Najera andLopez, 2008) . This can be done by using authentication protocols or provide a physical solution attached to the tag.

3.4.4 Attacks on RFID system

The nature of RFID systems makes the system vulnerable to several types of attacks. These attacks can prevent the system operation and expose RFID information. These types of attacks can be categorised into three categories.

3.4.4.1. Attacks on the singulation protocol between RFID reader and RFID tags

Singulation is the reader process for choosing the unique serial number of tag from a list of tags in the range of reader (Giusto et al., 2010). Readers need to have a protocol that can identify a unique tag to which it communicates at a time. The protocol of singulation is vulnerable to attack unless it is subject to protocol that limits the response process. The number of RFID tags that respond to the reader needs to be satisfied to avoid the collision in response to the RFID reader. The collision happens when two RFID tags or more respond together with a reader.

Anti-collision protocols have been developed for preventing collision during the process of identification between the RFID reader and RFID tags. Tree walking, ALOHA are common protocols that are used today to avoid collision (Banks, et al., 2007). ALOHA protocol is considered as the first singulation protocol which was invented in 1970. Usually, ALOHA protocol is mainly used for HF tags while tree walking protocol is used in UHF tags. During the communication between RFID reader and tags, tags are detected and send their ID to the reader in a specific time interval. If same data has been sent from different tags during the same time interval, then the collision has occurred, and attempt to resend after waiting a random time interval. The main problem with ALOHA protocol is the

time period for avoiding the collision which affects the efficiency of the protocol. Therefore slotted ALOHA was developed to solve the time problem of ALOHA protocol. Slotted ALOHA is mainly based on the same idea of ALOHA protocol but with more constraints on the tag's data. The tag's data is transmitted at synchronised time intervals called slots.

Tree walking algorithm is used for UHF RFID tags and run in a simple binary tree. A reader queries for each bit of tag identifier with either 1 or 0 to respond. If the reader receives more than one responds then the reader ask all tags with the serial number that starts with 01 and then 010 and so on. The tree walking protocol is an ideal method for searching tags but an attacker can eavesdrop the communication between the reader and tags.

3.4.4.2. Attacks on the communication protocols

The nature of the communication of RFID system that works via the insecure wireless channel is vulnerable to various types of attacks. Illegal RFID reader can exist in the range of RFID system from reader to tag (forward) or tag to the reader (backward). During the communication between the RFID reader and RFID tags, a closed adversary can eavesdrop both sides of the channel while a far located adversary can just eavesdrop the forward channel because the backward channel signal is weaker than the signal of forwarding channel (Ahson and Ilyas, 2008). The attacks of RFID communication can be classified into eight categories.

1- Denial of Service attack (DoS): In this type of attack, the attacker can cause a loss synchronization between a server and tags by blocking the transmitted message (Weis, et al., 2003; Lee and Yi, 2011). The attacker sends a large number of tag's identifier to the reader then to the back-

end server (Sandhya and Rangaswamy, 2011). This attack also can make a smashing to a server when receiving fakes request.

2- Replay attack: In this type of attack, the attacker can listen to the message exchanged between a server and tags then replay the query to the tag, reader and then the back-end server as a valid tag with a successful authentication process (Dimitriou, 2005; Wei, et al., 2011).
3- Man in The Middle attack (MitM): In this type of attack. The attacker interferes and listens to the communication between a server and a tag, then manipulate information by insert, modify, delete and redirect it (Jules, 2004).
4- Tag impersonation: In this type of attack, an attacker can communicate with a server instead of specific tag and be authenticated as a tag (Weis, 2003).
5- Location tracking attack: In this type of attack, the listing and analysing of the communication between RFID systems can track the location of a specific tag (Wei et al., 2011).
6- De-Synchronization attack: In this type of attack, an attack can prevent the information from reaching the reader or the tag when the update of information is sent from the back-end server. On the next session, the back-end server cannot authenticate the reader or the tag because no information validates the authentication.
7- Backward Traceability: In this type of attack, an attacker might be able to trace previous transactions between a service and a tag. this trace can be done by using the knowledge of the internal state of the tag and given all the internal state of the target tag at time T. The attacker can identify the tag's past transaction at time $T/$<T (Ohkubo et al., 2003)

8- Forward Traceability: In this type of attack, an attacker might be able to trace future transaction between a service and tags by using the knowledge of the internal state of the tag and given all the internal state of the target tag at tine T. In addition, the attacker can also identify the tag's past transaction at time $T^{/} > T$ (Lim and Kwon, 2006).

Moreover to these types of attacks, Song and Mitchell (2009) classified attacker into two groups; weak attacker and strong attacker.

A weak attacker has capabilities to observe and dominate the communication between a reader and a tag. Commonly, this attacker has the ability to modify, insert or delete messages that agree with the corresponding protocol's procedures.

A strong attacker has the same capabilities as the weak attacker but also the ability to compromise tag and access all tag's internal information.

3.4.4.3. Side Channel Attack

Side channel attack is a set of attacks that can analyse the behaviour of a device for the sake of learning about the device (Kasper et al., 2012). This attack is used to obtain information from the physical implementation of the cryptosystem. Additionaly, side channel attack can involve of measuring the time which is taken by a tag to respond to a valid request in order to capture the tag's information and to learn details about the tag from the response time. Additionally, a power analysis attack analyses the amount of energy that is spent by the tag during the computation. These attacks can involve different types of RFID layers from the physical layer to the application layer (Burmester and Demedeiros, 2007).

3.5 Conclusion

Although, RFIDs can enhance the competence of supply chain management there are also some issues that have to be considered, data, reliability and security challenges (Liu et al., 2010). For the data challenge, RFID tags can store much information, and RFID reader can read many tags in one second in comparison with barcodes. However, these facilities can also cause some effects to the original data, such as incorrect data or duplicate data or inaccuracy in an inventory system. Collision tags or tags that can be in readable form are also issues for the RFID systems since many readers can confuse the whole system. In addition, the security and privacy problem is the important issue that can affect RFID systems in supply chain management. Confidentiality, integrity, authentication, anonymity, and availability are critical issues in SCM.

4. Survey on Existing RFID Privacy and Security Solutions

4.1 Introduction

As described in the previous chapter, the widespread threats that emerge with the use of RFID technology is related to security and privacy. Researchers have introduced abundant numbers of RFID authentication protocols in order to moderate the security and privacy problems. Meanwhile, the proof of security and privacy ensures that the security and privacy requirements are to formalise the problem regarding adversary ability that can raise the vulnerability of RFID systems. Such security and privacy formalised definitions will be introduced in this chapter, followed by cryptographic authentication protocols that are based on solving the security and privacy in the single tag to reader authentication protocol, multi-tag or grouping tag authentication protocol and secret sharing schemes for RFID systems.

4.2 RFID Security and Privacy Models

Several models for privacy have been achieved in order to formalise privacy and security threats.

In models, there is an RFID system composed of a set of tags, single reader and a data base.

Legitimate tags information are stored in the system database. In all RFID systems, there are a series of composed procedures to set up a reader, tags

information and complete the interactive protocol between them. These procedures are as follows (Voudenay, 2007):

- InitiateReader (1^{λ}) a probabilistic algorithm to generate a pair of public and private key and a database for a reader.
- InitiateTag ($ID_{\mathcal{T}}$) is a probabilistic polynomial which returns a pair of the secret key of tag then store its ID alongside with the secret in the reader database.
- IDentify: A polynomial time interactive protocol which is executed between the reader and tag to identify legitimate tag $\mathcal{T}$ or reject the tag.

The ability of an adversary $\mathcal{A}$ to interact with the RFID system is varied. Such adversary can play the role of legitimate reader and interact with tags, intercept the message exchange between the tag and reader, access to the tag's information and can also access to the reader output.

Vaudenay 2007 formalise the adversary ability as follows:

- CreatTag($ID_{\mathcal{T}}$): This oracle is used to create a unique identifier and is used as a legitimate tag and Setup tag to add into the database.
- Launch($\mathcal{T}$) : This oracle is used to execute a new Identify protocol between the reader and tags.
- SendReader: This oracle is used to send a message m to the reader wait for protocol execution to output the response of the reader.
- SendTag: This oracle is used to send a message m to the tag and wait for protocol execution to output the response of the tag.
- ReTurn: This oracle is used to return the bit value of the verification from the reader.

- CorruptTag: Stores new information on the tag and output the secret of the tag.
- DrawTag: Selects tags randomly and gives new pseudonyms and output all new pseudenms.
- FreeTag: Makes the tag unavailable by moving tag with its pseudonym from the status of DrowTag to the status of FreeTag.

In any RFID system, a legitimate tag is the tag that corresponds to the information that is provided by the database. To ensure the security of an RFID system, there are two notions of security which are correctness and soundness. The correctness notion means that an RFID system always accepts legitimate tags by a reader. In other words, an RFID system is called to be correct if the probability of failing legitimate identification tags is negligible. Soundness notion refers to the denial of the database server of accepting illegitimate tags. An RFID system is called to have soundness notion if the probability of accessing illegitimate tags or impersonation tags to the database is negligible. Candara et al., (2010) and Vaudenay (2007) formalised the Soundness notions by the ability of an adversary to impersonate tag during the IDentyfy protocol. When the challenger initiates the reader and send the security notion 1^{λ} with the parameter and the private key to an adversary, then the adversary interact the RFID system and use the launch oracle at any time to output the protocol identifier π. The bit value b is returned by using ReTurn(π). The stong correctness notion is formalised by the ability of an adversary to interact with uncorrupted tag. When the challenger initiates the reader, sends the security notion 1^{λ} with the parameter and the private key to an adversary, then the

adversary interact with the RFID system and select uncorrupted tags identifier. At the final stage, the adversary uses the oracle Launch to output the return value b.

The Avoine model for privacy

Avoine (2005) proposed an adversary model that that conducts two notions of privacy Existential untraceability and universal untraceability. These two notions are based on the ability of an adversary to distinguish between two recognised tags as a challenge tags based on the protocol execution. The ability of capturing the challenge tags at any time is the notion for universal untraceability while the capturing of the challenge tags at a specific time is called existential untraceability. Since Avoine model focuses on two tags in the system, this model does not allow corrupted tag to be traced.

A modification model of Avoine was introduced by Juels and Weis. In their protocol, they introduced the notion of strong privacy and based on indistinguishability of tags. In their challenge model, an adversary allows to return the current secret of tag and allows the adversary to set new secret selects two challenge tags, and these tags should not be corrupted. The adversary selects two uncorrupted tag T_0 and T_1, then the challenger gives the adversary access to the challenge tag T_b . The adversary wins the challenge phase of strong privacy if the bit output $b = b'$. Thus an RFID system ensures strong privacy if the probability of winning the challenge phase in negligible.

The Voudenay Model

Voudenay (2007) introduced the most completed model for privacy as he defined and included the adversary classes such as strong, destructive, forward

and weak adversary. He also differentiated between the adversary ability in each class in terms of the using of corrupted oracle such as strong or wide adversary. The term of strong adversary refers to the unlimited access to the corrupted oracle by an adversary, while the destructive adversary has no access after the corruption of tag. The term of forward adversary refers to the using of corrupted tag by the adevrsary, while weak adversary cannot access to the corrupted oracle. Furthermore, Vaudenay added Result oracle that allows an adversary to decide if the adversary has completed the session successfully, which is called wide or narrow if the adversary failed. If the session does not complete, the adversary can access to the result oracle. Vaudenay also defined the privacy by introducing the notion of blinder and trivial adversary. Vaudenay considered the blinder as polynomial algorithms such that a blinder adversary knows nothing about the secret just simulate the communication oracles such oracles are Launch, SendReader, SendTag and Result oracles. Hence the blinder adversary does not use these oracles. A trivial adversary is called trivial if there exists a blinder such that the probability of adversary to win the output subtract the probability of blinder adversary to win the output is negligible. The privacy definition according to Vaudenay consists of two phases, attack phase and analyse phase. The phase one involves the adversary with the system through the oracles. The second phase of privacy depends on the ability of an adversary to analyse the given a table of the tag's information and output true or false. The protocol is called P-Private if the actual outputs are trivial. The Vaudenay privacy definition does not depend on the corruption of tags but depends on the information leakage during the communication between the RFID tag and the RFID reader. Vaudenay also showed that the statement of wide-strong privacy cannot be reached due to the ability of distinguishing

between the system and the blinder. While the statements narrow strong privacy can be reached if there is a key agreement protocol in the RFID system. The Voudenay model does not consider the case of using mutual authentication in RFID system. However, Paise and Vaudenay (2008) extended the case of privacy in mutual RFID authentication protocol.

Canard et al Model

Canard et al. (2010) introduced a model that builds on the Voudenay work which supports the same oracles. The privacy has been introduced as a non-obvious link by using dummy adversary instead of using blender. In their model, the ability of linking multiple authentication of the same tag leads to winning the game by an adversary. Linking multiple authentications is achieved if the adversary wins the game by successfully linking the pseudonyms through multiple sessions into the same tag. This notion is called non-obvious link. They defined the protocol to be untraceable in present or past or even in future if the probability of dummy adversary outputting non- obvious link subtracts the probability of blinder adversary outputting non-obvious link is negligible. Their model is similar to forward and backward secrecy because under corruption, an adversary can distinguish tag at present, past and future times.

Hermans et al model

Hermans et al. (2014) proposes a privacy model that is based on using the notion of simulation. The privacy is defined by the advantage of an adversary to win a privacy experiment between the simulation and the adversary. The experiment starts when the challenger picks a random bit b, then the adversary

interacts the system with oracles and output b'. The adversary wins if the $b' = b$. The adversary is to determine the value b based on the interaction with the oracle DrawTag(T_i, T_j). T_i and T_j cannot be redrawn unless the using of the Free tag oracle. If $b = 1$, then Drawtag $= T_i$ otherwise DrawTag=T_j. According to Hermans et al, An RFID system is said to be unconditionally provide privacy with the notion X, where X meaning all types of adversaries, if and only if for all adversaries there is no advantage in winning the privacy experiment.

4.3 Single tag RFID Authentication Protocol

Various RFID authentication protocols have been designed to overcome security and privacy problems. The objective for designing an RFID authentication protocol is not only for security and privacy, but it should be fitted for the ability of the RFID tags in terms of power consumption and memory store. Therefore, several researchers introduced different and various methods for implementing RFID authentication protocol.

4.3.1 Non-public key cryptography authentication protocol

Hash based access control

Weis et al. (2003) proposed a protocol based on using a one-way hash function to control the access of a tag by locking or unlocking the tag. In their protocol a back-end server stores unlocking keys for tags and each tag stores ID_i and $Hash(ID_i)$. In the authentication process, RFID tag answers temporary ID to all queries from random readers. Tag responds only with $Hash(ID_i)$ when it's

locked. The reader unlocks tag by sending K_i to the tag. If the value of $Hash(ID_i) = Hash(k_i)$ then the tag unlock its information. The storing of the unlock key in the back-end server prevent the secret key from information leaking. The authentication process can be summarised as follows:

1- The server sends query to the tag, then the tag replies with $Hash(ID_i)$.

2- Server check $Hash(ID_i)$, then sends K_i to the tag

3- If $h(k_i) = Hash(ID_i)$, then unlock the value of the tag.

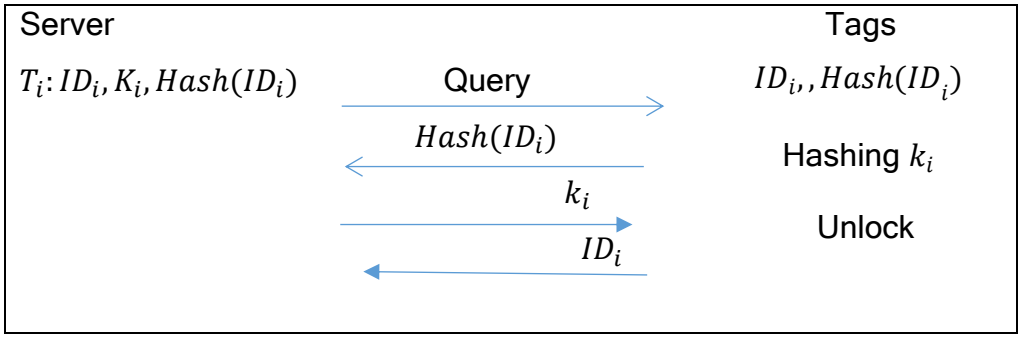

Figure 4. 1: Hash-based access control protocol

However, this protocol does not protect tag from tracking because of the fixed temporary ID repeatedly used in different sessions of authentication. Also, this scheme is vulnerable to replay attack, an adversary can impersonate a tag by temporary ID_i (Osaka, et al, 2006).

Randomly access control

Weis et al., 2004 proposed a protocol which is based on using a one-way hash function with a random number generator. Their protocol is based on the same idea of hash based protocol but using random number generator can prevent ID_i to be replied.

The authentication process can be summarised as follows:

1- Server stores ID_i for each tag T_i where T_i stores ID_i.

2- T_i generates a random response with different values in each session by making the pair of random number generator with the hash value of ID_i concatenated with a random number.

3- The server identifies the tag by total search to compute the hash value of $(ID_i//r)$, where r is a random number

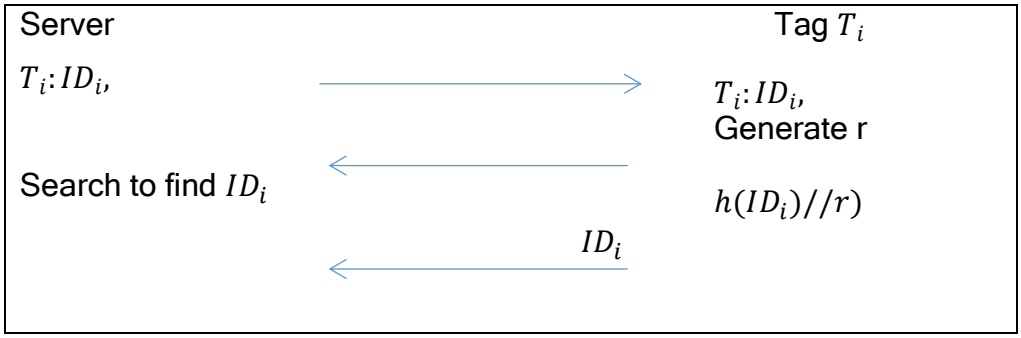

Figure 4. 2: Randomly access control

However, their protocol can be impersonated by the tag impersonated attack and a replay attack (Osaka, et al, 2006), . In addition, the fixed ID can be traceable and interceptor response can reply which leads to backward untraceability.

Henrici and Muller Protocol

In 2004, Henrici and Muller (2004) proposed a hash-based varying identifier protocol, which is based on using a one-way hash function and conjunction operation to protect the privacy of RFID tags. In their protocol, the shared secret keys are updated after the authentication process and the back-end server stores the hash value of tag's ID in order to accelerate the search process.

Basically, tags store the value of $ID_i, Hash(ID_i)$ transaction number TID_i and last successful transaction number LST_i. In order to update tag values andID_i, two hashed value of tag response are sent to the reader query during the authentication process. Tags usually reply with the same hashed ID which means that an adversary can trace the tag before the next authentication session (Chien and Chen, 2007).

The authentication process can be summarised as follows:

1- After the sent query from the reader, tag increases the TID_i by one, then computes $HID_i \; and \; \Delta TID = \; TID_i - LST_i$.

2- Tag calculates the value of $M1 = (TID_i \oplus ID_i)$ and sends the set of $\{HID_i, \Delta TID \; and \; M1\}$ to the server.

3- Server matches the set of $\{HID_i, \Delta TID \; and \; M2$ to find identification of the tag.

4- After satisfying the identity, server generates a random value r with the message $M2 = (r \oplus TID_i \oplus ID_i)$ and sends message to the tag.

5- Tag checks $r, m2$ if it satisfys the value of m2 then tag update its $ID_i and \; LST_i$ as $ID_i \oplus r$ and $LST_i = TID_i$.

Server	Query	Tag
Stores$[ID_i, HID_i, TID_i, LST_I]$	$\longrightarrow$	Stores $[ID_i, TID_i, LSt\_i]$
Find HID_i	$\longleftarrow$	$TID_i \Leftarrow TID_i + 1$
Compute TID_i from ΔTID	$HID_i, \Delta TID \ and \ M1$	$M1 = (TID_i \oplus ID_i)$
if $M1 = (TID_i ID_i)$	$\longleftarrow$ $M2$	
Generate r $M2 = (r \oplus TID_i \oplus ID_i)$ $update\ ID_i \leftarrow ID_i \oplus r$ $LTD_i \leftarrow TID_i$	$\longrightarrow$	Checks if $M2 = (r.TID_i.ID_i)$ $ID_i \leftarrow Id_i \oplus r$ $LTD_i \leftarrow TID_i$

Figure 4. 3: Henrici and Muller Protocol

According to Chein and Chen (2007), this scheme cannot provide backward untraceability because an attacker can do the computing of the identifiers used in the previous session.

OSK protocol

OSK (Ohkubo-Suzuki-Kinoshita) protocol was proposed in 2003 for privacy protection. Their scheme is based on using one-way hash function to update the tag identifier in each query between the RFID tag and the RFID reader. Each tag stores a random identifier S_i^1. During the authentication process, reader queries the tag, then the tag will send its hashed identifier number H1 and uses second hash function H2 to update the identifier number.

The OSK scheme proposed a backward untraceability requirement which blocks the identification of tag in the past sessions of communications. Using hash chain in this scheme prevents an adversary from revealing the identification of the tag from the past secret key (Osaka et al., 2006).

The OSK protocol can be summarised as follows:

1- Reader sends identification query to tags and receives $r_i^k = H_1(s_I^k)$.

2- Each tag replaces S_i^k by $s_i^{k+1} = H_2(S_i^k)$.

3- The server checks the value of tag identifier by computing the $H_1(H_2^j(S_i^1)$, where j is the number of function iteration between two updates.

Server $[ID_i, S_i^1]$		Tag $[s_i^k]$
For each ID_j make iterations with check $r_i^k = H_1(H_2^j(S_j^1)$	ID query → $[r_i^k]$ ←	Computes $r_i^k = H_1(s_I^k)$ $s_i^k = H_2(S_i^k)$

Figure 4. 4: OSK protocol

However, this protocol is vulnerable to replay attacks by using communication messages to impersonate a tag without knowing the secret key. Moreover, this scheme is considered as an unscalable scheme due to numbers of tags in the lists (Osaka et al., 2006; Chien and Chen, 2007).

Avoine and Oechslin protocol

Avoine and Oechslin (2005) proposed a technique based on the OSK protocol to solve the scalability issues and enhance the privacy of OSK protocol. Their protocol is based on using hash function and aimed to reduce the complexity of the OSK protocol by using a novel time-memory trade-off which is reduced the workload on the server tag identification.

The authentication process can be summarised as follows:

1- Tag queried by a reader with fixed public string w and nonce r will send $a_i = H_1(s_i \oplus r)$ to the reader and renews $S_i + 1 = H(s_i)$.

2- Server checks a_i if it satisfies the value then calculate $H_2(S_{i+1} \oplus w)$ to the tag.

Server $[ID_i, S_i^1, w]$		Tag $[s_i^k, W]$
For each ID_j make iterations with check $H_2(S^{i+1} \oplus w)$	ID query → ← →	Computes $S^i + 1 = H_1(s_I^i)$ $H_2(S^{i+1} \oplus r)$

Figure 4. 5:Avoine and Oechslin protocol

Their protocol provides a modified identifier for improving privacy. The protocol uses a fixed string and nonce r to query tag during the authentication session which prevents a replay attack. (Osaka et al., 2006)

Dimitriou protocol

Dimitriou (2006) proposed a mutual authentication scheme for providing privacy and protect tag cloning. His scheme is based on the use of a one-way hash function that can guarantee the untraceability of past communication session and the server stores the hash value of tag's ID to make the identification process efficient.

The protocol can be summarised as follows:

1- Server sends a random nonce r_1 to the tag through the reader, then the tag will generate another random nonce r_2.

2- Tag evaluates $H(ID_i)$ and $M_1 = f_{ID_i}(r_2 \parallel r_1)$, where f is keyed hash function.

3- Tag sends a set of $\{r_2, H(ID_i), M_1)\}$ to the server through a reader.

4- Server finds $H(ID_i)$ in the data base list to check if the value $H(ID_i)$ is in the list then checks the value of $M_1 = f_{ID_i}(r_2 \parallel r_1)$ also.

5- Server replaces ID_i with $H_2(ID_i)$ and computes$M_2 = f_{ID_i'}(r_2 \parallel r_1)$.

6- Tag receives M_2 and calculates $ID_i' = H_2(Id_i)$ to satisfy the equation. Then tag authenticates the reader and updates $ID\ as\ ID_i \leftarrow\ ID_i'$.

Server		Tag
$ID_i, H_1(ID_i)$	Query	ID_i
	r_1	Generate r_2
Search $H_1(ID_i)$ in the server's list	$r_2, H(ID_i), M_1$ ←	Computes $H(ID_i)$ $M_1 = f_{ID_i}(r_2 \parallel r_1)$, calculates $ID_i' = H_2(Id_i)$
$ID_i' = H_2(Id_i)$ Calculate$M_2 = f_{ID_i'}(r_2 \parallel r_1)$.	$M_2 = f_{ID_i'}(r_2 \parallel r_1)$ →	$M_2 = f_{ID_i'}(r_2 \parallel r_1)$. $ID_i \leftarrow ID_i'$.

Figure 4. 6: Dimitriou protocol

This scheme is subject traceability threat as the Henrici and Muller protocol; an adversary can randomly query the tag to reveal the fixed hash value of the ID. Moreover, if the last message from the server cannot reach the tag, then the server will update tag identifier and that can cause DoS attack.

Lim and Kwon Protocol

Lim and Kwon proposed a mutual authentication forward untraceability scheme which protects tag identification in future communication sessions (Lim and Kwon, 2006). Their mutual authentication scheme is identified as linear search work. Their protocol provides untraceability and backward untraceability by using three pseudorandom functions. In the authentication process, both reader and tag update their shared secret key by using the old secret key blended by two random numbers exchanged in the authentication session.

Their protocol uses three pseudorandom functions, $f: \{0,1\}^l * \{0,1\}^{2l_1} \rightarrow \{0,1\}^{2l_1}$, $g: \{0,1\}^l \rightarrow \{0,1\}^l, and\ h: \{0,1\}^{2l_1} \rightarrow \{0,1\}^{2l_1}$, where l is the bit-length of a tag

secret and l_1 is the bit of length of random response. Usually, a server stores current data set and old data set for each tag T_i. The current data set contains tag identification which contains random secret s_i, m identifiers $t_i^j = ext(g^j(s_i),)$, where$0 \leq j \leq m-1$, a random number u_i for backward key chain, the length v_i of the backward key chain, and two secret for server validation $w_i, s = h^{v_i-1}(u_i)$ and $w_i, T = h(w_i, s)$, where m is the maximum number of allowable authentication failures between two valid sessions, $ext(x, l)$, denote a simple extract function returning l bits out of x, g^jdenotes j iteration of the function g, and l^2 is the bit length of a tag secret sent by the tag to help the back end server to identify tags. The tag stores the tag secret s_i, the server validator w_i, T, a failure counter c_i and the maximum number of the counter m, where c_i is initialised to 0. When the authentication process completes successfully, the tag secret s_i is refreshed within both the tag and the server by using exchanged random numbers r_1, r_2 and a secret for server validation$s_i \leftarrow g(s_i \oplus (w_i, s \parallel r_1 \parallel r_2))$. If authentication fails, then a tag updates its stored tag secret s_i to $g(s_i)$.

Backward untraceability can be ensured by the pseudorandom number function and use of hash key chain makes it difficult to impersonate a server to tags. Forward untraceability can be achieved if an adversary misses one message from successful authentication session. However, the purpose of preventing denial of service attack can lead to another attack that allows an adversary to identify the tag without corrupt the tag (Ouafi and Phan, 2008).

Server		Tag
$[T_i: S_i, u_i, v_i, w_i, s, w_i, T, s_i'$		
Generate r_1	r_1 ⟶	Generate r_2
		$ext(s_i, l_2)$
		$ext(f(s_i, r_1 \parallel r_2), l_1)$
Find $t_i^J = ext(s_i, l_2)$	$ext(s_i, l_2)$	
Computes	$ext(f(s_i, r_1 \parallel r_2), l_1)$	
$s_i' = g^j(s_i)$	r_2 ⟵	
Satisfying		
$ext(f(s_i, r_1 \parallel r_2), l_1)$		
Computes	$f(s_i', r_2 \parallel r_1) \oplus w_i, s$	
$f(s_i', r_2 \parallel r_1) \oplus w_i, s$	⟶	
$s_i' \leftarrow g(s_i)$		$w_i, s \leftarrow f(s_i, r_2 \parallel r_1)$
$t_i'^k \leftarrow ext(g^k(s_i'), l_2)$		Verifying $w_i, T = H(wi, s)$
$v_i' \leftarrow v_i$		If OK update $c_i \leftarrow 0$
$w_i', T \leftarrow w_i, T$		$s_i \leftarrow g(s_i \oplus (w_i, s \oplus r_1, r_2))$
$w_i', s \leftarrow w_i, s$		$w_i, T \leftarrow w_i, s$
$s_i \leftarrow g(s_i' \oplus (w_i, s \parallel r_1, r_2))$		Else
$t_i^k \leftarrow ext(g^k(s_i), l_2)$		$c_i \leftarrow c_i + 1$
$n_i \leftarrow v_i - 1$		$s_i \leftarrow g(s_i)$
$w_i, T \leftarrow w_i, s$		
$w_i, s \leftarrow h^{v_i}(u_i)$		

Figure 4. 7: Lim and Kwon protocol

The Chien and Chen protocol

Chen and Chen protocol proposed a mutual authentication protocol based on using pseudorandom number generator and Cycle Redundancy Code (CRC). Their protocol is proposed to be used with EPCglobal Class-1 Gen 2. The back end server stores new and old session keys to prevent denial of service attacks also their protocol prevents backward untracebility. However, their protocol can

be affected by denial of service attack, forward tracing and tag impersonation attacks (lopes et al., 2011).

In this protocol server stores $[EPC_i, K_i, P_i, K_i', P_i']$ and tag stores $[EPC_i, P_i, K_i]$, where EPC_i is the EPC identifier and P_i, K_i are the access and the current authentication keys and P_i', K_i' is the old access and authentication access.

The authentication process can be summarised as follows:

1- Server queries tag by a random number r_1, then tag replay with another random number r_2.
2- The tag also computes the message $M_1 = CRC((EPC_i \parallel r_1 \parallel r_2) \oplus K_i)$ and sends it as a response of the query.
3- After receiving the value of M_1, server checks whether the value of M_1 is equal to $M_1 + k_i \; or \; M_1 + K_i'$.
4- If the value matches then sever computes the value of $M_2 = CRC(EPC_i \parallel r_2) \oplus P_i$, or $M_2 = CRC(EPC_i \parallel r_2) \oplus P_i'$.
5- The server updates the secret value as $K_I' = K_i \; and \; P_i' = P_i . K_i = PRNG(K_i) \; and \; P_i = PRNG(P_i)$.
6- The server sends M_2 to the tag and the tag updates the secret value to complete the authentication process.

However, lopes et al (2009) showed that their protocol can be affected by tag and reader impersonation and de-synchronization attack. Also, their protocol is vulnerable to tracing attacks and forward security.

Song and Mitchell protocol

In 2008, Song and Mitchell proposed an RFID authentication protocol. In their scheme, s server stores secret u_i and t_i for each tag T_i as well as the most

recent secret $u_i^{/}$ and$t_i^{/}$. Secret u_i is a string of l bits assigned to T_i, and t_i is a hash of u_i. A tag keeps the value of t_i as its identifier. This scheme uses a hash function to update the secret t_i, a keyed hash function f to protect the message and a combination of simple function such as right and left shifts and a bit-wise exclusive or operation to combine data string. When an authentication session complete successfully, the server updates the secret values of $(t_{i,}u_i)\ and\ (t_i^{/},u_i^{/})\ )\ for\ T_i$, and the tag also updates its secret t_i using hash function.

Reader		Tag
$u_i, t_i, u_i^{old} t_i^{old}$		t_i
	$\longrightarrow$ r_1	Generate r_2
Search t_i^{old} t_i		Computes $M_1 = (t_1 \oplus r_2)$
		$(M_2 = r_2 \oplus r_1)$,
Computes	M_1,M_2 $\longleftarrow$	C=CRC($M_1 \oplus r$)
$M_3 = u_i \oplus (r_2 \gg \frac{l}{2})$		
Updatde t_i^{old}, u_i^{old}	$M_2 = f_{ID_i'}(r_2 \parallel r_1)$	Checks:
$u_i = (u_i \ll l/4) \oplus (t_i \gg$		$h(M_3 \oplus r_2 \gg l/2)) = t_i$
$l/4) \oplus r_1 \oplus r_2$		updates
$t_i = h(u_i)$		$t_i = h(u_i \ll l/$
		$4) \oplus (t_i \gg l/4) \oplus r_1 \oplus$
		r_2

Figure 4. 8: Song and Mitchell protocol

Shaoying (2009), showed that the Song and Mitchell protocol enables an adversary to impersonate any legitimate reader or tag which is vulnerable the protocol to both tag impersonation attack and reader impersonation attack.

Chia-Hui, Min-Shiang, and Augustin Protocol

Chia-Hui, Min-Shiang, and Augustin (2011) proposed a mutual authentication scheme. Their scheme is based on a using hash function and pseudorandom number generator. When a reader queries with a tag, reader sends a random number to the tag then tag sends its hashed random number by using a shared secret key between the tag and a back-end server. The reader uses a random number and hashed its ID within the data of tag and sends the value to the back-end server. The backend server verifies the tag and hashes the tag's ID with its own random numbers then generates new secret key, stores the old secret key, and send the hash value of the tag's ID and its random as they assumed, the protocol is secure against tracking attack, cloning attack, replay attack, forward security and DoS attack (Chia-Hui, Min-Shiang, and Augustin, 2011). However, there is no authentication between the reader and the back-end server. Although the reader sends random numbers, it does not prevent an illegal reader from penetrating the backend server and act as a trusted reader. In addition, the proposed protocol does not prevent or recover from the de-synchronization attack.

4.3.1.1. ECC RFID authentication scheme

In 2005, Wolkerstorfer (2005) introduced and discussed the concept of using elliptic curve cryptography within RFID system and the feasibility of ECC. However, the author did not propose any specific authentication scheme.

Tuyls and Batina (2006) proposed an identification scheme based on using Schnorr identification protocol, which is a zero-knowledge proof on elliptic curve discrete logarithm problem ECDL. They proved that their protocol can resist

against a passive attack such as counterfeiting and replay attack (Lee et al., 2008). The protocol starts when the RFID tag picks a random number r and sends $x = \propto^r \ mod\ p$ to the reader as a commitment. After receiving the commitment, the reader then selects a random number e as a challenge to the tag. The tag responding to the challenge by computing $y = a.e + r\ mod\ q$ and send it to the reader. After receiving y, the reader computes $z = \propto^y . v^e mod\ p$. If the the value of $x = z$, then the authentication success.

Lee et al. (2008) proved that the Tuyls and Batina protocol suffered from tracking attack and cannot provide anonymity. Also, this protocol cannot provide forward security and suffered from scalability problem.

In 2007, Batina et al. (2007) proposed Okamoto's identification RFID protocol which is also based on using on ECDLP. They proved that their protocol can resist against active attack,. Batina et al 2007, protocol is described in figure 4.10.

The protocol is also based on zero knowledge protocol, in their scheme the tag selects two random numbers and computes $x = \propto^{-k1} \propto^{-k2} (mod\ p)$ then sends the value to the reader. After receiving these values, the reader sends a random number e and sends it to the tag as a challenge.

The tag computes and sends $y1 = k1 + a1e \pmod q$, $y2 = k2 + a2e \pmod q$ to the reader. The authentication success if $z = \propto_1^{y_1} \propto_2^{y_2} v^e\ (mod\ p)$.

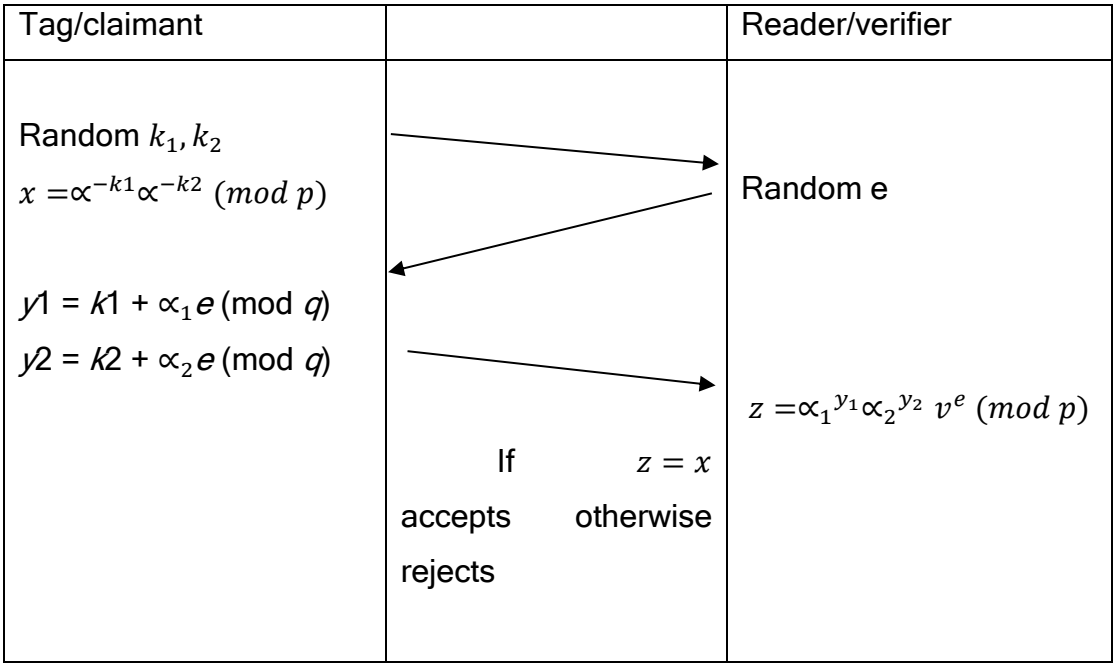

Figure 4. 9: Batina et al protocol (Batina et al, 2007)

Lee et al (2008) presented that Batina et al protocol has issues with location tracking and forward attack.

In 2007, Mcloone and Robshaw (2007) implemented an authentication protocol based on using GPS identification protocol (Girault, Poupard and Sterm protocol) which is a version of zero-knowledge proof of elliptic curve. The idea of GPS protocol is similar to Schnorr protocol in term of the zero-knowledge proof but GPS protocol "does not require knowledge of the order of the group nor the group element" (Mcloone and Robshaw, 2007). The protocol starts generating $r_1 \in [0, A-1]$then computes $x = g^{r1}$, then sends the value to the verifier. The verifier sends a random challenge c, where $c \in [0, B-1]$ and A, B are integer number. After receiving c, the prover calculates $y = r1 + sc$. The verifier then checks $I \in L\ such\ that\ g^y x^{-1} = I^c$ and $0 \leq A - 1 + (B - 1)(S - 1)$. Complete these calculations then the identifying is achieved.

In their protocol, they proved that the GPS scheme can resist against passive attack. However, the authentication protocol does not provide privacy.

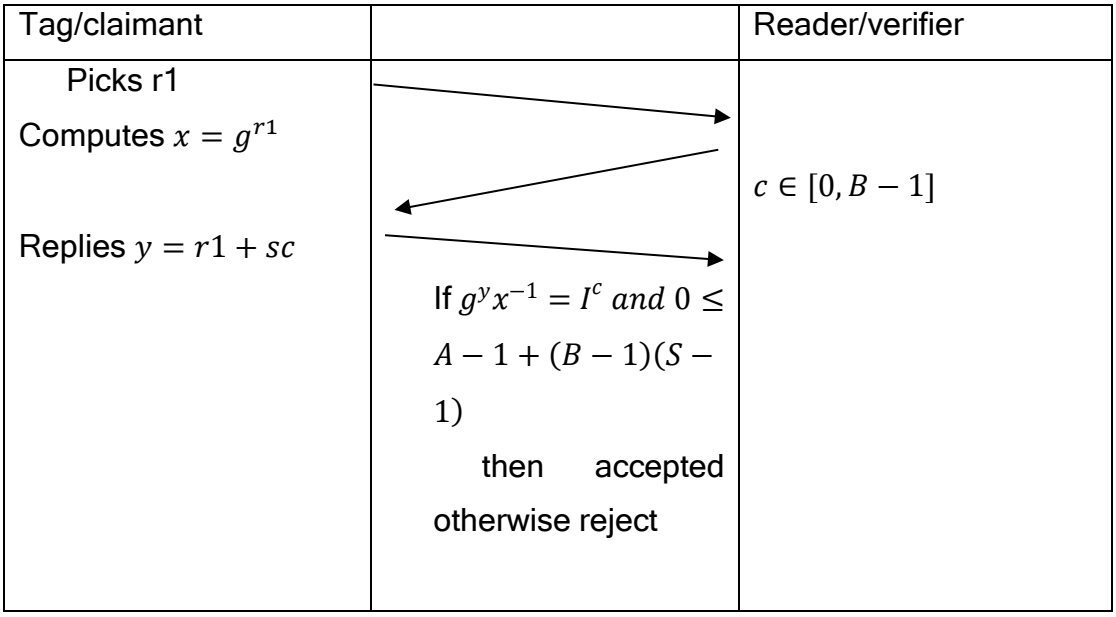

Figure 4. 10: GPS protocol (Mcloone and Robshaw, 2007)

In order to provide privacy to the GPS protocol, Bringer et al. (2009) also proposed the randomized GPS to ensure privacy. Similar to the GPS protocol, the randomized GPS protocol starts generating a private and a public key for the claimant and the verifier where the secret key for the claimant is $s \in [0, S-1]$ and the public key for the claimant is $I = g^s$. For the verifier, v is the secret key and $g^v$is the public key. The protocol is shown in figure 4.11

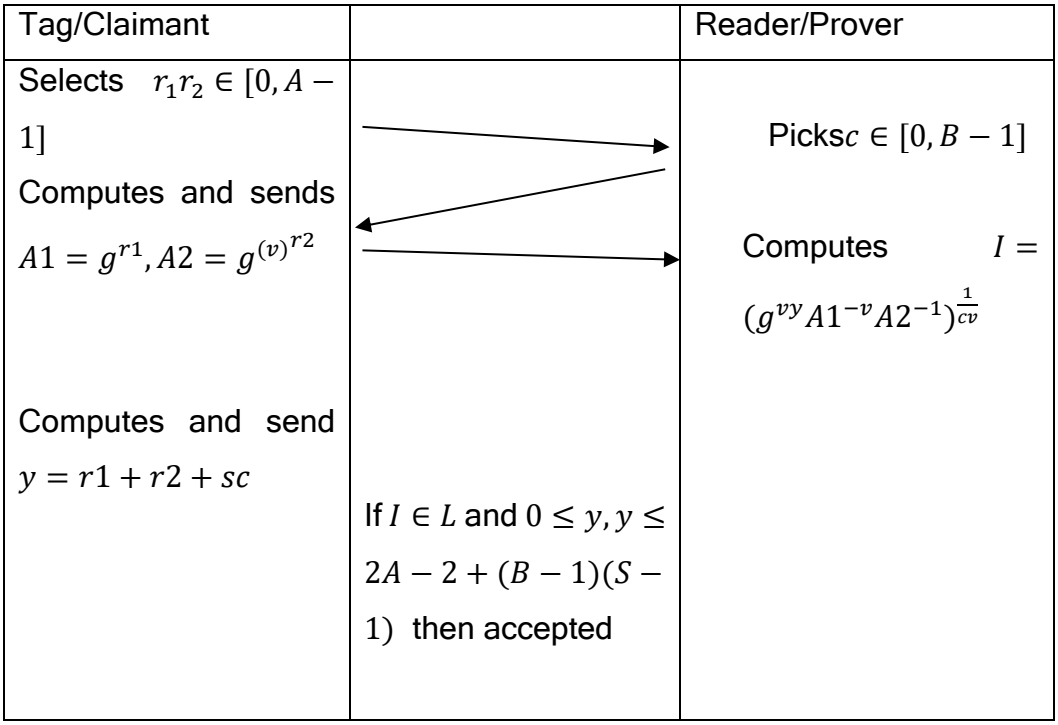

Figure 4. 11: The randomized GPS protocol (Bringer et al, 2009)

The difference between this scheme and the original GPS protocol is that the privacy here insured and the calculation of $A2^v$ ensuring that the verifier can make the final verification. Since the verifier performs constant numbers of operation without slowing down the identification process. This protocol is considered as a scalable protocol.

Bringer et al (2009), proposed randomized hashed GPS protocol which is a zero-knowledge proof protocol. This protocol is similar to the previous protocol and provides security against active attack. Also, this protocol can ensure privacy even the data is corrupted. The only difference here is using the hash function to the first message, and the calculation of this protocol can be done off-line with the property that all information can be revealed. The protocol is shown in figure 4.12.

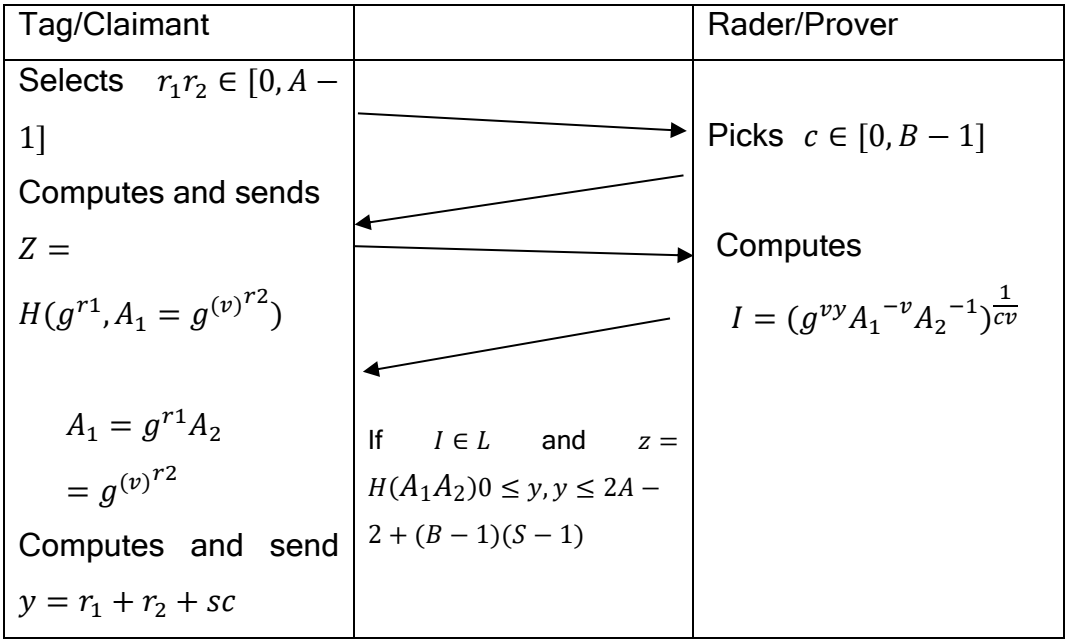

Figure 4. 12: the randomized hash GPS protocol

In 2008, Lee et al proposed an elliptic curve called EC-RAC (Elliptic curve Random Access control) protocol. This protocol is aimed to enhance security and provide anonymity. However, Bringer et al (2009) broken the protocol and found that Lee *et al.*'s scheme cannot resist to tracking attack as well as the tag impersonation attack. Furthermore, lee et al. (2010), proposed a second version called EC-RAC II to solve these problems. However, Deursen and Radomirovic (2009) proved that the EC-RAC II suffered from man in the middle attack and tracking attack. Lee et al.. (2010) proposed a new scheme called EC-RAC III to avoid the security problem of EC-RAC and EC-RACII. According to Fan et al. (2010), The EC-RACIII protocol is still not secure against man in the middle attack. Later, Lee et al. (2010), proposed a final version of EC-RAC family called EC-RAC IV, this protocol is proposed to solve the privacy problem. However, Deursen and Radomirovic (2010) pointed out that EC-RAC IV still vulnerable to the man-in-the-middle attack.

Bringer et al. (2008), proposed the randomized Schnorr protocol to solve the privacy issued by the original Schnorr protocol. The randomized version of

Schnorr protocol ensures the privacy of by computing $A_2 = \beta vP$ to ensure that only the prover can verify and compute the identity.

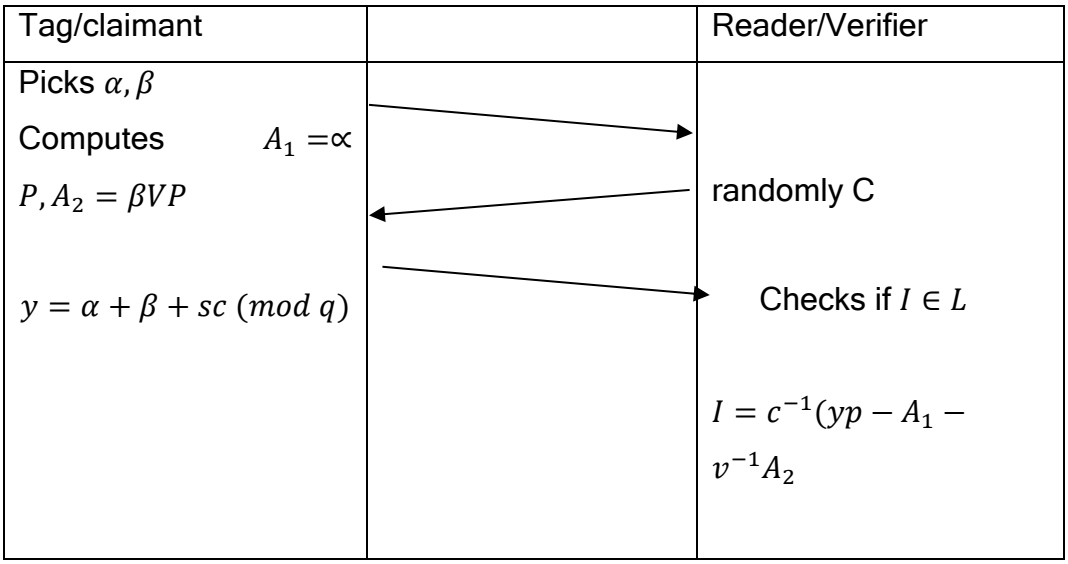

Figure 4. 13: Randomized Schnorr protocol

In 2009, Martinez et al. proposed an authentication protocol that is based on using zero knowledge prove and ECC. In their scheme, they worked on the finite field of 137 to make the work faster than other. They proved that Schnorr protocol is secure against relay attack and man in the middle attack. In the case of DoS attack, the author assumed that there is no danger against DoS attack. This is because tag only changes its secret key when authentication successfully. In the case of tracking tag, the only information to be considered is random number challenge. According to Lv et al. (2011), Martinez et al scheme is vulnerable to tracking attack.

In 2011, Zhang et al., (2011) proposed two modifications to improve EC-RAC and Schnorr protocol. Their scheme is aimed to resist to tracing attack. However, Babaheidarian et al., (2011) proved that the impersonation attack could affect Zhang et al. schemes.

Chen et al. (2011) proposed a based RFID authentication protocol based on using ECC. Their scheme aimed to overcome security and privacy issues of

RFID system and to be more efficient than EC-RAC IV. In their scheme, the server responsible for generating, random numbers and time stamp. The RFID tags run ECC computations, random numbers generator, and a hash function for the tag's ID. He and Zeadally (2015) pointed out that Chen et al. protocol is suffered from a replay attack. The Chen et al. protocol is shown in figure 4.14

RFID tags $Id_i = d_iP, P, Y, t_i$		Server $Y = sP, P$
check the value of time stamps t_s and $t_i \in Z_q$ Generate random r & k Compute $C_1 = r_s kP$ $C_2 = r_i + r_s kY$ Compute $x_1 = h(r_i, C_1)$ $x_2 = h(r_i, C_2)$ Check the value of $\gcd(x_1, x_2) = 1$ Compute $BB_1 = x_1 h(Id_i)$ $B_2 = x_2 h(Id_i)$ $B_3 = h(r_i, C_1, r_s)$	←	Generate random r_s, time stamp t_s
	→	$r_i = C_2 - sC_1 = B'_3$ Checks the value $B'_3 = B_3$ Compute $x_1 = h(r_i, C_1)$ $x_2 = h(r_i, C_2)$ Check the value of $\gcd(x_1, x_2) = 1$ Find $k_1 and\ k_2$ by using Euclidean algorithm $h(Id_i) = k_1 B_1 + k_2 B_2)$

Figure 4. 14: Chen et al. (2011) protocol

Liu et al. (2013) proposed a lightweight ECC- based authentication protocol (LRAP) aimed to reduce computation cost over RFID tags by reducing operation over RFID tag and put the high operation over the RFID reader. Their scheme is aimed to provide security and mutual authentication by using elliptic curve digital logarithm problem and to avoid using the digital signature. Their system consists of four phases that starts with initial phase, tag identification phase, mutual authentication phase and updating phase. Although Liu et al protocol is produced as a lightweight authentication protocol; the tag produces a number of operations that increase the complexity and efficiency of the protocol as shown in figure 4.15. Therefore, their system is not applicable in a real RFID application. Together with the complexity of the protocol, Liu et al protocol is lack of providing anonymity and tracking attack (He and Zeadally, 2015).

RFID tags		Reader
$Id, oldId, newId$ $dncryption\ key\ dk$		$Id, encryption\ key\ ek$
Sends tag Id	→	Generates random numbers (n_1, n_2, n_3) Computes (c_1, c_2)=$n_3 ek$ $cipher1 = c_1 n_1$ $cipher2 = c_2 n_2$ $cipher3 = n_3 P$ Computes $A = (Id + n_1 + n_2) \oplus ek$ Sends $cipher1, cipher2, cipher3 \| \| A$
Computes (c_1, c_2)=$dk.cipher3$ $n_1 = cipher1.{c_1}^{-1}$	←	

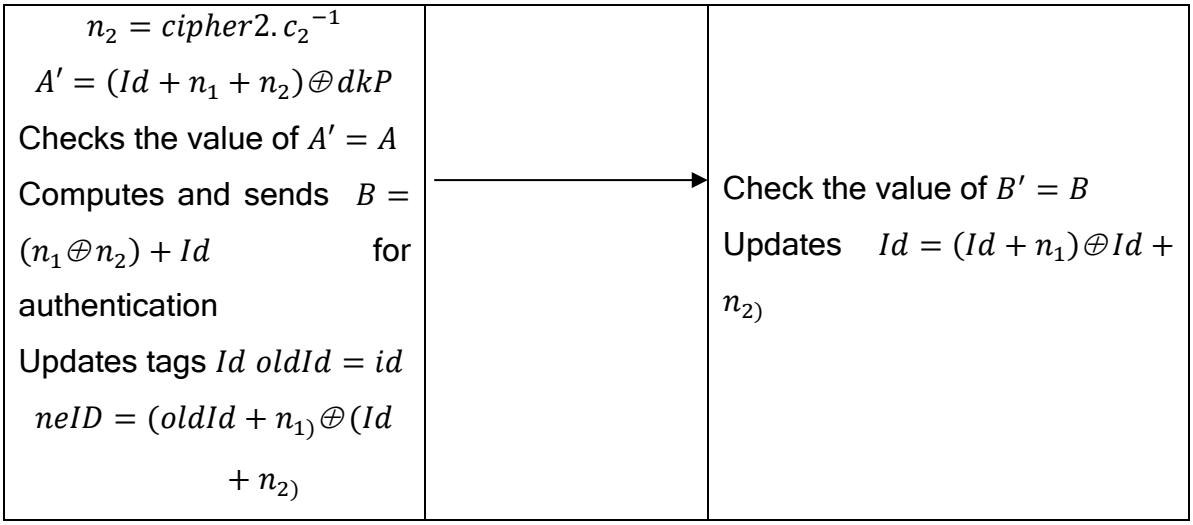

Figure 4. 15: Liu et al (2013) protocol

Wang et al. (2014) proposed an authentication protocol that is based on using ECC and hash function to achieve backward secrecy. In their scheme the server is responsible for generating system parameter and store the secret and public key values in database and tag's memory. Their protocol is aimed at providing mutual authentication, however, the protocol receives random number from the back-end server and cannot authenticate with the data-base server also their protocol is not scalable with large numbers of RFID tags.

Wang et al authentication procedure is shown in figure 4.16

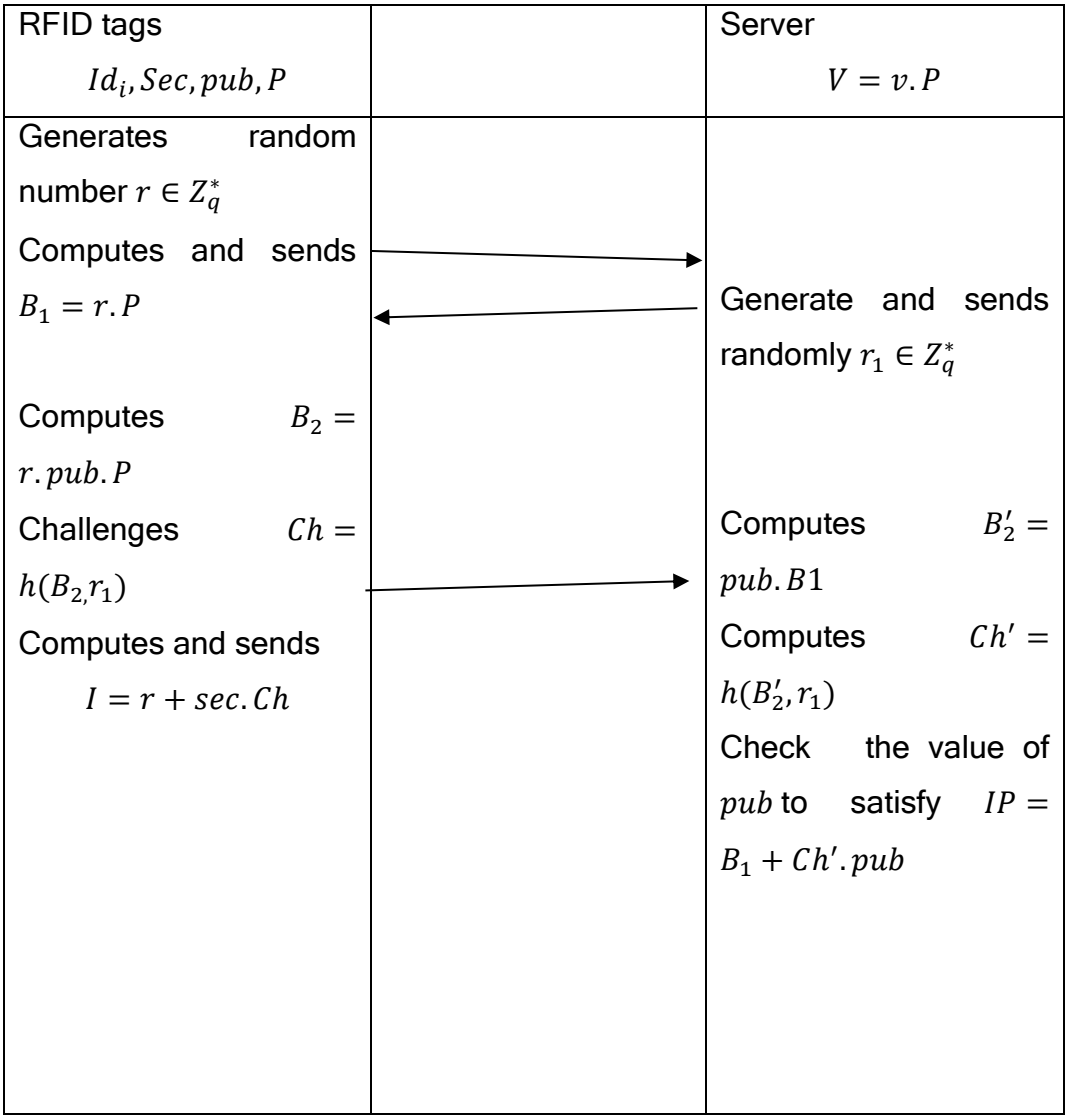

Figure 4. 16: Wang et al. (2013) protocol

Songhela and Das (2014) proposed an authentication protocol that is based on using ECC and a pseudo-random function. Their scheme is aimed at providing a strong privacy preservation and to provide a mutual authentication with a backward and forward secrecy. However, Ryu et al (2015) showed that Songhela and Das protocol does not provide forward secrecy and vulnerable to replay attack and impersonation attack. Songhela and Das (2014) authentication protocol is shown in figure 4.17

RFID tags sec_i, pub, P		Server sec_i, P, pub
Generates random numbers $r, k \in Z_q^*$ Computes a pseudo-random function $f(r, k.pub.P)$, $K = kP$, $B_1 =$ $f(r, k.pub.P).sec_i pub.P$ Sends r, K, B_1	→	Computes $f'(r, k.pub.P)$, Compute $Seci.P =$ $pub^{-1}.f(r, k.pub.P)^{-1}B_1$ Checks validation of $Seci$ Computes and sends $B_2 =$ $pub.f'(r, k.pub.P).K$
Checks if $k^{-1}s^{-1}B_2 =$ $Pub.P$	←	

Figure 4. 17: Songhela and Das (2014)

Chou (2014) proposed an authentication protocol by using elliptic curve cryptography and hash function. In contrst with Songhela and Das protocol, Chou protocol store shared secret tag identifiers with the server and does not need to store private or public key for tags. Chou et al. protocol is aimed to provide forward secrecy and mutual authentication and resistance to RFID security attacks, however, Zhanq and Qi (2014), and Ryu et al. (2015) showed that this scheme cannot provide forward privacy and vulnerable impersonation attack. Chou et al. protocol is shown in figure 4.17

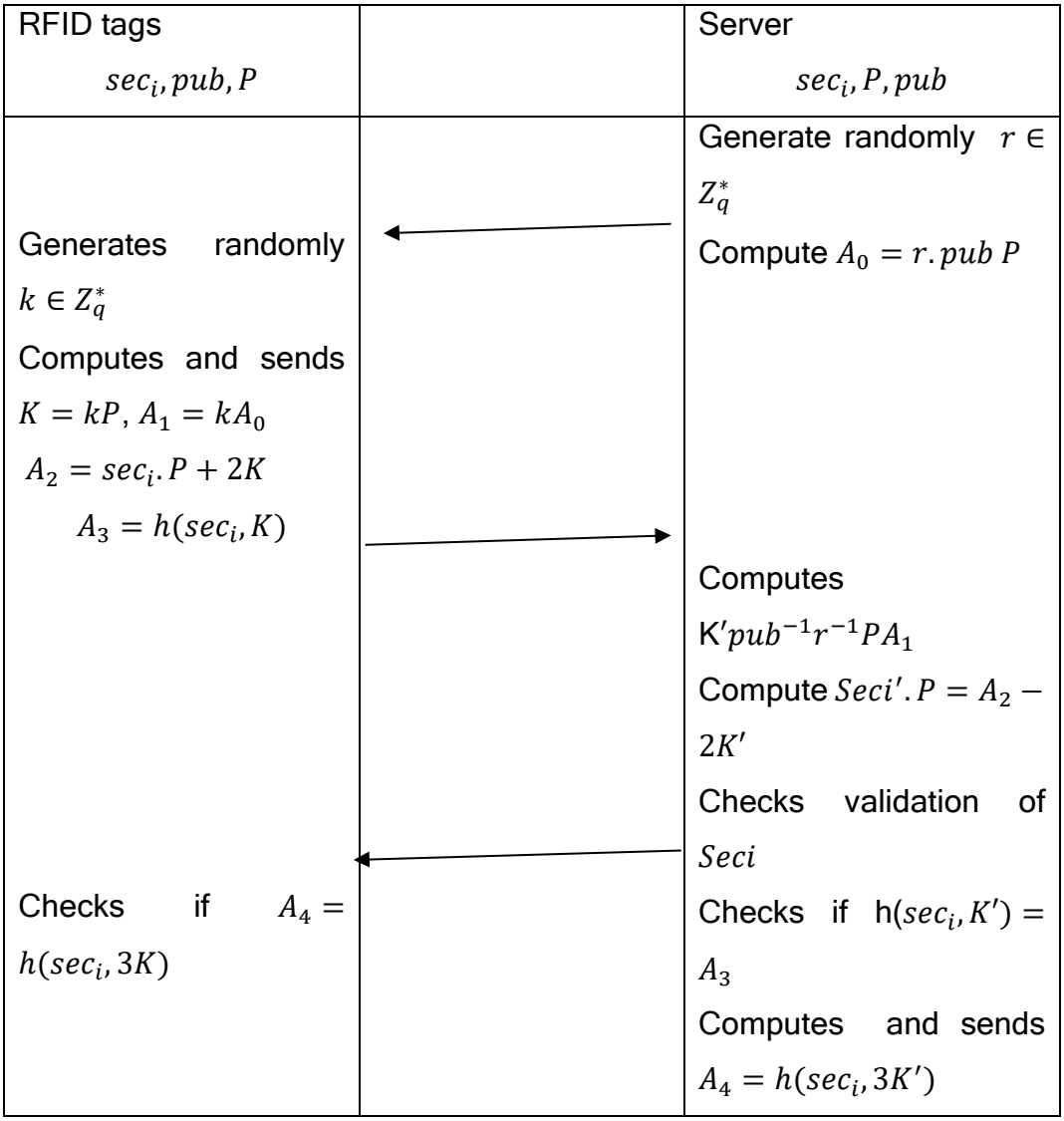

Figure 4. 18 Chou et al. authentication protocol (2014)

4.4 Multi-Tag RFID Authentication protocol

The term of enabling multi-RFID tags to be presented within an RFID reader and to be scanned simultaneously is referred to be called a grouping proof. The grouping proof protocol aimes to enable a pair of multi-tags to generate a proof which shows that they have been scanned at once by a reader. An RFID reader can read several tags by recognising all tags after identification procedure. The result of the yoking protocol is to produce a proof of verification to offline parties

instead of direct involvement. Grouping proof was first introduced by Jules in 2004 as a yoking proof protocol (Jules, 2004). The yoking proof starts with two tags Tag_A and Tag_B share the secret keys sec_A and sec_B and a back-end server. The protocol starts when a reader sends a command to Tag_A , then the tag start to generate and send a random number r_A to the reader. Upon receiving tag identify and r_A, the reader sends r_A to the Tag_B. Tag_B calculates the MAC of r_a with the key sec_B , generate a random number r_B and then sends to the reader $MAC_{sec_B}[r_A]$ with r_B with tag's identity. The reader sends r_B to the Tag_A to compute the challenge M_a. The tag calculates and sends $M_a = MAC_{X_A}[r_B]$. For the proof verification, the reader sends to the back-end server $Proof_{AB}(A, B, M_A, M_b)$ with r_A and r_B. The back-end server generates a proof $Proof'_{AB}(A, B, M_A, M_b)$ and compares it with the previous proof. If these values are equal then both tags T_A and T_B are scanned simultaneously.

The Yoking-proof protocol produces only a basic computation and does not include cryptographic functions only including MAC function.

Saito and Sakurai (2005) pointed out weaknesses in the work of Jules and showed that it is vulnerable to replay attacks. So they proposed a new solution using timestamps. They also generalized the concept for a group of tags and introduced the corresponding grouping proof. Piramuthu (2006) proved that Saito's protocol is also vulnerable to replay attacks so Piramuthu (2006) proposed to include random values instead of timestamps to thwart replay attacks. However, in 2007 Peris-Lopez et al. (2007) and Lin et al. (2007) proved Piramuthu's protocol using random numbers is subject to multi-proof session replay attacks and proposed a new grouping-proof protocol.

Chien and Liu (2009) proposed an anonymous tree based yoking proof to reduce the computational cost of identifying a tag in the verifier from O (n) to O (1). Huang and Ku (2009) proposed an online protocol for enhancing medication safety of inpatient which supported low-cost RFID tags. Unlike previous proposals, it replaced message authentication code (MAC) and hash function with 16-bit pseudorandom number generation (PRNG), bitwise operations and cyclic redundancy check (CRC) function.

Burmester (2011) proposed three grouping-proof protocols based on the idea of sharing of group identifier ID, and the third supports anonymity and forward-security properties.

Peris-Lopez (2011) did security analysis about protocols described above, and the result showed that: the scheme of Burmester (2011) is vulnerable to multiple impersonation attacks and the schemes of Huang and Ku (2009) are vulnerable to privacy attacks and forgery attacks. Shen et al. (2014) proposed a grouping proof protocol that provides mutual authentication and can support a group of tags with multiple readers. In their protocol, the communication is run through using fixed values of the tag' identifier with the tag's group identifier and reader identifier which lead to information leakage as these values are plain text valued (Shi et al., 2017).

A key distribution method for RFID grouping proof was introduced by Huang and Mu (2015). In their challenge-response protocol, it functions by interacting tag and reader three times during the process. In the first process, the reader sends a secret value with a random number then tag responds and updates it secret and send the previous secret to the reader. The reader checks the legality of the tag and response by generating random values that are related

to the secret key and send it back to the tag. After receiving from a reader, tag updates its secret again and send it to the reader as a proof of authentication.

Shen et al. (2016) proposed a grouping authentication protocol that uses bitwise operation and without any encryption method or hash function. In their protocol, a series of signature methods are used to generate the grouping proof through three authentication process. In the first process, a reader obtains a number of tags identity and associate them with random numbers. In the second process, linking the whole tag group is produced by proofing a set of tags. The final process is the verification process that verifies the final verification to a database. As the protocol does not involve any encryption method, their protocol can be vulnerable to eavesdrop the message in the first process and assume the group key and the tag sequence of numbers.

The grouping-proof protocols which have been discussed above use only simple bit-wise operations, like XOR, AND, OR and rotation, and non-public key encryption techniques such as: hash functions, MACs, pseudo-key functions, due to their simplicity compared to public-key algorithms. However, security and privacy alongside with scalability are issues with these schemes.

Public key algorithms provide strong security and privacy for RFID system as well as scalability when applying for a large amount of RFID tags. The only problem was the hardware ability of low-cost RFID tags. Since the introduction of ECC with RFID tags, many researchers proved that ECC can be suitable for implementation in low-cost RFID

Batina et al. (2011) proposed an ECC protocol that can allow a pair of tags to be authenticated at once. As shown in figure 4.19, a reader starting from left to send a message to the first tag then start right to point out. The first tag generates a random number with the corresponding to the EC point then sends

this message to the second tag. The second tag also generates a random number and compute a corresponding message and compute its response by using the private key. Both correspondings from each tag are exchanged to the reader then the reader will send the corresponding message of the second tag to the first tag to compute the challenge phase by using its private key. All of the challenge response will be forwarded to the reader for the final authentication. Lv et al. (2011) and Hermans and Peeters (2012) showed that it Batina et al. (2011) is unsecured to the tracking attack, man-in-the-middle-attack (MITM) and impersonation attack.

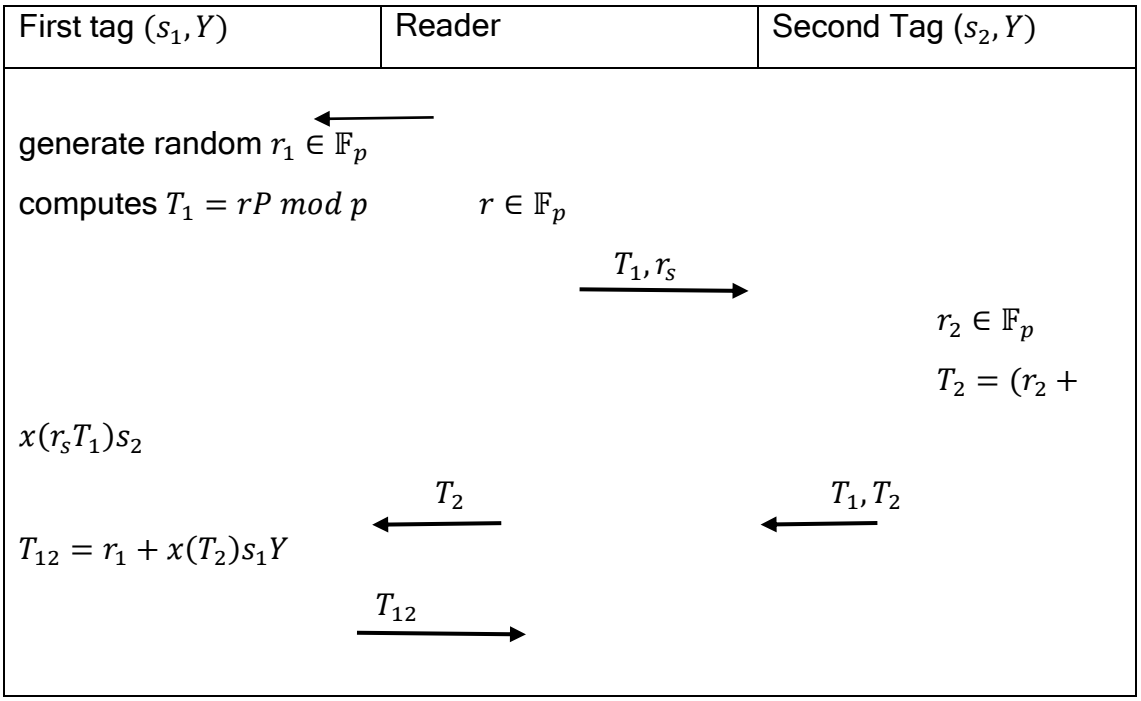

Figure 4. 19: Batina et al.(2011)

Lin et al. (2012) proposed an improvement for Batena *et al.* protocol. However, Ko et al. (2014) found that Lin et al. protocol is vulnerable to impersonation attack as well as to tracking attack. Hong- Yang (2015) proposed a grouping-proof protocol to overcome the low generation efficiency of the grouping-proof protocol by conducting a parallel processing. Later, Cheng et al. (2017)

proposed a protocol that is based on ECC and should be secured against MITM attack and provide strong privacy. However, these protocols have been not tested in terms of tag impersonation attack.

4.5 Secret Sharing Techniques

This idea was first introduced by Langheinrich and Marti (2007) The idea of their work is mainly based on using Shamir secret sharing scheme to split the tag's ID into the secret. Although their protocol is secured against eavesdropping attack, Lv et al. (2011) found that there is a scalability issue within large numbers of tags.

Another attempt of using Shamir secret Sharing scheme was in Langheinrich and Marti (2007) protocol; they proposed a distribution process to share the tag's secret ID between reader and tags as a set of encoded shares and stored in the RFID tag memory. In their authentication process, a combination of secret shares is required to verify that the secret key is correct.

Later a key distribution protocol has been proposed by using the Ramp secret sharing scheme where the size of each share is smaller than the size of the secret (Jules et al., 2008). However, Cai et al. (2009) found that Jules et al.(2008) suffers from tracking and counterfeiting attacks.

In order to reduce the computation cost of Shamir base scheme, Lv et al. (2011) proposed two secret sharing schemes based on XOR and an addition operator. However, Sato et al., (2016) showed that a shared secret can be learned by eavesdropper adversary.

Cai et al. (2009) proposed a scheme that can be secured against tracking by using a hash function to update the tag secret key. However, their scheme still vulnerable tracking attack (Abughazalah et al., 2014)

In order to ensure privacy and prevent tracking attack, Abughazalah et al. (2014) proposed a secret key update scheme that is based on dividing and distributing shares by addition operator, then using a hash function to ensure privacy. Later, Sato et al., (2016) proposed an idea for a secret sharing scheme by suing Shamir secret sharing scheme and used dummy RFID tags and real RFID tags. However, Sato et al. (2016) assumed that when using a few amount of products, the number of dummy tags are increased. For this reason and for the real implementation of RFID applications, the dummy tags need to be reduced (Sato et al., 2016).

4.6 Summary

Cryptographic RFID authentication protocols have been briefly reviewed in this chapter. Authentication protocols which are based on using non-public key encryption and public key encryption and secret sharing schemes are also reviewed.

In general, Authentication protocol based on non-public key encryption is faster and less complex than public key encryption protocols. However, non-public key encryption protocols are more vulnerable than public key encryption. This is because of the sharing key between the participant. Therefore applying public key encryption for a large distributor is more suitable than non-public key encryption. The feasibility of public key encryption protocol is till now an open problem due to computation complexity and high cost.

5. Zero-Knowledge Authentication protocol and RFID Tags

5.1 Introduction

The design of an RFID authentication protocol is considered to be complicated due to the ability of building a secure protocol that can fit the lack of computational capability of RFID hardware. As described in the previous chapters, there are some major security and privacy problems that occur during the wireless communication between the RFID tag and the RFID reader. These security problems are varied and depend on the ability of an attacker to eavesdrop and interrupt the communication session, modify the broadcast exchange message and even can block the channel between the RFID tag and the RFID reader. Therefore there is a need to build up a secured system that can prevent these types of attacks. In addition to the security concerns, the privacy of tag is also an issue due to the ability of an attacker to track the RFID tags during the communication. An RFID system needs to be provided with un-traceability property for the RFID system by disabling an attacker to recognise the interaction with RFID tags and derive the tag's information. This information can be used by an attacker to trace the tag.

Therefore, this chapter will discuss the weaknesses of an existing protocol which was introduced by Tulys and Batina (2006). They proposed an RFID authentication protocol that used ECC version of Scnorr identification protocol that can be used for RFID systems. The weaknesses of their protocol are highlighted in terms of Tracking attack problem and Man in The middle attack

problem. Although another modifications of the original Tulys and batina (2006) protocol have been introduced, most of these protocols are still vulnerable to tracking attack as mentioned previously in chapter four. Some other researchers have introduced other protocols which are based on using Zero Knolwege proof with the ECC. However, as mentioned in the previous chapter some of theses protocol do not satisfy the security and privacy requirement for an RFID system. Therefore, in this chapter, we try to beat the vulnerability of tracking attack of an existing RFID protocol proposed by Tuyls and batina 2006. The modification of the existing protocol can overcome the problem of tracking attack by enhancing the ability of the protocol is to prevent such attack by ensuring the integrity of the message exchange between the RFID tag and the RFID reader.

5.2 Backgrounds

In this section, we will introduce some of the basic techniques that will be used in the improvement of the modification protocol.

5.2.1 Definitions

The Discrete Logarithm Problem (DLP)

"Given the finite cyclic group Z_p^* of order $p-1$ and a primitive element $\propto \in \mathbb{Z}_p^*$. The DLP is the problem of determining the integer $1 \leq x \leq p-1$ such that $\propto^x \equiv \beta \ (mod\ p)$ "

Coprime number

Two integer numbers a and b are said to be **relatively prime** or **coprime** if the only positive integer that evenly divided both of them is 1. i.e the greater command divisor gcd(a,b)=1.

Primitive root modulo n

A number g is a primitive root modulo n if every number coprime to n is congruent to a power of g modulo n such that $g^k \equiv a \ (mod\ n)$. In other words, g is the multiplicative group of integers modulo n.

5.2.2 a Schnorr Identification Protocol

In 1989, C.P Schnorr (Shnorr, 1989) introduced Schnorr identification scheme as an improvement of ElGamal signature scheme which was introduced in 1985 (ElGamel, 1985). At that time, Schnorr proposed a signature and authentication protocol that is based on the discrete logarithm problem and satisfying zero knowledge proof. The Schnorr identification scheme is based on the idea of ElGamal and Fiat and Shamir Schemes. However, Schnorr was enhanced further in a way that can improved the speed of ElGamal signature scheme. The advantage of Schnorr scheme comes from the intractable computation of a discrete logarithm over a finite field.

The Schnorr scheme needs a trusted authority to choose a domain parameter with the properties:

1- P is a large prime number such that $p-1$ is divisible by another large prime number q.
2- $\propto$ is an element in $\mathbb{Z}_p^*$ with order q.
3- t is a security parameter such that $q > 2^t$.

The user chooses a private key a such that $0 \leq a \leq q-1$. Then computes the public key $v \equiv \propto^{-a} \ (mod\ p)$ or $v \equiv \propto^{q-a} \ (mod\ p)$.

As shown in figure 5.1, the protocol starts after the claimant chooses a random value $r \in \{1, \dots, q-1\}$, then computes $x = \propto^r \ (mod\ p)$ and sends it to the

verifier. The verifier sends random value $e \in \{0, \ldots, 2^t - 1\}$ to the claimant. The claimant computes and sends $y = a * e + r \ (mod\ p)$. The verification process is done after the verifier computes $z = \beta^y * v^e \ (mod\ p)$. It will be accepted if $z = x$ and rejects if $z \neq x$.

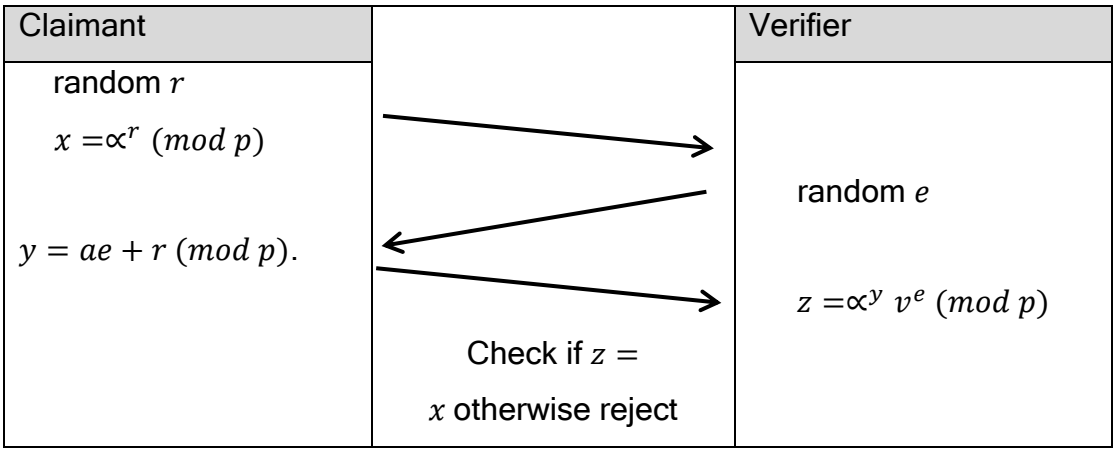

Figure 5. 1: The Schnorr Identification Scheme

The last step of the scheme is the conviction of the claimant's identity to the verifier, therefore the calculation of $\propto^y v^e \equiv \propto^{r+ae} v^e \ (mod\ p)$

$$\equiv \propto^{ae} (\propto^{-a})^r \equiv \propto^{r+ar-ar} \equiv \propto^r \equiv z \ (mod\ p)$$

The claimant uses the private key a along with the protocol without revealing its value. The claimant convinces the verifier about the private key information and the verifier should compute the calculation without knowing the value of the private key. This scheme was introduced to enhance the speed of the computation and was designed to fit smart cards that have a limitation in computation. The desirable property of the Schnorr protocol is that it can used in applications that have a low computation capability with a limit power such as smart card and RFID tags.

Schnorr identification scheme was designed to be very fast and secure. However, there is a possible probability to cheat and break this scheme. Taking

into consideration, if an attacker guesses the correct value of the verifier challenge e then the attacker can calculate the value of $\alpha^y v^r \ mod\ p$ and sends the value to the verifier that will send the challenge e. Then the attacker can send the value of y to the verifier and then the verifier accepts. The probability of guessing the verifier challenge is equal to 2^{-t} because $e \in \{1,2, \ldots . 2^t\}$ unless the verifier sends the same challenge in each round (Schnorr, 1989). Another possibility to break this scheme, if an attacker tries to compute the value of the private key a which is $a = -\log_{\propto} v$. However, the computation is infeasible (Schnorr, 1989) because it will take a lot of time and memory until finding the value a.

Another possibility to break the protocol is suggested by (Stinson, 2006). He supposed that an attacker can compute $r_1\ r_2$ with $y_1\ y_2$ from the value of x.

Then $x \equiv \propto^{y_1}\ v^{r_1}\ mod\ p \equiv x \equiv \propto^{y_2}\ v^{r_2}\ mod\ p$

$\propto^{y_1 - y_2} \equiv v^{r_2 - r_1}\ mod\ p$

Since $v = \propto^{-a}\ mod\ p$ then $\propto^{y_1 - y_2} \equiv \propto^{-a(r_1 - r_2)}\ mod\ p$

Since $\propto$ has order q so $y_1 - y_2 \equiv a(r_1 - r_{2)}$.

The attacker aim is to find the private key a. Since $r_1\ r_2 \in \{1,2, \ldots, 2^t\}$, then $1 \leq r_1 \leq 2^t, 1 \leq r_1 \leq 2^t$ and $0 < /r_2 - r_1 / \leq 2^t$. Hence $\gcd(r_2 - r_1, q) = 1$ and $(r_1 - r_2)^{-1} mod\ q$ exist, then

$a = (y_1 - y_2)(r_2 - r_1)^{-1} mod\ q$.

Now, if the attacker knows the private key a then he can impersonate the claimant's private key with probability P = 1. This implies that at schnorr protocol is soundness if the private key not impersonate. The Schnorr protocol is completeness and soundness that leads to proof of knowledge.

The amounts of the message exchanged requirements during the authentication process are acceptable because of the multiplication property in the finite field. Using discrete logarithm problem in the finite field makes computation efficient and intractable. In the first step, 1024 bits of information are sent from the claimant's computation including the amount of certification. This information is the most important information that needs to be transmitted. In the second step, 40 bits of information are for generating verifier random number. 160 bits of information for the claimant calculation of $y = ae + r\ (mod\ p)$. The last process of the protocol is for the verification and does need transition (Stinson, 2006). The efficiency speed of the protocol is because the claimant does not do too much computation. Most of computation is done by the verifier. Taking in consideration that the value of $p, q, t\ and \propto$ are public and used by every user in the network range.

5.2.3 Random oracle model and Keccak Hash function

Random oracle model is used in security proofs as an ideal comprehensive hash function. The Keccak hash function now a standard SHA-3 is a close practical realisation of the random oracle model. Keccak is a cryptographic hash function that wins the competition SHA-3 by National Institute of Standards and Technology (NIST). It is a hash function that based on a sponge construction as a building block of permutations. The sponge construction is used to build a function that maps an input of variable arbitrary lengths then outputs arbitrary permutation lengths by using a repetitive construction. The width of the function f is a fixed number of bit b which operates on a state $b = r + c$ where r is the bit rate and c is the capacity. Initially, the root bit of the state are zero then the

input message M is interleaved to the function by padding the r-bits and dividing into blocks then XOR with the first r- bit such as $M||pad[r](|M|)$, where $|M|$ is the length of the message M. This process is called absorbing phase. After that, the squeezing phase starts with returning the output blocks from the first r-bit and interleave with the function f of the fixed length permutation.

The sponge construction has advantages to be used in cryptographic applications due to its permutation process and the output of arbitrary variable length also it can be used as a stream cipher. The security of sponge construction is varied, since it can determine the pseudorandom bit generator with the output that leads to providing security against generic attack (Bertoni et al., 2009).

The Keccak hash function *Keccak-f[b]* is depending on the choosing of width permutation *b* where *b* is the [25,50,100,200,400,800,1600]. To obtain the *Keccak-f*[*b*] sponge function, it needs to apply the sponge construction to *Keccak-f*[*r*+*c*] with the parameters capacity *c*, and bit rate *r*. The state is a three dimensional array of *5*5*w* with the property that each length of *w* bits are [1, 2, 8, 16, 32, 64] such that b=25*w*. Depending on the permutation width rounds, there are 7 types of Keccak hash function. The *Keccak-f [25]* has 12 number of rounds, where the number of rounds can be calculated as $12 + 2 \times l$ such that $2^l = w$. As a result, the other Keccak functions such as *Keccak-f[50]* has 14 number of rounds*, Keccak-f[100] h*as 16 number of rounds*, Keccak-f[200]*has 18 number of rounds, *Keccak-f[400]* has 20 number of rounds, *Keccak-f[800]* has 22 number of rounds and finally *Keccak-f[1600]* has 24 number of rounds (Bertoni et al., 2009).

5.3 Review of RFID Authentication Protocol based on Schnorr Identification protocol

Tuyles and Batina (2006) introduced an authentication protocol that is based on challenge response mechanism by using Schnorr identification protocol. In their model, the Schnorr identification protocol is introduced as an elliptical curve version of the original Schnorr identification protocol. Their protocol is aimed to design an RFID authentication protocol that can resist to passive attacks. Their protocol is based on the challenge-response mechanism that is used a zero knowledge procedure to generate a verification for the verifier.

Their protocol has two phases, setup phase and the challenge phase between the prover and the verifier. In the first phase, a server is responsible for generating the elliptical curve and finding the base point which it will be used for the further processes. After generating the base point of the elliptical curve, the server will generate the public key, the private key, the tag's identifier and other system parameters that need to be used for the challenge phase. The challenge phase is used for the generating the proof of verifier. As shown in figure 5.2, the challenge phase starts after the hello message of the reader, and then the RFID tag response to the hello message by generating and sending a random $r \in \mathbb{F}_p$ and generating and sending $R = [r[P$ where P is a base point is in the elliptical curve proof. The verifier responds to the received message is by generating and sending a random $e \in \mathbb{F}_p$. The prover starts to challenge the verifier by computing $y = ae + r \; mod \; p$, then sends the challenge to the prover. The verifier accepts the challenge by computing $[y]P + eX \; mod \; p$. If the

value of the $[y]P + eX = R$ then the challenge is accepted otherwise the session is rejected.

The zero knowledge proof in the elliptical curve version of Schnorr identification scheme provides the perfectly zero-knowledge property that means providing completeness, soundness and honest-verifier zero-knowledge. The only problem with this protocol is when applying within RFID system there is a danger in providing security and privacy when illegitimate reader involved with the protocol. Illegitimate reader can choose many challenges to the tag which leads to losing the zero-knowledge property. However, the elliptical curve version of Schnorr identification protocol is secure against passive attacks but leak some security and privacy property with active attacks.

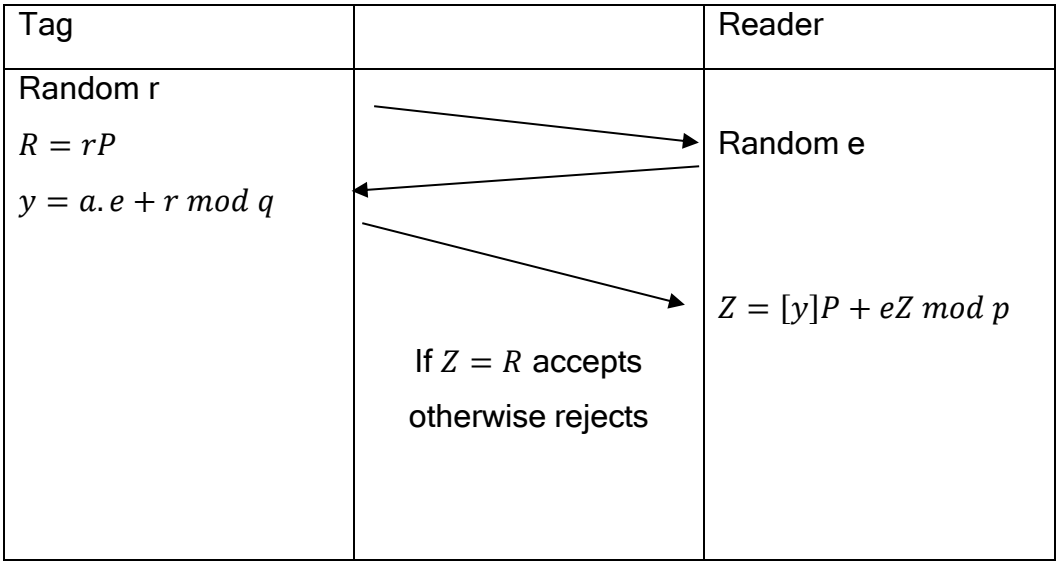

Figure 5. 2: Tuyls and Batina (2006) Protocol

5.3.1 Tracking Attack

An important property for a tag is to provide un-traceability property so that an adversary cannot able to recognise any change in the tag information. In other words, an adversary can intercept the channel between the RFID tag and the RFID reader and derive tag's information from the tag's response to the reader. As the RFID tag response to the reader by a challenge response, it is necessary to deny illegitimate reader to derive the tag's information. In the challenge-response between the RFID tag and the RFID reader, the tag usually responds with a function of the secret and the challenge and keeps the created nonce. If the attacker involves his control with the challenge information, then he can evaluate the value of the function of the secret and the challenge after the round challenge. This leads to the tracing problem with the RFID tags.

To ensure untraceability property in challenge-response protocols, an RFID tag needs to ensure the validity of its nonce and refresh the nonce after each successful session. In other words, if the tag receives two challenges with the same value of challenge, then the tag needs to respond with two different values of the nonce. An authentication protocol is said to be secured against tracking attack, if there is no polynomial time oracle that can control the protocol. For instant, let ch be the challenge, r is the tag nonce then $Track\big(ch_1, g(c, r_1) \ldots\ldots, ch_n, g(ch_n, r_n)\big) = Track'(ch_1, \ldots. ch_n)$

Where $Track'(ch_1, \ldots. ch_n)$ is the tracking polynomial time function that output the tag's information by an adversary. Hence, by recording the challenges, the adversary track the tag's response then compute the function $Track'_{secr}(ch_1, \ldots. ch_n)$. An adversary also can determine if there are

two different tags or one by repeating the tracking function and check if the outputs are equal for one tag there are two different tags.

Lee et al., (2008), showed up the Tuyles and Batina (2006) protocol is suffered from attacking attack by eliciting tag's information $-aP$ by computing:

$$Track(rP, e, ae + r, P) = Track'(rP - (xe + r)P.e^{-1}$$

Thus, there is a polynomial time oracle that can output the value– aP and there is no random notion from the tag reminded by the equation.

5.3.2 Man-in-the-Middle Attack

A man-in-the-middle-attack is a type of eavesdropping attack that can be used to intercept the message exchange between the RFID tag and the RFID reader. An attacker should have the ability to monitor the communication channel to launch this attack. The ability of the attacker can be just monitored the channel or can be injected some message in the message exchange between the RFID tag and the RFID reader and act as an intruder who relays the communication. The man-in-the-middle attack affects many cryptographic protocols including elliptical curve cryptography.

Tuyles and Batina (2006) protocol is also vulnerable to man-in-the middle attack. The data in the exchange message are related to the tag by generating the nonce e. The tag sends the message $y = se + r \; mod \; p$ to the verifier. In the meantime, the adversary can tamper the message and send his own message $y' = y + c \; mod \; p$. Upon receiving the message from the adversary, the verifier thought that the message from the tag then the final computation complete the session by checking if the value of $y'P + eZ = R'$ otherwise reject. The final value usually leads to R' since $y'P + eZ = (y + c)P - esP = (se + r + c)P -$

$esP = (r + c)P = T'$. Thus lead to having the tag information by an adversary. An adversary can intruder between the RFID tag and the RFID reader then tamper the original message while the tag and the reader complete the session. Upon receiving the first message from the tag which includes $R = rp$ the adversary can modify the message and send his own message $R' = rP + cP$. The reader response normally.

5.4 The Zero- Knowledge authentication Protocol

In this section, we will introduce an enhancement method that can be applied to the elliptical curve version of Schnorr identification protocol. These enhancements can determine the ability of an intruder adversary by introducing a secure mutual authentication between the RFID tag and the RFID reader. Ensuring the privacy and security properties of the modification protocol will be also introduced.

5.4.1 Design Aim Statement

The consideration in enhancing or designing a new protocol is depending on the status of the adversary ability assumptions. The aim of the proposed design is to achieve the set of security and privacy requirements such as providing confidentiality of the information, integrity and availability against threats. In addition, provide secure mutual authentication between the RFID tag and the RFID reader. Moreover, the proposed RFID authentication protocol will take into consideration the possibility of an adversary to threaten the protocol. The statement of defining the goals of an authentication protocol is needed to

simulate the adversary ability and furthermore to discuss the vulnerability according to the possible threat.

The confidentiality of the proposed protocol will be achieved by using elliptical curve version of Schnorr identification protocol which is secure against passive attacks and the enhancement of the system will achieve immunity against tracking attack. The integrity of the data will be achieved by using Keccak hash function so an adversary cannot able to modify, insert or delete the message exchange between the RFID tag and the RFID reader. Ensuring availability of the system is ensured by refresh the random nonce of the proposed protocol so using different random nonce in each authentication session will keep the property of the availability of the proposed protocol. The proposed system will consider the following adversary ability:

- During the authentication, an adversary can eavesdrop and change the message exchange between the tag and the reader. At the end of the authentication protocol, the adversary win if the reader returns a successful response.
- An adversary select random tag to inject his own message and complete the authentication
- An adversary can intercept the message exchange and monitor the communication or send his own message to get the tag's information
- An adversary blocks the message exchange in order to prevent further message exchanges.
- An adversary can trace RFID tags
- An adversary can impersonate tag and act as legitimate tag.
- An adversary can impersonate the reader and act as a legitimate reader.

5.4.2 System Overview

The proposed system considers uses the elliptical curve version of Schnorr identification protocol. It also supposed that RFID tags have the ability to calculate a Keccak hash function with an elliptical curve. The consideration of the protocol is to suppose that the communication between RFID tag and RFID reader is insecure as the communication is run through RF channel while it supposes that the communication between the RFID reader and the back-end server is secured. The adversary ability in the proposed protocol consideration it can be a passive adversary that try to eavesdrop the communication channel or active adversary that try to reply the intercept message or adding, deleting or modifying the exchange message. As a challenge-response protocol, the proposed protocol has two phases, set up phase and authentication phase. The set up phase is responsible for generating system parameter such as generating the elliptical curve, the based point of the elliptic curve, private and public keys. The public key in the proposed scheme is public for the reader and for the tags. The authentication phase is a challenge-response procedure to get the proof of identity by sending a challenge then encrypting the message and then sending to the verifier for the verifying process. The final stage in the authentication process is to check if the information is valid otherwise the process will be rejected. To be specific, in the authentication phase, there are two verifying processes. The first process is to check if the reader is a legitimate reader that can compute the hash value from the first challenge otherwise the process is rejected. The second process is to compute and verify the hash values of the encryption message and then compute the verifying challenge.
The following notations in table 5.1 will be used for the proposed description:

Notations	Description
E	an additive cyclic group of prime with order p .
P	a base point of G.
x	the tag private key
Z	the cryptographic tag of x
$Sha3$	The Keccak hash function standard
r	a random number
e	a random number

Table 5. 1: The zero- knowledge authentication protocol notations

Setup Phase

The set up phase involves with the following steps

1- The system generates an elliptic curve group $E$of prime order p.

2- The system will choose the base point that satisfies the elliptic curve E, then find a based point on the curve E .

3- The system will generate a 128 bits of information as a private key x which assigned to tags then compute the public key Z such as $Z = -Sha3(x)$ which is also 128 bits of information.

4- The system will initialise each tag with the private key x and the public key Z while the system public key Z will be available in a data base for the verifier process.

Authentication Phase

The authentication phase is going through the following processes and summarised in figure 5.3

1- The tag generates a random notion r

2- Computes $R = [r]P \ mod \ p$

3- Computes $S = Sha3(R)$.

4- Sends S, R to the verifier to check the first consistency of the authentication procedure.

5- Upon receiving R and S, the verifier checks the value of S if the value is correct then , chooses a random notion e and sends it back to the prover. Otherwise the authentication will be failed.

6- Upon receiving the random notion e from the verifier, the prover will computes $v = Sha3(S).e$ then sends v to the verifier.

7- Upon receiving the message v, the verifier will check the value of v if the value is correct then accept the next step then sends it back to the prover as prove of calculation.

8- Next step, the prover will compute a new secret $x_1 = [x.v]\ P\ mod\ p$ then hash the value of the new secret as $x_2 = Sha3(x_1)$.

9- The prover also computes $y = x_2 e + r\ mod\ p$ and finally sends the value of y to the verifier with its hashed value for verification.

10- Upon receiving the message y, the verifier will first compute $x_1 = x.v$ the check the value of x_2 then compute the challenge as $x_2.r + r - Z_1 P = R$ if the value is correct then accept otherwise the process will be rejected.

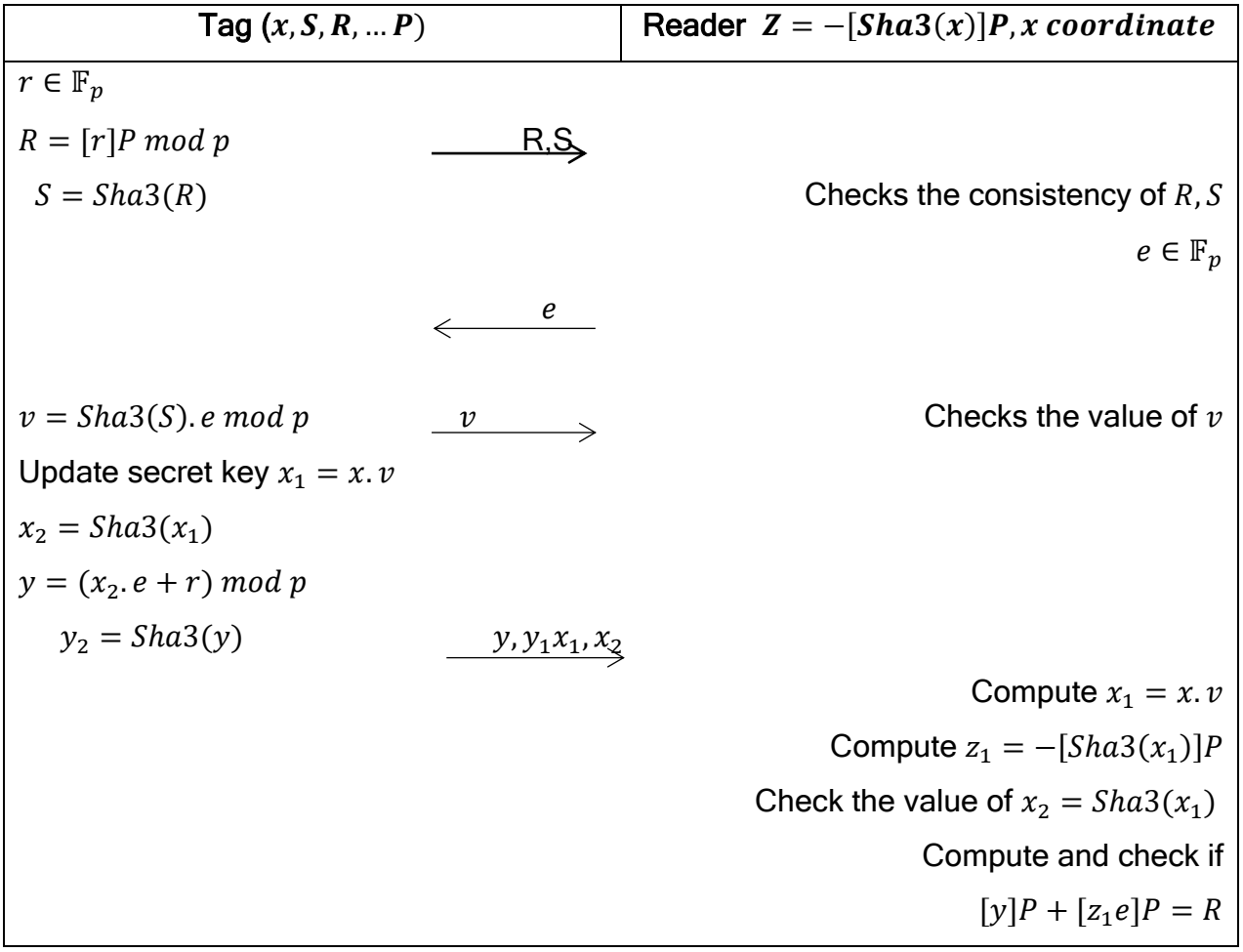

Figure 5. 3: Zero-Knowledge authentication protocol for RFID tags

5.4.3 Security analysis

In order to analyse the security of the proposed protocol, we will assume there are security and privacy threats that can affect the protocol. The following privacy and security threats will be considered:

- Correctness

The Correctness property gives the confirmation to the data base oracle to refuse illegitimate or corrupted tags from accessing to the RFID system. The Zero-Knwoledge authentication protocols runs three mutual authentication

process between the RFID tags and the reader. These process give different values in each process. Therefore, its hard to an adversary to insert corrupted tags to the system and complete the verification process. To penetrate the system, An adversary will first try to check the status of the RFID system by interacting the system with a legitimate tag's information. Subsequently, the adversary will run interactive protocol to check the ability of the system to accept illegitimate tags. In this case, the system directly will detect illegitimate tags as there are three mutual authentication processes that uses different hashed values. This fact will lead to confirm that the zero-knowledge authentication protocol has the correctness property.

- Soundness

The Zero-Knwoledge authentication protocol has the soundness property as its nigligiable to an adversary to penetrate the tag's information. An adversary can access to RFID system in order to access to the output of the protocol or to corrupt tags. In this case the adversary involves with the execution protocol between the reader and the RFID tags to impersonate some iligitimate tag's information. The adversary takes the advantage of the soundness property if his challenge return a correct bit of information from the output protocol execution. As the Zero knowledge authentication protocol based on using mutual authentication between the RFID reader and tags, the adversary needs to impersonate the reader information to have a correct bit of information. Thus, its hard to an adversary to complete the all verification process in the system until know both the authentication process and all private information. Therefore the Zero-knwoledge authentication protocol satisfy the soundness property.

- **Resists to Man-In-the-Middle-Attack**

The proposed protocol satisfies the property of resistance against the man-in-the-middle-attack. Let us suppose that an adversary can interrupt the communication between the prover and the verifier and establish his own communication. The adversary tries to monitor the communication then insert his own message and act as a legitimate party. In this case, the adversary will first intercept the message T which should be received by the verifier. Upon receiving the message T, the adversary tries to compute the hash value of T. Therefore; there is no chance to the adversary to compute the hashed value without knowing the original value of the random nonce r. Suppose that the adversary intercepts the second challenge (v, y, y', x, x'') previously exchanged by the prover and then re-exchange with his own message. However, this phase cannot be completed since the encrypted message y and its hashed value are combined with the new value of the private key and the hashed value of the private key. If the adversary succeeds to insert his own message $y1 = y + c$ and complete the verification process. In this case, the verification process will be not valid since the adversary cannot complete the hash procedure of his own message, therefore, the proposed protocol is immune against the replay attack.

- **Supports mutual authentication**

The proposed protocol provides the concepts of mutual authentication between the verifier and the prover by ensuring the authenticity of the parties. The authentication process first takes the proof of the reader by calculating the hash value of the first entity otherwise the tag turn to be silent. This shows that it's important to prove the authenticity of parties before exchanging data. In fact,

the first mutual authentication to verify the first message and to start to send the exchanged message. In the earlier phase, the authentication is generated by the tag for the reader to verify the first value is correct. Then the second phase is to check if the random response from the reader is legitimate, therefore, the tag verifies the reader message with a challenge. After that, the exchanged message is also sent with a challenge to the reader in order to complete the verification process and check the tag's information. Thus, the only genuine party can take part in the authentication process.

Supports anonymity

The protocol provides anonymity property, in facts, the random notion is generated randomly in each authentication session and has different values since they are generated through pseudo random number generator. Moreover, the protocol generates one way hash function that the challenge phase has different values in each authentication session and the tag secret is exchanged through a hash function. Therefore, the only legitimate party can respond to verification and update the status of the tag secret to complete the challenge response and verify the final message. Furthermore, as the protocol works on zero knowledge proof with elliptical curve operations, extract the tag's secret from the exchange message is a difficult process for an illegitimate party.

- Resistance to tracking attack

The protocol is resistant to tracking attack since an adversary has no control over the value of the first message R. It's obvious that the protocol hashed the value of R then the protocol transcripts is impractical for an adversary since the success of predicting the message exchange is negligible. Moreover, if a tag

executes two messages, an adversary cannot determine if these executions are executed by the same tag or not. For instant, if an adversary receive the challenge messages R, S, v, x_2, y_1 andR', S', v', x_2', y_1' of the protocol and try to determine if these values are generated from different tags or one tag. Hence, there is no normal distinguished response from tag since these values are different and based on using random nonce r from tag and e from the reader also these values are hashed by Keccak hash function. Moreover, even if an adversary act as a reader and send his own random message e_2 and send it twice to get different responses from the tag. In this case, the tag will generate the message v that is based on the value of R and its hashed value S. Thus the result cannot be a fixed value for determining a tag. Therefore, the adversary cannot be tracked the system.

- Resistant to replay attack

A replay attack happens when an adversary replaies to the previous message exchanged between the RFID tag and the RFID reader. If the adversary successfully eavesdrops the messages exchange R, S, v, x_2, y_1 previously exchanged by a tag then, the adversary now able re-exchange these message to the reader. In this case, the authentication process will not succeed as the reader need to complete the verification stage by using Keccak hash function then re-compute and update the secret to complete the final verification stage. The tag' secret is combined with Keccak hash function and the message v which is related to the random nonce e. These combinations require multi verification by a reader to complete the final verification process. Moreover, if the adversary tries to use some information from the past valid messages, then the adversary need calculate the valid response

$v = Sha3(S).e \bmod p$ then update the secret $x_1 = x.v$ and compute $x_2 = Sha3(x_1)$ which is negligible to get to the final response. As a result, the adversary cannot pass through the mutual authentication process and cannot replay the query. Therefore, the protocol is resistant to replay attack.

- **Impersonation attack**

Suppose that an adversary tries to communicate with the reader instead of a specific tag in order to complete the authentication process. In this case, the adversary needs to reply to the reader with the specific challenge. Meanwhile, the adversary needs to have a valid message to respond to the reader. In other word, the adversary needs to have the knowledge about the secret information about the tag which is combined with the Keccak hash function and generated after successful verification stages. Moreover, it is intractable for the adversary to compute $x_1 = x.v$ since the the message v is generated by the random nonce e that combine with the hash value S. Therefore, tag impersonation is infeasible for an adversary to act as a legitimate tag.

In the case, if an adversary tries to impersonate the reader and act as a legitimate reader in order to get the tag's information. The only possibility for the adversary is to generate a nonce e and wait for the response from the tag. After receiving the first challenge R and S, the reader will force the difficulty in computing the verification process as the Keccak hash function is involved with the verification process before the stage of sending the exchange message. Therefore, there is no information to be considered if the adversary successfully impersonates the reader. Thus, there is no information leakage since the tag secret is combined with the Keccak hash function.

- **Denial of service**

An adversary attempts to intercept or block the exchanged message exchange between the RFID tag and the reader. This can be done by numerously replaying the exchanged message in order to enforce the system from completing the verification session. However, as both the RFID tag and the RFID reader update their record in each authentication session, there is no further consideration for this type of attack.

- Forward traceability

Through the authentication process of the protocol, there are calculations of random nonce in each session and these are always fresh. So an adversary cannot obtain the final t message from the communication messages. Therefore, the protocol can ensure the forward traceability.

Table 5.2 presents comparison results for the protocol with related elliptical curve authentication protocol. The table shows that the protocol is the most secure protocol compared with the other schemes.

Attack / Protocol	IA	TA	RA	MitM	DoS	FT
Tuyles and batina	☒	☒	☑	☒		☒
Batina et al	☒	☒	☒	☒		☒
GPS	☑	☒	☑	☒	☑	☒
Randomized GPS	☑	☑	☑	☒	☑	
Randomised Hash GPS	☑	☑	☑			
Randomised Schnorr	☑	☑	☑	☑		☑
Martinies et al	☒	☒	☑	☒	☑	
Zhang et al	☒	☑	☑			
The zero knowledge authentication protocol	☑	☑	☑	☑	☑	☑

Table 5. 2: Comparison between related work and the zero knowledge authentication protocol

5.5 Summery

In this chapter, a mutual authentication scheme has been proposed which is based on enhance the security vulnerability of the Tulys and Batina (2006) protocol. The vulnerability is related to replay attack and lack of resistance to tracking attack. Their protocol is based on zero-knowledge proof by using elliptical curve version of Schnorr identification protocol. Based on these two attacks, a modification and enhancement of the original protocol is proposed to overcome these issues. So far, the protocol uses the Keccak hash function in the authentication phase to ensure the data integrity and to provide mutual authentication between entities. Alongside with the tracking attack and replay attack, the protocol aims to provide privacy properties such as anonymity and mutual authentication. Security analysis of the protocol is discussed with a comparison with related work.

6. Quorum RFID Based Systems

6.1 Introduction

Supply chain management is one of the important applications of RFID system as it allows products to be genuineness verified. Products are needed to be traced in every step from manufactures to distributors to confirm verification and identification of products. However, the use of RFID tags comes with some major problems that are related to distributing secure keys among multi-RFID tags and providing secure authentication protocol. In this chapter, two models of a secure based RFID system protocol are proposed for the case of distributing keys among multi RFID tags. The security and privacy properties of the proposed protocols are varied and depend on the hardness of the cryptographic techniques that are used in the design of the protocols. The design goal of the two proposed protocols is to build up a secure RFID based system that can be applied in a supply chain management scenario when key distribution techniques are required. Ensuring security and privacy for the RFID system is a major aim for the proposed protocols to overcome the security and privacy threats in RFID based system. Our protocols are addressing these security and privacy requirement alongside with the hardware limitations of the low-cost passive RFID tags. Therefore, the major cryptographic techniques are used in the server.

The main idea of the first proposed protocol is to store tags keys securely by using elliptical curve version of ElGamal cryptosystem and distribute keys among RFID tags by using Shamir secret sharing scheme. The ElGamal cryptosystem enables parties to securely identify tags while using symmetric key cryptosystem to ensure mutual authentication between parties. After each

successful authentication, the information is gathered securely in other servers and using authentication techniques are used to complete the process.

The second proposed protocol is also stored tag securely but using elliptical curve version of the light scheme of Cramer-Shoup to achieve forward and backward traceability and to provide high security and privacy against RFID threats.

The sequence of this chapter is as follows: section 6.2 will introduce the design aim and define strategies of the proposed protocol then followed by the design approach. Section 6.3 will introduce the first system approach with all system characters. Section 6.4 will introduce the second protocol approach. Section 6.4 will introduce the implementation of both systems with worked example and algorithms. Section 6.5 will discuss the performance and security analysis of the protocols. Finally, section 6.6 will conclude this chapter.

6.2 A Quorum RFID based system Design

RFID was applied to the supply chain management system to ensure work accuracy and to automate work as well as ensuring security for the system. RFID applications with supply chain management consist of series of steps that take a product to go through a series of steps. Each step is equipped with an RFID reader, and when a product moves to the subsequent step of a supply chain, an interaction takes place between the product's RFID tag and the reader associated with the steps. The verification is in the last method in each step to know whether a product in their range went through a correct sequence of steps in the supply chain or not.

Like other RFID applications, supply chain management is susceptible to security and privacy threats that can lead to information leak about products. Thus specious products can be introduced with the system. Therefore, security and privacy in system is a major part of the system to stop specious products and to prevent the vulnerability of the system.

6.2.1 System scenario

Usually, in supply chain management products are packed into boxes at manufacturers, shipped to warehouses, and then sent to retailers and distributors. As an RFID-tagged box leaves the manufacturer, it scans the information of the tag and records the tag's ID to create lists of items for the inventory purpose. The manufacturer then updates their database that lists the tags associated with the shipped items. This database tracks the tags and the tagged items. So, for example, the manufacturer may mark the state of shipped items as possible with the location of the warehouse. When the warehouse receives the package, it scans the case and the tags which are attached to items, and then a scanner compares the results of the scan with the listing of goods. The warehouse system can detect any of the goods that are lost or stolen even that tags that did not respond or failed to deliver to an appropriate place. The warehouse can determine these faults by checking the existing parcel with the listing of goods

For this scenario of distributing and shipping goods, the design model will be introduced to help with the procedure of this scenario. The idea of the design is based on a problem of scanning one parcel which contains multi-RFID tags that attached to products. Taking into consideration, the distribution keys

among multi-RFID tags and the way of authenticating multi-RFID tags in one package. Therefore, the system design will introduce a procedure for distributing and authenticating multi-RFID tags.

6.2.2 System Design

An application example application using RFID tags considers that a package in the process of supply chain management moves from manufacture to other transit steps. During the transit, the package will contain several RFID tags and critical logistic about the package is stored on the RFID tags. Since the package contains several RFID tags, there are possibilities to lose one or more RFID tags during the transit or there is a possibility to fail in transit. Therefore, the consideration for this case is to use a threshold cryptosystem system that ensures there are no effects of loss or damage in some part of the tagged package. Moreover, the threshold system considers the cases when intruder adversaries try to eavesdrop the communication channel in order to have some information related to the package. Therefore, multi servers will be used to ensure the authenticity of the package and to ensure there is no danger of possible RFID privacy and security threats.

An illustration of the system is sown in figure 6.1 which describes the main ideas of the system and the main goals for using multi-server for authenticating the tagged package.

The threshold cryptosystem uses multi-servers such as dealer server, re-construction server, decryption server and finally logistic information. The dealer server is responsible for creating encrypted tag identifier then generating a threshold polynomial with degree of $k-1$, where k is the number of tags, to distribute the encrypted identifier among multi-RFID tags. By considering, the

encrypted identifier will not use on the RFID tags, only a pair of polynomial coefficients will be distributed among the tags. This procedure is used to prevent threats that can expose the system. The re-construction server is responsible for reconstructing the coefficients pair on the RFID tags, then sends these information to the decryption server after a successful authentication. Upon receiving the pair of coefficient, the decryption server will decrypt the coefficient by extracting the key identifier from the logistic dealer after a successful authentication. The final step is to send the tag's identifier for the logistic information after a successful authentication to complete the final procedure.

Alongside with threshold cryptosystem, the integrity of the data is ensured by using Keccak hash function to prevent any possibility for an attacker to tamper the information during the authentication process. Additionally, the authentication protocol between servers in each step is based on using TLS protocols.

The main features of the system are basically as follows:

- Using secret sharing schemes for distributing multi-RFID tags.
- A package of multi-RFID tags will be authenticated within one reader.
- The system is applicable to use a large scale of RFID tags when the scalability in other schemes is an issue.
- In normal operation, the tag does not need to use a heavy computation just only ensure a mutual authentication between parties.
- The system is considered on using one reader to send the tag information to servers,
- The most complex computation is dependent on servers.

- The system is ideal to be used for low-cost RFID tags that can be applicable with supply chain management scenarios.
- The security and privacy of the system are ensured by using threshold cryptosystem with a hash function.

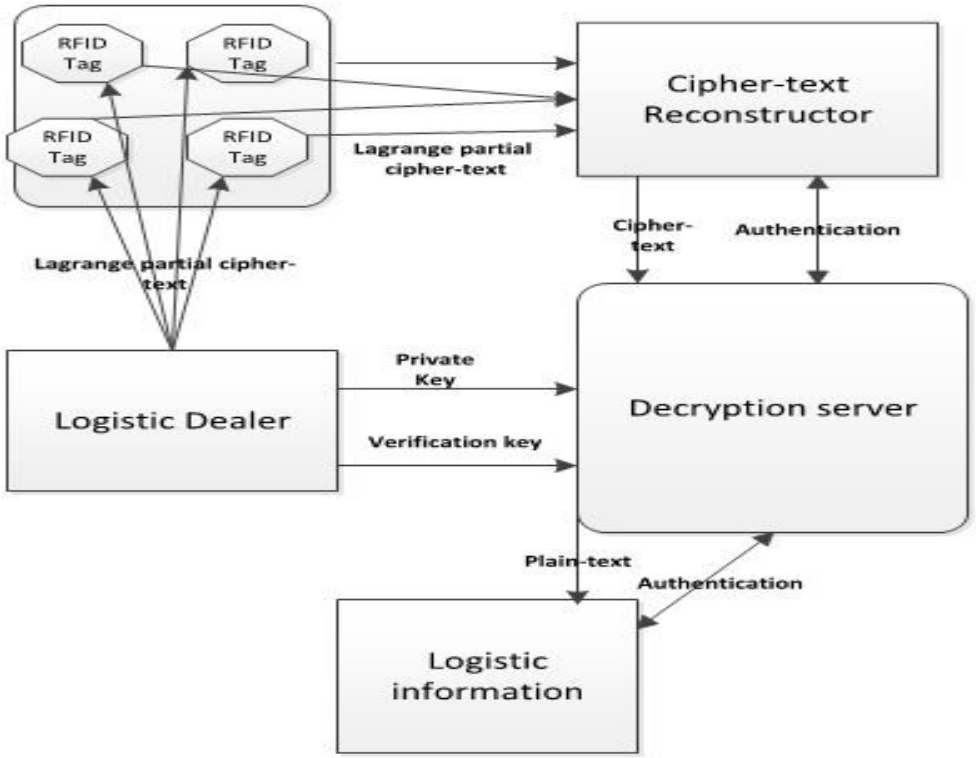

Figure 6. 1:Threshold cryptosystem using multi-RFID tags

6.2.3 Adversary threats

In the threshold cryptosystem, we assume that the communication channel between the tagged package and the reader is insecure therefore there is a chance for an adversary to have the following ability:

- During the authentication, an adversary can eavesdrop and change the message exchanged between the tag and the reader. At the end of the authentication protocol, the adversary wins if the reader returns a successful response.

- An adversary selects a random tag to inject his own message and complete the authentication
- An adversary can intercept the message exchange and monitor the communication or send his own message to get the tag's information
- An adversary blocks the message exchange in order to prevent further message exchanges.
- An adversary can locate the tagged package by tracking the movement of the tags
- An adversary can impersonate tag and act as legitimate tag.
- An adversary can impersonate the reader and act as a legitimate reader.
- An adversary can impersonate one of the communication channels between servers.

6.3 A quorum RFID based system first Approach

Here, we present our approach for the quorum RFID based system that securely distributes tag's key and using encrypted value for the tag identifier. The protocol approach has four phases and relies on combining threshold system with public key cryptosystem.

The threshold system is used to securely distribute the secret message on RFID tags while the cryptosystem is used to encrypt the message before the distribution process.

The protocol uses the elliptical curve version of the ElGamal cryptosystem combined with the Shamir secret sharing scheme. The integrity of the message

exchange is ensured by using the Keccak hash function associated with the threshold cryptosystem

6.3.1 Shamir Secret Sharing Scheme

In 1979, Shamir came up with an idea to share a key within a based polynomial interpolation (Shamir, 1979). The threshold scheme is divided as a secret key within parties. A secret key is defined by a polynomial $f(x)$ such that

$$f(x) = a_0 + a_1 x + a_2 x^2 + \cdots . a_{n-1} x^{n-1} \qquad (6.1)$$

Where the polynomial coefficients are from a finite field F_p, where p is a prime number. The polynomial coefficients have to be different and unique to each party and should not be equal to zero as that leads to revealing the secret key. The idea behind the scheme is to create shares by choosing points on a polynomial $\big(x, f(x)\big)$. These points are distributed randomly to the parties by using equation (1) where $a_0 = s$ and $s \in \mathbb{F}_p$ then find k points in 2-dimension plane $(x_1, y_1), \ldots . (x_k, y_k)$ there is only one polynomial of degree k-1such that $q(x_i) = y_i$ for all i . The secret key is split it out to the points $\big(x, f(x)\big)$, where $\big(x_0, f(x_0)\big)$, the secret keys, at all other points, are called shares.

To reconstruct the secret share from the shared points which are received, the party that involve revealing the secret value need to have the knowledge of any i shares. The secret key is reconstructed by using Lagrange interpolation as shown in equation (6.2) and equation (6.3) respectively.

$$f(x) = \sum_{i=1}^{n} L_j(x) f(x_i) \qquad (6.2)$$

Where

$$L_j(x) = \prod_{i \neq j, j=1}^{n} \frac{x - x_j}{x_i - x_j} \qquad (6.3)$$

6.3.2 6.3.2 ElGamal Cryptosystem

ElGamal cryptosystem was first introduced in 1984 and based on the difficulty of Diffie-Hellman one way trapdoor function which turns the function into a public key cryptosystem ElGamal Cryptosystem can be formed using a cyclic group in which the discrete logarithm problem is intractable.

In ElGamal Cryptosystem there are three processes, the key generating, and the encryption and the decryption process

- **Key generating process:** the sender is responsible for generating the private key and the public key by firstly

1- Chooses a random prime p then computes a random multiplication generator elements $g \in \mathbb{F}_p$.

2- Picks a random number $x \in \mathbb{F}_p$ such that $1 \leq x \leq p-1$ as a private key then computes the public key

$$X = g^x \, mod \, p \qquad (6.4)$$

3- Sends *(p,g,X)* as a public key.

- **Encryption process:** the encryption process starts when receiver chooses a plain-text m such that $m < p$

1- picks $k \in \mathbb{F}_p$

2- computes the cipher-texts such that

$$Y = g^k mod \, p \qquad (6.5)$$

$$C_m = X^k m \, mod \, p \qquad (6.6)$$

- **Decryption process:** After receiving the cipher-text, the sender decrypts the cipher-text to find the message m by computing

$$m = \frac{C_m}{Y^x} = \frac{X^k m}{g^{kx}} = \frac{g^{xk} m}{g^{kx}} = m \qquad (6.7)$$

The elliptical curve version of ElGamal cryptosystem is considered as IND-CPA which can provide security against eavesdropper adversary.

The ElGamal cryptosystem can be used as a threshold cryptosystem. As the cipher-text of ElGamal cryptosystem consists of C_m and Y, these values are encoded into a polynomial $f(x)$ where $f(0) = C_m$ and $f(1) = Y$. The degree of $f(x)$ is equal to $k - 1$.

$$f(x) = C_m + (Y - C_m - a_2)x + a_2 x^2 \,..mod\ p \qquad (6.8)$$

The secret shares values can be determined for any n distinict values of x except 0 and 1. The weights applied to the shares are dependent on the particular combination of k from n.

There are $\frac{n!}{k!(n-k)!}$ such combinations each combination may be expressed as an index j, which runs from 1 through to $\frac{n!}{k!(n-k)!}$.

Generalise back k and n leads to equations (6.9) and (6.10) respectively.

$$C_m = \sum_{i=1}^{n} \alpha_{j,i} f(i)\ mod\ p \qquad (6.9)$$

And

$$Y = \sum_{j=1}^{n} \beta_{j,i} f(i) mod\ p \qquad (6.10)$$

Where $\alpha_{j,i}$ and $\beta_{j,i}$ are the Lagrange coefficients for the particular combination of k shares from n shares, represented by index j.

6.3.3 Dealer initialising phase

As shown in figure (6.1) the system design has four phases, here we will explain the first phase which is the dealer initialising phase.

The logistic dealer server is a main server for generating the whole system. Its responsibility is to generate the system parameter, generate the tags keys and distribute the shared keys among RFID tags.

The logistic dealer phase aim is to protect the identifier of the tag by using the elliptical curve version of ElGamal cryptosystem to encrypt the tag secret key. We assume that it is not possible for revealing the secret key for any parties except the decryption server and that happens when the two servers authenticate successfully.

The dealer phase has three processes, the set up process, the encryption process and the distributing key process. In the setup process, the dealer generates the parameter for the system such that choosing the private and public key and choosing the secret point on the curve. The encryption process involves the encryption of the logistic information about the package. The distributing process is to distribute information among RFID tags.

- **The set up process:**

The first process of the dealer is to choose a prime number, then to define the elliptical curve $E = y^2 = x^3 + ax + b \, mod \, p$ and any point on the curved is defined by coordinates x and y.

The dealer chooses the base point $P = (x, y)$ which satisfies the elliptical curve E. Any point P has order Φ such that $\Phi \mathrm{P} = 0$.

The dealer picks a random $r \in \mathbb{F}_p$ then securely sends the random value to the decryption server.

The dealer chooses a public key by choosing randomly value $q \in \mathbb{F}_p$ such that the public key $Q = qP = (x_q, y_q)$.

The dealer generates the secret point on the curve S such that $S = S.Q = r.q.P = (x_s, y_s)$.

- **The encryption process:**

The second process of the dealer server is to encrypt the critical logistic information of the package into two parts messages m_1 and m_2.

This encryption process is done by transforming the two parts of the message into a cipher-text message. This process includes a point of the elliptic curve by multiplying the first message by the first coordinator of the secret point and alternatively, the second message by the second coordinator of the secret point. Therefore the cipher-text message will be represented as a point C_m on the curve E where $C_m = (m_1 x_s, m_2 y_s)$.

The cipher-message point C_m and the public key point Q are the information that will be hidden in the key distribution process.

- **The Distributing key process**

The distributing key process is used for a quorum system that needs to present n tags. The dealer will first determine the number of tags that needed to store information then generates two polynomial of degree $k-1$ $f_1(x)$ and $f_2(x)$ each with $k-2$ random coefficients. For example, if the package needs four tags to

be presented, then the dealer will generate four shared keys that can be stored in each tag by generating two polynomials of degree thress.

The dealer will use the Shamir secret sharing approach to generate a shared key that will be stored in each RFID tag. The information that are stored on each tag is determined by the dealer and depend on the memory of the RFID tag. Generally, we consider that each RFID tag can store 256 bits of information. After generating the two polynomials, the dealer knows that the secret key point (x_s, y_s) and the public key (x_q, y_q) therefore the dealer will hide the values of the cipher-message and the public key points into the polynomials $f_1(x)$ and $f_2(x)$ respectively. Alternatively, each coordinator of the cipher-message point and the public key point will be represented as secret values for the polynomials. The dealer will keep these values securely in its data base and distribute the shared values from both polynomials into RFID tags. For the size k polynomial using n RFID tags, the dealer generates two degrees $k-1$ polynomials $f_1(x)$ and $f_2(x)$ each with $k-2$ random coefficients.

The values of the first polynomials $f_1(2), f_1(3), \ldots . f_1(n+1)$ alongside with the values of the second polynomial $f_2(2), f_2(3), \ldots . f_2(n+1)$ are stored on n RFID tags respectively. Note that the first two values of the first polynomial and the second polynomial are the secret information that will be not be revealed. The secret information are the values of x_c, y_c, x_q and y_q ,theses information are the values of $f_1(0), f_2(0), \ f_1(1) \text{ and } f_2(1)$.

The determination of the cipher-message and the public key points are not possible to be revealed only just for the cipher-text reconstruction process as a legitimate party.

6.3.4 RFID tags and Cipher-text reconstruction phase

After the successful process of the dealer to distribute the stored information among RFID tags, the step now is to reconstruct the secret information that have been hidden into two polynomials and send these secrets to the decryption process for revealing the critical information about the package.

In this process, the RFID tags will include response to cipher-text reconstruction. The first step is sending a hello message by the reader to the RFID tags for the first identification of the package. Then, the cipher-text reconstruction server will determine the values of x_c, y_c, x_q and y_q from the given information in the RFID tags. As described before, the only information that have been stored in the RFID tags are the values of polynomials $f_1(2), f_1(3), \ldots . f_1(n+1)$ and $f_2(2), f_2(3), \ldots . f_2(n+1)$. To determine the secret keys from each polynomial, the cipher-text reconstruction will use the Lagrange interpolation to reveal the secret information of each polynomial $f_1(0), f_2(0)$, $f_1(1)$ and $f_2(1)$.

In order to determine and calculate these information, the cipher-text reconstruction implements these equations respectively.

The first secrets of the first polynomial are calculated as

$$x_c = \sum_{i=1}^{n} \alpha_{j,i} f_1(i) \ mod \ p, \qquad (6.11)$$

And

$$x_q = \sum_{i=1}^{n} \alpha_{j,i} f_2(i) \ mod \ p, \qquad (6.12)$$

To evaluate the other point, the cipher-text reconstruction calculates the secrets from the second polynomial as

$$y_c = \sum_{i=1}^{n} \beta_{j,i} f_1(i) mod \ p, \qquad (6.13)$$

And

$$y_q = \sum_{i=1}^{n} \beta_{j,i} f_2(i) mod\ p. \quad (6.14)$$

The index j depends on the number of RFID tags responding and only k of the $\alpha_{j,i}$ and $\beta_{j,i}$ coefficients will be non-zero.

At this stage the cipher-message reconstruction has successfully reconstructed the cipher-message C_m and the public key$Q$.

The final stage of the cipher-text reconstruction is to ensure the integrity of the message exchange by hashing the value of the cipher-message and the public key. The integrity of the exchanged message is ensured by using the Keccak hash function$f[1600]$. The value of adding Keccak hash function is to ensure there is no option for an adversary to construct message exchaneged , insert or modify his message and outcome the same hash output with avalue of 256 bits of information. Before sending the hashed value of the secrets, cipher-message reconstruction will be authenticated with the decryption server in order to verify the secrets C_m and Q.

6.3.5 Decryption server phase

After successful authentication between the cipher-message reconstruction and the decryption server, the decryption server phase is to determine the original message of the package. In order to determine the original message of the package, the decryption server will first determine the secret point that has been introduced by the dealer, then calculate the two messages that have been used in the cipher-message point.

This process first determines the point S by deriving the component of the secret point as was calculated. Firstly by using the pre-stored private key, the secret point $S = S.Q \ mod \ p$ thus leads to$S = r.q.P \ mod \ p = (x_s, y_s)$.

To determine the messages of the package information, the decryption server derives the two messages from the cipher-text point C_m, then evaluate the first message m_1 such that

$m_1 = \frac{x_c}{x_s} = x_c x_s^{p-2} \ mod \ p$. Similarly, the second message m_2 is evaluated as $m_2 = \frac{y_c}{y_s} = y_c y_s^{p-2} \ mod \ p$.

The messages m_1 and m_2 will also be hashed by using Keccak hash function $f[1600]$ then sends the outputs from the decryption server after a successful authentication with the recipient of the information.

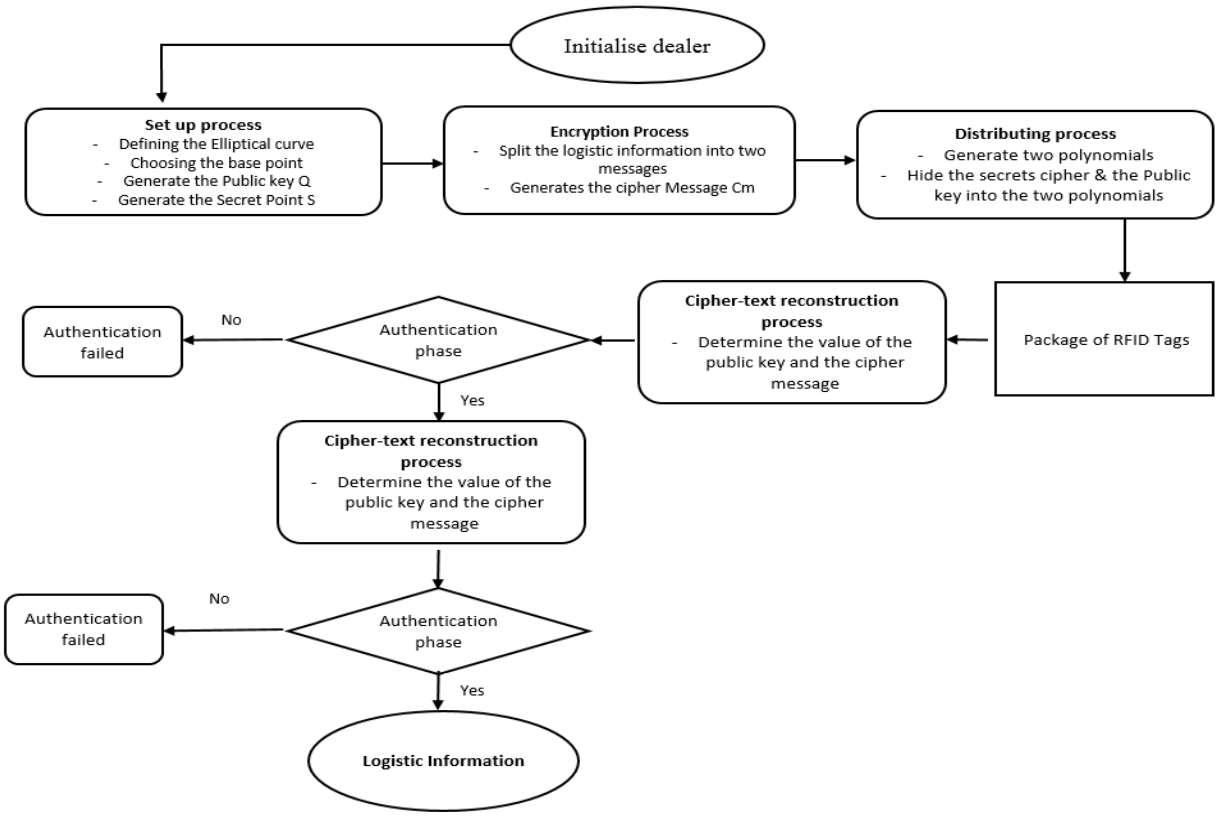

Figure 6.2: Flow diagram for the process of the threshold cryptosystem using multi-RFID tags

6.3.6 Server authentication Phases

This phase involves with using Transport Layer Security (TLS) protocols to ensure privacy between servers. TLS protocols allow servers to authenticate each other using a certificate and exchange an encryption algorithm based on their computational capability to encrypt the data being exchanged between servers. There are several TLS methods that have been proposed such as TLS 1.1, TLS 1.2 and TLS 1.3 is currently proposed. Different cryptographic techniques are used in these methods to achieve security and efficiency by implementing two protocols in the TLS algorithm. These two protocols are TLS record and TLS handshake protocols. The TLS record protocol is responsible for ensuring a private connection between servers. Most of the TLS record protocols depend on using hash function generated by a message authentication code. The TLS handshake is used to authenticate parties by producing an agreement upon using an encryption method and the keys that are used for the authentication. After the agreement is established, parties send their information for verifying the authentication process. The data that has been exchanged is encrypted by the agreement between servers. The flexibility of using TLS leads researchers to implement different cryptographic approaches in TLS (Oppliger, 2016).

In the quorum system, we consider using perfect security method that relaies on using elliptical curve Diffie-Hellman key exchange and certificate based authentication in the TLS authentication between servers. The first authentication is done by the cipher-text reconstruction and the decryption server. As shown in figure 6.2, the authentication session starts by assuming that the secret hashing key k_{hash} and the secret key k_{secret} are known for both

parties. The secret key contains the message $C_m$and Q, while the hashed key contains the hashed value of the messages C_m and Q. In the authentication phase, both servers set up a secure communication between each other using elliptical curve Diffie-Hellman key exchanged and certificate based authentication. The cipher-text reconstruction and the decryption server start their session by generating a prime p, then choosing elliptical curve E and a base point P that satisfies the elliptical curve equation.

The key exchange process will start after the cipher-text reconstruction generates a nonce $\mathbb{F}_p$, then sends q to the decryption server. Upon receiving the nonce q, the decryption server chooses a nonce $r \epsilon \mathbb{F}_p$ and sends back to the cipher-text reconstruction.

Both servers are required to complete the key exchange session by checking the value of $q.rP\ mod\ p$ and $r.qP\ mod\ p$ respectively. If both values are correct, then the key exchange session is complete otherwise reject the session.

Consequently, the cipher-text reconstruction will compute and send the value $p_1 = Sha3(k_{secret}, k_{hash})$ to the decryption server. Upon receivingp_1, the decryption server will fetches p_1^* from secure database, then compares both values p_1 and p_1^*. If both values are equal, then decryption server sends a verification message. Otherwise, the connection will be failed.

The same procedure is required between the decryption server and the logistic information for the final step of the quorum system approach. Same as before, the decryption server and the logistic information set up the communication by using elliptical curve Diffie-Hellman key exchange and generate the parameter for the key exchange. Accept if the value of $q.rP\ mod\ p$ and $r.qP\ mod\ p$ is equal. After that, the decryption server will hash the value of the two messages

m_1 and m_2 using Keccak hash function $f[1600]$. The K_{secret} holds the values of the messages m_1 and m_2 respectively. The K_{hash} holds the hashed value of the original messages m_1 and m_2. The decryption server sends $p_2 = Sha3(K_{secret}, K_{hash})$ to the logistic information, then the logistic information fetches the value $p_2^* = Sha3(K_{secret}, K_{hash})$. If both values are equal then the logistic information sends a verification to the decryption server. Otherwise, the communication failed.

The TLS authentication process in our scheme replaces the idea of using $HMAC$ with an encryption method and replace it with the using of Keccak hash function for keys and the authentication process. The Diffie-Helman key exchange ensures the forward security, so there is no chance to track the previous session.

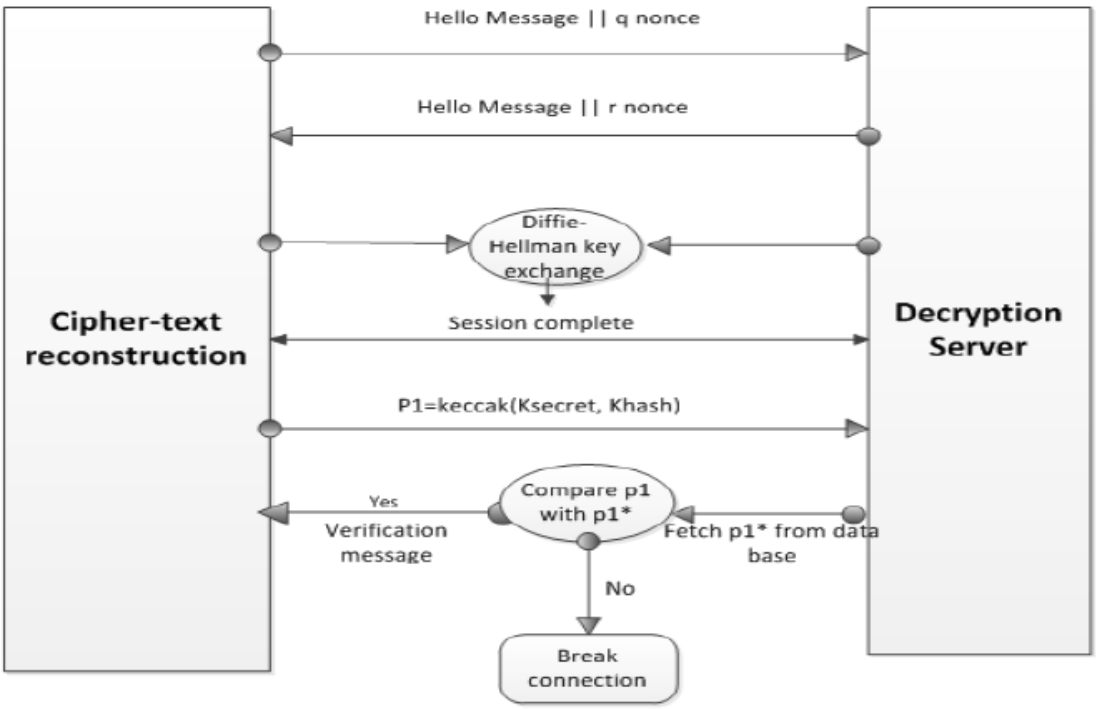

Figure 6. 3: TLS Keccak authentication protocol

6.3.7 Security analysis

As described before, the quorum RFID based system uses elliptical curve version of ElGamal cryptosystem and Shamir secret sharing schemes. Therefore, we consider that any RFID system is considered to be secured if the following properties are satisfied:

- **Correctness**

Correctness ensures that when a tag stores its secret information, it follows there is a verifier that accept tags. Thus, only legitimate tags are accepted. The experiment for an adversary is considered able to access to the database oracles in order to change the situation of the system.

In the case of the quorum RFID based system, the use of threshold ElGamal ensures that only uncorrupted RFID tags have the ability to be verified. However, we consider if the adversary launches his own experiment to check the status of the system. Similar notion will be used as in section 4.2 to prove the correctness property of the system.

For the first step, the challenger initialises the system and sends the InitiateReader algorithm with the public key to an adversary. The adversary chooses a random uncorrupted tags information by interacting the system. The adversary then executes a new interactive protocol between the tag and the reader. If the adversary wins the output then the correctness property does not satisfied. The quorum RFID based system satisfy the correctness property because only un corupted RFID tags have secret shares that have been generated by dealer. When cipher-text reconstruction obtains the secret shares, directly use reconstruction procedure to obtain the secret values. The secret

values have been distributed through a polynomial then only shared keys have been stored in the RFID tags. The cipher-text reconstruction will refuse any corrupted tags. Therefore, the adversary has a negligible advantage to win the procedure for accepting corrupted tags.

- Soundness

Soundness property ensures that it's infeasible for an adversary to forge tag's information. In other words, fake tags cannot be accepted by the system. As a package of RFID tags moves from place to another, there is a chance to replace a legitimate RFID tag with another illegitimate RFID tag. To ensure the soundness property, an RFID system needs to refuse illegitimate tags to go through other processes. The legitimate RFID tags are issued by the dealer and the information of tags has been updated. Therefore, an adversary cannot have the ability to update the tag's information. Consequently, we need to consider if the adversary launches his own experiment to break the soundness property. The challenger initialises the system and sends the InitiateReader algorithm with the public key to an adversary. The adversary is able to interact the system and output a challenge tag with a new identifier. The adversary now successfully creates his own challenge tag and produces it to the system.

In the quorum RFID based system, only legitimate RFID tags have secret shares that are generated by using threshold ElGamal. In this case, the adversary needs to know the way of how the tag information has been stored in order to complete the process. Illegitimate tags will be directly refused by the cipher-text reconstruction as they do not have valid shares similar to other legitimate tags. Therefore, the system is provided with soundness property as the adversary has no chance to complete the verification process.

6.3.8 Evaluation in terms of RFID security threats

The quorum RFID based system can be used in low-cost RFID tags. The information that will be stored in the tags are only shared keys that have been generated two polynomials and the secret information about the package are encrypted by using ElGamal cryptosystem with the elliptical curve. Therefore, the complexity of the protocol is very low in terms of tag calculation and reader calculation. The reader only reads tag's information and send it to servers while keeping the heavy computation running over servers. Additionally, the dealer can store hashed value by using Keccak hash function alongside with the shared key to provide mutual authentication.

In order to analyse the security of the protocols, the following security attacks on RFID systems need to be considered to determine the functionality of the system.

- Anonymity

The quorum RFID based system provides anonymity by generating random shared keys that have been derived from the secret keys. Therefore, these shared keys are randomly generated from the encrypted secret keys and cannot obtain the original secret shares from the shared keys.

- Resistance to the man-in-the-middle-attack:

The quorum RFID based system satisfies the property to resist this attack. As the quorum RFID based system has three phases, we consider this attack in term of the process of the tag to cipher-text reconstruction only.

Suppose that an adversary establishes an independent communication with the RFID tags and can intercept the message communication between the dealer

and RFID tags and tries to redirect them. In the quorum system, the cipher-text message is distributed as secure shares by using Shamir secret sharing and only the dealer can know the cipher-text message and only the cipher-text reconstruction can reconstruct the cipher-text message and compute the secret key and authenticate with the decryption server. Hence the adversary cannot decrypt the cipher-text.

- Resistance to Replay attack:

An adversary intercepts the data transmission between the logistics dealer and the RFID tags. In case of successful interception of the message exchange between the logistic dealer and RFID tags, the adversary will receive some key shares keys but cannot obtain a quorum of key shares needed to reconstruct the cipher-text message and cannot exploit the information between the RFID tags and the logistic dealer.

- Resistance to tracking attack

The tracking attack involves tracking the user behaviour and the movement of the tags. Since the RFID tags consist only of randomly chosen shares and the secret shares are stored in the logistic dealer, the only information available to the eavesdropper is the random shares. The server reconstruction does the only way to compute the secure shares from the original shares. Thus, there is no useful information available for tracking attack.

- Denial of service

The denial of service consists of blocking the transmission between the logistic dealer and RFID tags in order to lose their synchronisation of them by sending a large number of tag's information. In that case, RFID tag shares may be checked for authenticity to avoid DoS attack. However, without using Keccak

hash function, an adversary can write some content that affects the verification process. Even though, this type of attack is considered to be limited as only legitimate parties are involved with the verification and decryption server. However, we consider that it may be causing the denial of service with the system.

- Cloning Attack

This type of attack is considered one of the major attack against the quorum RFID based system as the system is assumed to be used with low-cost RFID tags. The low-cost RFID tags cannot perform authentication with a reader. Therefore, an adversary can clone the content of the RFID tags in the stage of the movement of the package as it impossible in the stage of the generating keys by the dealer. However, this type of attack cannot perform for further stages as the servers can recognise any changes with the tags by checking the tag status. Even though, we consider the case when an adversary can impersonate the reader in order to clone the tag's information by using an illegitimate reader.

6.4 Anti-cloned quorum RFID based system (second approach)

Although the first quorum approach provides an excellent solution for privacy and security, the system has two drawbacks. An adversary can clone the tag itself in order to copy the information about each tag. This can happen when there is a chance to impersonate the reader itself which lead tags to respond to an illegitimate reader. The second possible attack is the denial of service attack, this can happen when an adversary tries to tamper the tag's information in order

to stop the verification process. The information of the tags cannot be used until the session of the decryption server is completed. However, we consider these threats against the previous approach even these threats can be a complex and heavy in the computation for an adversary to perform the final steps with other servers.

Therefore a new approach will be introduced to perform better security and privacy against cloning attack and other attacks that can affect the system. Based on the same idea of the first approach, the anti-cloned system is also used for solving the problem of distributing multi-RFID tags in a package and authenticating these tag within a reader. The procedure also is illustrated in figure 6.1 and based on using encrypted shared keys alongside with hash keys and involve with using symmetric key encryption by the tag to ensure privacy. The shared keys are generated by using polynomials that keep the encrypted secret information as points inside each polynomial .

The new approach is designed to be the anti-cloned approach that provides a strong encryption in the tag itself as well as provides mutual authentication for the tag to reader authentication to avoid penetration of the reader or tags. The dealer system will use Cramer-Shoup lite scheme for the secret information about the package and distribute secrets by generating shared keys using Shamir secret sharing scheme.

6.4.1 System features

In the anti-cloned system, an RFID tag stores a random shared key that has been encrypted and distributed through a dealer by using elliptical curve version of Cramer-Shoup lite scheme combined with Shamir secret sharing scheme.

This combination offers a secure design for any system as Cramer-Shoup provides IND-CCA property that can prevent active and passive attacks on the system. The main feature of the anti-cloned system system is it can be implemented in passive RFID tags that can perform AES encryption. Therefore, a strong privacy and security will be achieved by the combination of the secured system with a secure tag that will lead to having the anti-cloned quorum RFID based system that can be used in passive RFID tags.

Similar to the previous approach, the anti-clone approach will have a dealer phase that generates an elliptic curve with Cramer-Shoup cryptosystem and generates polynomials to distribute the secret shares among RFID tags. The main difference to the previous approach is the tag authentication. The tag authentication will ensure the privacy and security of the tags during transiting from stage to other stages. The confidentiality and integrity of the tag authentication phase will be ensured by using AES cryptosystem with a Keccak hash function to prevent further attacks. The other verification steps by cipher-text reconstruction and decryption server will be slightly different from the previous approach.

6.4.2 Passive RFID tag

In the anti-cloned system, the tags themselves need to be inexpensive for the system to be viable. Up until now, some passive RFID tags had no security apart from manufacturers' ID, but disreputable manufactures clone these tags. Therefore, we consider using new NXP tag called NTAG DNA as it passive RFID tag support and mutual authentication by using symmetric key or

asymmetric key cryptosystem. Thus, these types of tags can solve the problem of cloning attack.

The NTAG 413 DNA tag support mutual authentication by ECC signature or by using AES cryptosystem. The mutual authentication is challenge-response protocol to authenticate tags with a reader. In the anti-cloned system, we will use a mutual authentication protocol that uses three rounds of AES protocol with 128 bit support. The block diagram of the NTAG 413 DNA is shown in figure 6.4.

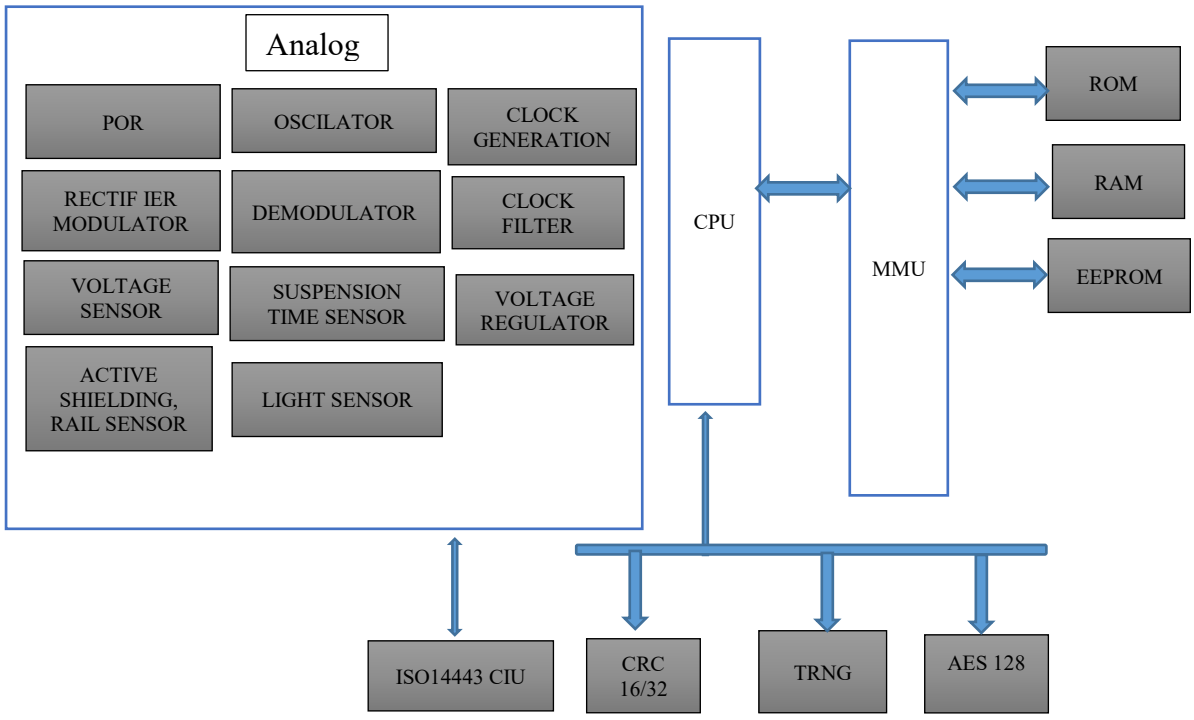

Figure 6.4:Block diagram of NTAG 413 DNA (NXP, 2017)

6.4.3 Cramer-Shoup lite scheme

Cramer and Shoup (1998) developed a lite cryptosystem that its security is based on the hardness of the Diffie-Hellman decision. The scheme was developed to achieve security against the adaptive chosen cipher text attack.

The ElGamal cryptosystem does not satisfy the property of the IND-CCA and only satisfies the IND-CPA. Therefore, Cramer-Shoup scheme achieves the required level of security by using a one-way hash function in the encryption process.

In this section, we will describe the Cramer-Shoup lite scheme that does not need hash function, and it is proved to be secure in terms of IND-CCA1. The lite version of Cramer-Shoup scheme has three processes: the key generation process, the encryption process and the decryption process.

- **Key generation process**

The key generation process involves with generating private and public keys by computing the following steps.

1- Selects a group G of prime order q.

2- Selects a prime $p \in G$ such that $p - 1 = 2q$.

3- Selects two primitive roots g_1 and g_2 of the prime p such that $g_1, g_2 \in G$.

4- Selects four private keys such that $x, y, a, b \in [0, \dots, q-1]$.

5- Compute the first public key h such that

$$h = g_1^x . g_2^y \; mod \; p \qquad (6.15)$$

6- Computes the second public key C such that

$$C = g_1^a . g_2 b \; mod \; p \qquad (6.16)$$

7- Sends (g_1, g_2, C, h) as a public key to the receiver.

- **Encryption process**

The encryption process starts when the receiver chooses a message $m \in \mathbb{F}_p$, then randomly selects $r \in [0, \dots q-1]$. The sender computes and sends the cipher text (u, v, w, e) as follows:

1- $u = g_1^r \ mod \ p,$ (6.17)

2- $v = g_2^r \ mod \ p,$ (6.18)

3- $w = m.h^r \ mod \ p,$ (6.19)

4- $e = C^r \ mod \ p.$ (6.20)

- **Decryption process**

The decryption process starts to decrypt the cipher (u, v, w, e) as follows:

1- The sender uses the private keys a and b to check if the value of $e = u^a.v^b \ mod \ p$, otherwise reject.

2- If the value is correct, then the sender uses the private keys x and y to check the value of the message m by computing $m = \frac{w}{u^x v^y} \ mod \ p$.

6.4.4 Logistic dealer initialising phase

Similar to the first approach, the logistic dealer initialising phase has the responsibility for setting up the whole system by configuring out the key parameters that are needed in another verification stage. Not only that, the system provides protection for the data exchange by encrypting the secret information then hide these secrets as points in polynomials then distribute these points into required number of RFID tags.

This phase involves three processes; the setup process, the encryption process and the key distributing process.

- **Set up process**

The setup process involves with selecting an elliptical curve E over a finite field $\mathbb{F}_p$ such that $y^2 = x^3 + ax + b \; mod \; p$. There is a base point $P = (x, y)$ whose order is Φ such that $\Phi \mathrm{P} = 0$. The dealer chooses two arbitrary values g_1 and g_2 then randomly selects four private keys x, y, a, b. The dealer next step is to compute the public key points on the curve Eas follows:

$$cP = g_1^a g_2^b \; P \; mod \; p = (x_{c1}, y_{c1})$$

$$hP = g_1^x g_2^y \; P \; mod \; p = (x_{c2}, y_{c2})$$

- **Encryption process**

The encryption process is run after successfully generating a random$r \in \mathbb{F}_p$. The critical logistic information about the package m encrypted into cipher-texts by generating secret points $S_1 = (x_{s1}, y_{s1})$ and $S_2 = (x_{s2}, y_{s2})$ on the curve E. The generation of cipher-text and the secrets points are as follows:

$$uP = g_1^r \; P \; mod \; p,$$

$$vP = g_2^r P \; mod \; p,$$

$$wP = m.h^r \; P \; mod \; p = (x_{s1}, y_{s1})$$

$$eP = C^r P \; mod \; p = (x_{s2}, y_{s2})$$

- The Key Distributing process

In order to split secret points and public key points, the logistic information generates four polynomials $f_1(x), f_2(x), f_3(x)$ and $f_4(x)$ each with $k-2$ random coefficients. The values of the secret points and the public keys will be hidden into these polynomials such that the value of $f(0)$ and $f(1)$ will represent the coordinate of secret key and public key points respectively. In other words, the value of public key points will be $f_1(0) = x_{c1}, f_2(0) = y_{c1}, f_1(1) = x_{c2}$ and $f_2(1) = y_{c2}$. Consecutively, the value of secret points will be represented as $f_3(0) = x_{s1}, f_4(0) = ys_1, f_3(1) = x_{s2},\ f_4(1) = y_{s2}$. The values $f_1(2)$ through $f_1(n+1)$, $f_2(2)$ through $f_2(n+1)$, $f_3(2)$ through $f_3(n+1)$ and $f_4(2)$ through $f_4(n+1)$ are stored on n RFID tags.

Now the information in RFID tags are stored with shared keys that are not related to the original secret information and ready to be used. Additionally, a hashed value of each shared key will be stored in tags as a key for the mutual authentication process.

6.4.5 A mutual authentication phase and cipher-text reconstruction

This phase is the main feature for the anti-cloned quorum RFID based approach as it provides a mutual authentication between the tagged package and a reader before the stage of the cipher-text reconstruction. This phase starts when the package of RFID tags moves from the logistic dealer to transit toward its destination. Upon receiving the package, the first step is to scan the package and get the related information about the package. Before sending the package information to the cipher-text reconstruction, a mutual authentication process

will be involved to check the legitimacy of the process and to prevent cloning attack by an illegitimate reader.

As RFID tags can perform AES encryption process, we will consider the following mutual authentication procedure.

Both tags and cipher-text reconstruction have master keys $k_i = Sha3(y_i)$ where $y[i]$ is the shared key that is stored in each tags. The cipher-text reconstruction computes $C1 = AES(m, k_i)$ and send it to the reader. The reader challenge tags with $C_2 = AES(C1, K_i2)$ and sends $C2$ and K_i2 to tag and wait for the tag's response. Each tag will accept the challenge by computing $AES(C1, K_i2)$ and by using the master key k_i. If the challenge response is correct then tags can be read by a reader. Decrypt tag's responds with $Sha3(y_i)$. In the above procedure, only a valid tag has the correct AES key stored on it; thefore tags cannot be faked unless the AES keys are compromised.

Figure 6.4 illustrates the idea of the mutual authentication protocol.

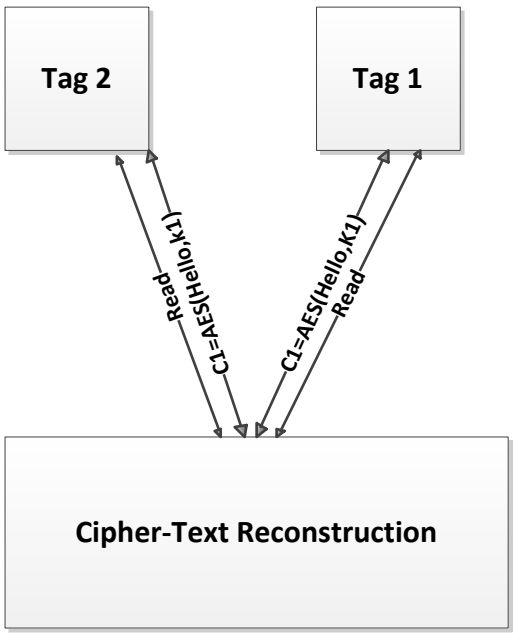

Figure 6. 5: AES mutual authentication scheme

Upon receiving the tag response, the reader will send the shared keys to the cipher-text reconstruction for extracting the information from the shared keys and then sends these information to the decryption server.

The cipher-text reconstruction determines the secret information by using the Lagrange interpolation and implement the following equation that determines the secret points and the public key points.

$$x_{c1} = \sum_{i=2}^{n} \alpha_{j,i} f_1(i) \, mdo \; p, \tag{6.21}$$

$$y_{c1} = \sum_{i=2}^{n} \beta_{ji} f_1(y) \; mod \; p, \tag{6.22}$$

And

$$x_{c2} = \sum_{i=2}^{n} \alpha_{j,i} f_2(i) \, mdo \; p, \tag{6.23}$$

$$y_{c2} = \sum_{i=2}^{n} \beta_{ji} f_2(y) \; mod \; p, \tag{6.24}$$

To determine the secret points

$$x_{s1} = \sum_{i=2}^{n} \chi_{ji} f_3(y) \; mod \; p, \tag{6.25}$$

$$y_{s1} = \sum_{i=2}^{n} \delta_{ji} f_3(y) \; mod \; p, \tag{6.26}$$

And

$$x_{s2} = \sum_{i=2}^{n} \chi_{ji} f_4(y) \; mod \; p, \tag{6.27}$$

$$y_{s2} = \sum_{i=2}^{n} \delta_{ji} f_4(y) \; mod \; p. \tag{6.28}$$

At this stage, the cipher-text reconstruction has reconstructed the value of secret points and the public key points. Before sending these information to the

decryption server, the TLS authentication process will be implemented in order to have a mutual authentication between the two servers. Before that, the cipher-text reconstruction will hash the value of the revealed information and send these hashed values to decryption server after a successful authentication process. The same procedure in section 6.3.5 is used in order to get a perfect security mutual authentication process.

6.4.6 Decryption server phase

At this stage and after successful authentication, the decryption server is now able to compute and derive the critical logistic information about the message from the secret points and the public key points. The first verification process is to check if the value of $= u^a . v^b \ P \ mod \ p$. If the value is correct, the decryption server will output the message m by computing

$m = \frac{wP}{u^x v^x P} \ mod \ p$. If the value is correct, then the decryption server will hash the value of m by using Keccak hash function and furthermore will authenticate with the logistic information.

6.4.7 Security and privacy analysis

The anti-cloned quorum RFID based system satisfies the following properties:

- Correctness

The system ensures the correctness property as only legitimate RFID tags have the ability to complete the verification process through a reader, then the reader sends the information about tags to the cipher-text reconstruction. Nevertheless, if corrupted RFID tags have been inserted, the system will

directly recognise these tags as there are no similar information between corrupted tags and uncorrupted tags. Thus, the only uncorrupted tags are accepted as they produce the mutual authentication process and have valid shared keys that are previously stored.

Therefore, the experiment for an adversary to create RFID tags with their ID information and execute these tags with the interactive protocol is negligible. The output will be recognised that the tag cannot hold the same pre-stored information which leads to the satisfying correctness property.

- **Soundness**

Soundness property is ensured by the system as there is no way for the system to accept illegitimate tags. If an adversary interacts the system through his own experiment, the adversary needs to know the way of how the tag's information has been stored. As the shared keys are generated by using IND-CCA cryptosystem then distributed randomly by polynomials, there is no chance for an adversary to output his own identifier through the system. The adversary experiment will fail to output a correct bit information in the mutual authentication phase and before the information goes through verification by cipher-text reconstruction. Therefore, the anti-cloned RFID based system is sound as the adversary experiment to succeed in the interaction tag identity is negligible.

- **Privacy**

The privacy of the system is ensured as there is no chance for an adversary to access to the tag authentication protocol. However, we will consider if an adversary knows the way of the mutual authentication procedure and tries to

create two tags to check the correspondence of his tags in order to break the privacy of the system. The main idea for the adversary is to generate corrupted tag then initialise the authentication protocol with the reader and receive the challenge then respond to the challenge by his own message. The adversary will receive the hello message with a key by a reader and send these message to the uncorrupted tag in order to compute the challenge-response procedure. As the adversary tags have not master keys then the adversary will fail to read his corrupted tags by the reader. The adversary experiment will fail to produce the challenge-response authentication. Therefore, the adversary cannot recognise his corrupted tag.

6.4.8 Evaluation in terms of RFID security threats

The anti-cloned quorum RFID based system can satisfy the following RFID threats:

- **Anonymity**

The system provides anonymity by generating random keys by a reader and random shared keys that have been pre-stored by the dealer. These keys are randomly generated in each session and can hardly obtain the original secret shares from the shared keys until the other verification and authentication phases are completed.

- **Mutual authentication**

By using AES challenge-respond phase, the mutual authentication property is achieved by a reader and a package of RFID tags before data being exchanged.

The tags can verify the challenge and respond to the challenge to complete the authentication. The first step of the mutual authentication involves with a hello message encrypted by AES, then tag complete the other challenge phase that involves AES encryption in order to complete the authentication and read the tag's information.

- **Replay attack:**

Suppose that an adversary with the ability to eavesdrop the data exchange between the RFID tags and a reader. If the adversary successfully eavesdrops on the message exchange, then producing a new message from the previous session is not compatible as each tag compute the response with the master key, and the master keys are only defined in the RFID tags and in the cipher-text reconstruction database. Thus, this yields to different values in each authentication session. Although the reader can gain the value of the tag's response, it can't get the correct response from the tags and can't know the tag's ID as only shared keys are stored in each RFID tags. As a result, the adversary cannot pass through the mutual authentication process and cannot replay the query to the server as a valid tag. Therefore, the system is resistant to replay attack.

- **Man-in-The-Middle Attack**:

Suppose an independent communication between an attacker and RFID tags has been intercepted. In that case, an adversary can intercept the message communication between the reader and RFID tags and try to redirect them or block by producing false messages. In the system, the secret messages are distributed as of secure shared keys by using Shamir secret sharing scheme,

and only the dealer knows the value of secrets, therefore, only the cipher-text reconstruction can reconstruct the secret shares from the shared keys and authenticate with the decryption server for the original message. If the adversary tries to produce the false message, the verification process will detect any modification in the original message. Hence the adversary does not have a related information be successful in the verification, and the system is secured against this type of attack.

- Tracking attack

The tracking attack consists some of the users' behaviour and tags movement. All of the tag's information are encrypted as random shares then only information to be considered is the master keys. Since the adversary has no control on the master keys values and all of the information are confidential, the attacker has to break the AES to access the tag's information which is not feasible. However, if the adversary succeeds in these proceedings, all of the information that found on the tags are shred keys which are not related to the secret details on each tag. Additionally, after each session different keys are produced therefore the system is unlikable, and the adversary cannot distinguish a typical response to obtain the tag location or the private information on each tag.

- Impersonation tag and cloning attack

Suppose that an adversary tries to copy and reuse tags' information by impersonating the tags shares. Since the shares of RFID tags are generated randomly, and these shares can be used for one authentication session and not valid for other sessions. In both cases for eavesdropping tags shares, an

adversary needs to obtain the secret shares from the server in order to impersonate tags. In the case of cloning attack, the adversary must obtain the tag's information.

Obtaining the tag's information requires completing the authentication session. Therefore, an attacker needs to compromise the AES cryptosystem and the master key in order to compromise the tag's information.

- **Denial of service:**

In this type of attack, an adversary attempt to intercept the message exchange between the RFID tags and the reader that contain the information for the authentication process. The idea of the adversary is to numerously replay the intercepted message in order to enforce the system from completing the verification session. Or to block the transmission between the reader and RFID tags which leads to failure in the system communication. In this case, the authentication process updates the keys in each session. Therefore, both tags and reader are available to communicate at every authentication session. Moreover, the records of each tag are stored in the cipher-text reconstruction in order to check the authentication process. Thus, the system can resist this attack.

- **Forward and backward traceability:**

These types of attacks consist of corrupting some tags challenge information at a specific time. In that case, if an adversary can corrupt some information about the tag information, he /she cannot link past or future execution from the corruption. Because no public information are available for the adversary to be tracked. The only information that is stored in the tag is share values from the

original secret key and it is hard to reconstruct the shares from random polynomial and know the way for decrypting the secret key, therefore there is no danger about future or previous information to be revealed especially when using the threshold version of Cramer-Shoup lite scheme.

6.5 Implementation and worked examples

The section will exemplify implementation procedure. In this section we haven't presented any hardware implementation. However, the bench test for the anti-cloned quorum system will consider the previous hardware works from Feldhore et al., (2004) , Chew et al.,(2010) and Fu et al (2014).

Feldhore et al., (2004) proposed an authentication protocol that is based on using AES with RFID tag. In their scheme, the implementation of data path of an AES-128 deisgn has a current consumption of 8.15 μA on 0.35 μm CMOS process. It operate at frequence of 100 kHz and needs 1.016 clock cycle of encryption 128-bit data block. The required hardware is estimated to be 3.595 Gate Equevelant (GE). In addition to Feldhore et al (2004) work, Chew et al., (2010) tested the time cost for the AES encryption and decryption process with RFID tag by using Demo RFID tag. The time cost experiment for running AES 128, 192, and AES 256 showed that the encryption process for AES 128 took only 2.8 Millisecond (ms) and the decryption process was 3.1 ms. While the time cost experiment for running AES 192 showed an extra time in running encryption process and decryption process 3.3 ms and 3.6 ms respectively. The last test was to estimate the time cost for running AES 256. The result showed that the time cost for encryption process is 4.1 ms while the decryption process was 4.3 ms .

Fu et al., (2014) designed a UHF rfid tag chip with AES encryption engine. Their design based on proposing an authentication protocol for ISO 18000-6C RFID system. In their protocol, the implementation of AES 128 in a tag chip has a current consumption is 20.9 μW on Semiconductor Manufacturing International Coperation (SMIC) 0.13 μm EEPROM process. The required hardware is estimated to be 4952 Gate Equevelant (GE).

The software implementation has been achieved by using the GNU Multiple Precision Arithmetic Library (GMP). The GMP library is a robust open source library that operates on integers, rational number, and float numbers. In the implementation, we use the prime p from the form $p = 2^{\alpha} - \gamma$. This type of prime is called pseudo-Mersenne prime and it is used to enhance the performance and to ensure there are no backdoors that could compromise security [10]. The curve selection is based on using Weierstrass equation but with the form $y^2 = x^3 - ax + b \, mod \, p$. This curve selection is much more efficient regarding of security (Costello et al,.2016). Valid curve satisfies the condition $4a^3 + 27b^2 \neq 0$.

Before going through worked examples for the quorum RFID based system and the anti-cloned quorum RFID based system, basic operation algorithms will be presented. These algorithms are mainly focus on the operation over the elliptical curve, the Shamir secret sharing scheme algorithm and the Keccak hash function algorithm.

6.5.1 Elliptical curve operations

An elliptical curve over a finite field of prime numbers has operations to produce valid points over the elliptical curve. Such operations over the finite field are

addition, subtraction, multiplication and multiplication inversion. These operations are used to produce operations over an elliptical curve such that points addition, points doubling, and scalar multiplication.

- **Addition over $\mathbb{F}_p$**

Algorithm 6.1 Addition over $\mathbb{F}_p$

$\boldsymbol{Input}$: A, B
$\boldsymbol{Output}$: $C = A + B$

1- $C = A + B$
2- $If\ C > p\ then$
3- $C = C - p$
4- $End\ if$
5- $Return\ (C)$

- **Subtraction over $\mathbb{F}_p$**

Algorithm 6.2 Subtraction over $\mathbb{F}_p$

$Input$: $A, B\ d = log_2 p + 1$
$Output$: $C = A - B$

1- $C = A - B$
2- $If\ C[d+1] == 1\ then$
3- $C = C + p$
4- $End\ if$
5- $Return\ (C)$

- **Multiplication over $\mathbb{F}_p$**

Algorithm 6.3 Multiplication over $\mathbb{F}_p$

$\boldsymbol{Input}$: $A, B, p\ d = log_2 p + 1$
$\boldsymbol{Output}$: $C = A * B * 2^{-d}$

1- $u = o$
2- $for\ (i = 0\ to\ (d-1)\ do$
3- $u = u + A[i], B$
4- $if\ (u[0] == 1) then$
5- $u = u + p$
6- $End\ if$
7- $u = u$
8- $end\ for$

9- $Return\ u$

- **Multiplication Inversion over $\mathbb{F}_p$**

Algorithm 6.4 Multiplication Inversion over $\mathbb{F}_p$

$\boldsymbol{Input}: p, a \in [1, p-1]$

$\boldsymbol{Output}: a^{-1}$

1- $u = a, v = p$
2- $x_1 = 1, x_2 = 0$
3- $while\ (u \neq 1\ \&\ v \neq 1)\ do$
4- $\quad while\ (u[0] == 0)\ do$
5- $\quad u = u$
6- $\quad if\ (x_1[0] == 0\ then\ x_1 = x1$
7- $\quad else \quad x1 = x_1 + p$
8- $\quad end\ if$
9- $end\ while$
10- $if\ u \geq v\ then\ u = u - v, x_1 = x_1 - x_2$
11- $else\ v = v - u, x_2 = x_2 - x_1$
12- $end\ if$
13- $end\ while$
14- $if\ (u == 1)\ return\ (x_1 mod\ p)$
15- $else\ return\ (x_2 mod\ p)$
16- $end\ if$

- **ECC Point addition**

Algorithm 6.5 ECC point addition

$\boldsymbol{Input}: E, \mathbb{F}_p, P = x_1, y_1, Q = (x_2, y_2), P \neq Q$

$\boldsymbol{Output}: P + Q = (x_3, y_3)$

1- $A1 = x_2 - x_1$
2- $A2 = y_2 - y_1$
3- $A3 = Multiplication\ Inversion(A1)$
4- $A4 = Multiplication(A3, A2)$

5- $A5 = Multiplication(A4, A4)$

6- $A6 = (x_1 + x_2)$

7- $A7 = (A5 - A6) = x_3$

8- $A8 = Multiplication(A4, A6)$

9- $A9 = Multiplication(A4, A8)$

10- $A10 = (A8 - y_1) = y_3$

- ECC point doubling

Algorithm 6.6 ECC point doubling

Input: $E, \mathbb{F}_{\boldsymbol{p}}, P = (x_1, y_1)$

Output: $2P = (x_3, y_3)$

1- $A1 = x_2 * x_1$

2- $A2 = 3 * A1$

3- $A3 = (A1 + a)$

4- $A4 = (2 * y_1)$

5- $A5 = Multiplication\ Iversion(A4)$

6- $A6 = (A3 * A5)$

7- $A7 = (A6 * A6)$

8- $A8 = (2 * x_1)$

9- $A9 = (A7 - A8) = x_3$

10- $A10 = (x_1 - x_3)$

11- $A11 = (A6 * A9)$

12- $A12 = (A10 - y_1) = y_3$

- ECC Scalar Multiplication

Algorithm 6.7 Scalar multiplication

Input: $E, \mathbb{F}_{\boldsymbol{p}}, P = (x_1, y_1)$

Output: $2P = (x_3, y_3)$

1- $k = n - 1$

2- $(x_1, y_1) = (x_0, y_0)$

3- $i = \log_2 k + 1$

4- $for\ (j = i\ to\ 2) do$

5- $(x_2, y_2) = ECC\ Point\ Doubling(x_1, y_1)$

6- $if\ (k[j-2] == 13$

7- $(x_1, y_1) = ECC\ point\ Addition(x_0, y_0), (x_2, y_2)$

8- $else$

9- $(x_1, y_1) = (x_2, y_2)$

10- $end\ if$

11- $end\ for$

12-$return\ (x_1, y_1)$

6.5.2 Shamir secret sharing scheme

Shamir secret sharing scheme consists of two algorithms, generating shares and reconstructing
Shares.

- Generating shares

Algorithm 6.8 Shamir secret sharing generated shares

$\boldsymbol{Input}$: $two\ secret\ a_0, a_1, n\ number\ of\ needed\ shares, threhold\ k, prime\ p$

$\boldsymbol{Output}$: $shares(x_i, y_i)$

1- $a[0] = a_0$

2- $a[1] = a_1$

3- $for\ (j = i\ to\ 2) do$

4- $a[i] = rand(a[i])$

5- $End\ for$

6- $for\ i = 1\ to\ n\ do$

7- $y = 0$

8- $for\ (j = 1\ to\ k\ do$

9- $y = y + a[i].i^{j-1}$

10- $end\ for$

11- $shares[i] = (i, y)$

12-$end\ for$

13-$return\ shares$

- Reconstructing shares

Algorithm 6.8 Shamir secret sharing generated shares

Input: $shares, prime\ p$

Output: $two\ secret\ a_0, a_1$

1- $a[0] = interpolation(Share_1, \ldots..share_n)$

2- $a[1] = Interpolation(Share_1, \ldots share_n)$

3- $return\ a[0], a[1]$

6.5.3 Keccak hash function

The procedure for generating Keccak hash function *f [1600]* starts as follows :

1- Chooses the rate *r* and the capacity *c* =1600such that $[r + c] = 1600$, where *r =576* and *c=1024.*

2- Chooses the input value of message and the input bytes provided in the input message.

3- Initialize the state as an array $s[x, y] = 0, \forall(x, y) = 0 \ldots 4, 0 \ldots 4)$.

4- Pads the message $M(K_{secret}, K_{hash})$ into blocks *P* such that,
$P = ||0x01|0x01|| \ldots ||0x01||$
$P = P \oplus 0x00||0x00||0x08$

5- Absorbs the last few bits and add the first bit of padding for every block *Pi* in *P*
$S[x, y] = S[x, y] \oplus P_i[x + 5y]$
$S = Keccak\ f[r + c](S)$.

6- Squeezes out all the output blocks
Z=empty string
While output >0
$Z = Z||S[x, y]$,
$S = Keccak - f[r + c](S)$
$return\ Z$

$Output: P_1 = Keccakhash(K_{secret}, K_{hash})$

6.5.4 Worked example 1:

This example is 256 bit implementation of the Quorum RFID based system. The system uses ElGamal threshold cryptosystem to hide two secrets per polynomial and reconstruct secrets by using Lagrange interpolation. The TLS authentication protocol is also illustrated in this example.

Consider that the prime $p = 2^{256} - 189$ and the order is also prime.

PrimeOrd=115792089237316195423570985008687907853233080465625507841270369819257950283813.

The parameters of the elliptical curve are a=3, b=152961.

For security, the base point that satisfy the curve selection is

X=85315459259865965063509974234124764663467110109491361643876622847762669153457.

Y=94335063273662609244296467982193282358273992615431147138830104827552912957087.

The public key are chosen arbitrarily such that $n = 313310$. The private key $m = 65123$.

The public key point Q is (x_q, y_q) =(73393268584027436773872785641238579702902848999791359428266995512926506693853, 11066116735565104935616751518411243267897776117069219462237989463868571730943 0)

The secret point (x_s, y_s) is (91409296272786049566707611508001408792348697973922642695666 02

6700740249971702,514367772875879925028355464383655956354637723
152706815814231244819884923164 84)

The cipher message $C_m = (x_c, y_c)$ is generated by randomly choosing two messages $m_1 = 60, m_2 = 70$.

(42329582213301789094620395071752858437232599150252051885455153
672497905234011,
110196437743574170677877154162665510310945374340902254764336094
93887443321723)

The secret information Q and C_m will be hided into two polynomial of degree 5 thus increase the security level for the polynomial generation.

$f_1(x) = 330799x^5 + 297165x^4 + 476633x^3 +$
$212938x^2$ +7373393268584027436773872785641238579702902848999791
359428266995512926506693853x+42329582213301789094620395071752858
584372325991502520518854551536724979052340 11 $mod\ p$

$f_2(x) = 330799x^5 + 297156x^4 + 476693x^3 +$
$212938x^2$+1106611673556510493561675151841124326789777611706921946
22379894638685717309430x+11019643774357417067787715416266551031
094537434090225476433609493887443321723 $mod\ p$

The information of the two polynomials is secret for the dealer and no information to be leaked to the RFID tags. The RFID tags store only shares such that $f_1(2)$ through $f_1(n+1)$ and $f_2(2)$ through $f_2(n+1)$. For example if the value of n is three then the shared keys are $f_2(2)$
$= 7332403014404046721879498134554210998976831248419420670253156069$
0437808987330

$f_2(3)$ =309252094907517085690967819780927818394011768183450020913409721954512852770 22.

$f_2(4)$=104318478074779145342969567619331361542304025818136361519607967708378121455401. These shared values are reconstructed by the cipher-text reconstruction by applying reconstruction equations to get the secret shares.

The two parties first use TLS to set up a secure communication between each other using perfect security method such as using elliptical curve Diffie-Hellman key exchange and certificate based authentication.

By defining an elliptical curve such that $E = y^2 = X^3 - ax + b \; mod \; p$ with prime $p = \; 2^{256} - 189$

And a base point on the curve $P = (X, Y)$= (5,790519252745528958983862769151119894653220055301022690873976439313530659952 3).

The cipher-text reconstruction will choose a random nonce q for example *q*=35021 and send to the decryption server. The decryption server replies with a nonce *r* for example *r=12587.*

The cipher-text reconstruction calculates $[q.r]\,P$ which is equal to a point (x_d, y_d) on the curve such that (x_d, y_d)= (1736762262281770658592360446217375664705702826749205567386035722361604166371 3,101809059659448805054566935821852676591899879755228464945640089281310437091414)

The decryption server calculate $[r.q]\,P \;\; mod\, p$ which is equal to a point (x_d, y_d)on the curve such that

(x_d, y_d)=(1736762262281770658592360446217375664705702826749205567386035722361604166371 3,10180905965944880505456693582185267659189987975522846494564008928131043709141 4), then the initiation of TLS is achieved.

After successful initialising, the cipher-text reconstruction will hashed the value of the secret shared C_m and Q in order to send $p_1 = Sha3(k_{secret}, k_{hash})$ to the decryption server $p1 = Sha3(x_c, Sha3(x_c)$ =0x004D 0x008F 0x0028 0x00EA 0x0052 0x0011 0x00C6 0x006A 0x0028 0x00B6 0x00A0 0x00F9 0x008B 0x0034 0x00B4 0x005E 0x009C 0x0051 0x00CA 0x0067 0x0059 0x0019 0x003F 0x00F5 0x002D 0x0010 0x00EE 0x0057 0x009D 0x0064 0x0044 0x001D, 0x0026 0x0008 0x009C 0x00A7 0x00D1 0x0029 0x00A7 0x004E 0x00AA 0x009E 0x00BE 0x007E 0x00B4 0x0038 0x00BB 0x0037 0x0007 0x00E1 0x00A2 0x00C4 0x0051 0x0016 0x0001 0x00F1 0x005B 0x0014 0x00E9 0x00B4 0x0043 0x0083 0x0099 0x00D6).

$p_1 = Sha3(x_q, Sha3(x_q) =$ (0x0001 0x0000 0x0000 0x0000 0x0000 0x0000 0x0000 0x0000 0x0058 0x0097 0x00DA 0x0006 0x00AB 0x007F 0x0000 0x0000 0x0050 0x0094 0x00DA 0x0006 0x00AB 0x007F 0x0000 0x0000 0x0048 0x00A8 0x006D 0x0085 0x00FD 0x007F 0x0000 0x0000, 0x00F9 0x0091 0x00D4 0x003C 0x0028 0x004D 0x00C8 0x007D 0x0085 0x00A2 0x00B1 0x003F 0x00FF 0x0010 0x0070 0x00AE 0x0096 0x0099 0x0018 0x0037 0x008A 0x0023 0x00C1 0x0063 0x00D1 0x00F3 0x00DF 0x0037 0x00A5 0x00D3 0x00F2 0x00A9). And so on for the other secrets.

The decryption fetches p_1^* from secure data base then compare the value of p_1 and p_1^* . If the match process is correct then the sends a verification. The

decryption server will apply decryption equation in order to decrypt the logistic information about the package and find the values of logistic information which are $m_1 = 60$, $m_2 = 70$. The decryption server sends these information to the logistic information after successful TLS authentication and by computing $p_1 = Sha3(k_{secret}, k_{hash})$ where , the k_{secret} is the value of m_1 and m_2 respectively and k_{hash} is the hashed value of m_1 and m_2 respectively.

6.5.5 Worked example 2:

This example is an implementation of 256 bits of the anti-cloned quorum RFID based system which uses the Cramer-Shoup lite scheme with Shamir secret sharing scheme. The secrets are hidden into four polynomials in order to complete the system. The AES protocol is considered to be implemented by the DNA 413 tag so the implementation is not involved in this implementation.

Consider that the prime $p = 2^{256} - 189$ and the order is also prime.

PrimeOrd=115792089237316195423570985008687907853233080465625507841270369819257950283813.The parameters of the elliptical curve are a=3, b=152961. For security, the base point that satisfy the curve selection is X=85315459259865965063509974234124764663467110109491361643876622847762669153457 and Y=94335063273662609244296467982193282358273992615431147138830104827552912957087.

The dealer chooses two arbitary values $g_1 = 3$ and $g_2 = 5$ and four private keys such that $x = 111, y = 121, x_1 = 131, y_1 = 141$.

The public keys are computed as $cP = (x_{c1}, y_{c1})$= (871128166658576241816358194048896203043732987723386908415513768831405872194 36,182172530918024216083241943068358296264651903216797241937638545862707170565 64)

$hP = (x_{c2}, y_{c2}) =$

(437374748101855268502796542615861028922768035334531976648009307454046752939 42,910071516081717080969902874822679170490515725008221676082631782240865372869 16)

The secret points $S_1 = (x_{s1}, y_{s1})$ and $S_2 = (x_{s2}, y_{s2})$ are generated after randomly selecting $r = 15$ and choosing the message $m = 150$.

$e = C^r \ mod \ p =$

795372404377098415172665749741992527915300445280770490657983931955311834795 42

$(x_{s1}, y_{s1}) =$

151078244002758690428349426745989180655509158792466402906032505426243244517 12,804445018949792034184151264383585630424866881860379674525573334028680442209 01).

(x_{s2}, y_{s2}) =(101809088752957308593671667622966993798430077589582396593271791787475146463820,876394368063002436058081528497882911381037646834276886428816835969691332962 79)

The secret information $(x_{c1}, y_{c1}), (x_{c2}, y_{c2}), (x_{s1}, y_{s1})$ and $x_{s2}, y_{s2})$ will be hided into four polynomials of degree 5.

$f_1(x)$ =988053478x^5+374403366x^4+810266434x^3+11332952x^2+151078244002758690428349426745989180655509158792466402906032505426243245171282

451712x+87112816665857624181635819404889620304373298772338690841551376883140587219436.

$f_2(x)$=988053478x^5+374403366x^4+810266434x^3+11332952x^2+151078244002758690428349426745989180655509158792466402906032505426243244517 12x+18217253091802421608324194306835829626465190321679724193763854586270717056564.

$f_3(x)$ =988053478x^5+374403366x^4+810266434x^3+11332952x^2+101809088752957308593671667622966993798430077589582396593271791787475146463820x+43737474810185526850279654261586102892276803533453197664800930745404675293942.

$f_4(x)$ =988053478x^5+374403366x^4+810266434x^3+11332952x^2+876394368063002436058081528497882911381037646834276886428816835969691332962 79x+91007151608171708096990287482267917049051572500822167608263178224086537286916.

The information of the two polynomials is secret for the dealer and no information to be leaked to the RFID tags. The RFID tags store only shares such that are $f_1(2)$ through $f_1(n+1)$ and $f_2(2)$ through$f_2(n+1)$. The hashed value of each shared keys are stored as a master key in each RFID tag to complete the authentication process.

Theses shared values are reconstructed by the cipher-text reconstruction by applying equations to get the secret shares.

Similar to worked example 1, the cipher-text reconstruction will authenticate with the decryption server in order to send the secret information for the decryption process. the authentication process

The decryption server verified if the $[e]P = [u^{x1}v^{y1}]P$= 795372404377098415172665749741992527915300445280770490657983931955311834795 42, then computes $m = \frac{w}{u^x v^y} \ mod \ p$.

6.5.6 Timing Analysis

The timing performance for the quorum RFID based system generating secret shares is shown in figure 6.5. The performance is evaluated by applying the generation of secret shares procedure on 5th - 25th polynomials degree amongst 10-100 RFID tags. Additionally, increasing the polynomials degrees leads to decrease in the security threats to the system. In the other word, the increase of polynomials degrees supports the strength of the system against such RFID threat through the difficulties of guessing how the shares are generated. Figure 6.6 presents the timing performance for entity system including the secret shares procedure, cipher-text reconstruction, decryption process and the authentication phase.

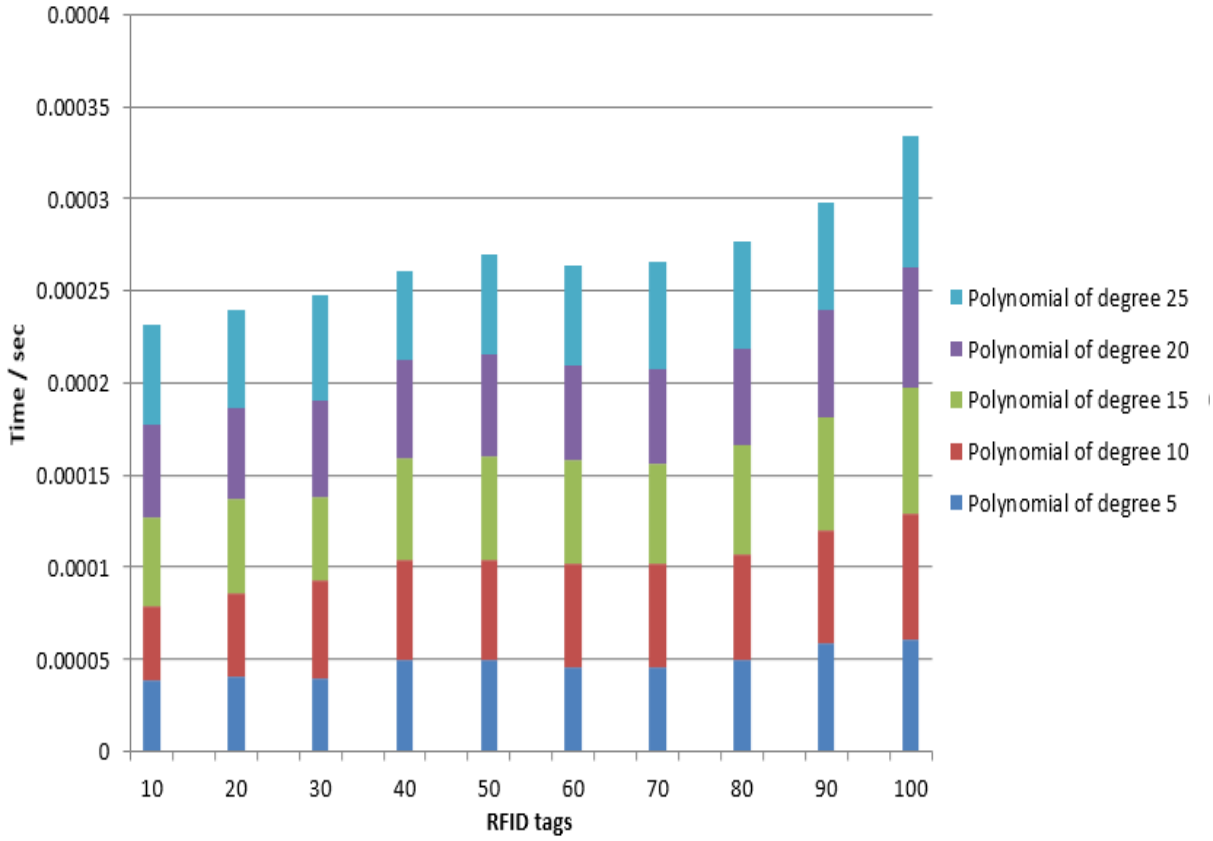

Figure 6. 6:Timing analysis for generating secret shares of the quorum RFID based system

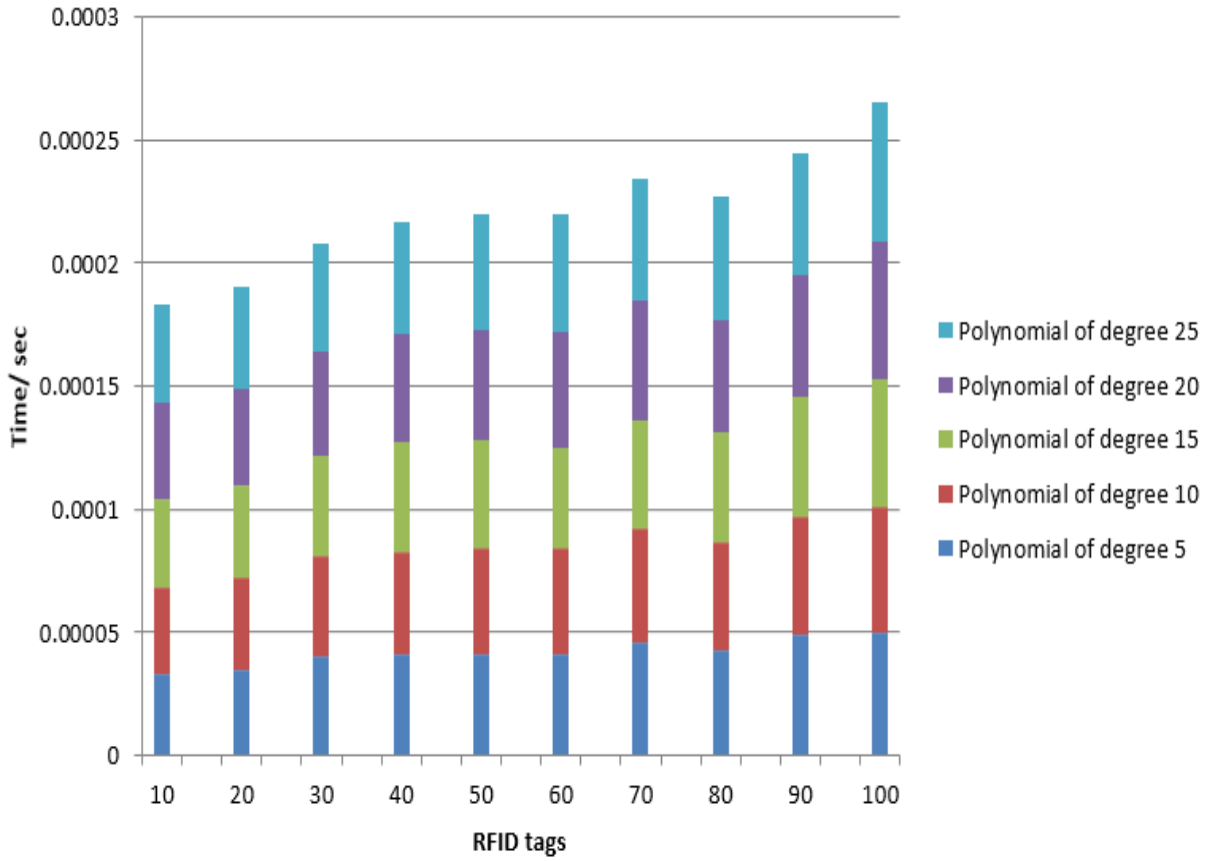

Figure 6. 7: Timing analysis for the quorum RFID based system

The timing performance for the anti-cloned RFID quorum system is presented in Figure 6.7 and figure 6.8. Figure 6.7 shows the key generation performance for generating and applying secret shares procedure on 5^{th} - 25^{th} polynomials degree amongst 10-100 RFID tags. Figure 6.8 presents the timing performance for the entity system.

It's clear that the firsts system requires less computation which leads to lower timing performance. However, both systems have achieved the requirements for the low -cost RFID tags implementation

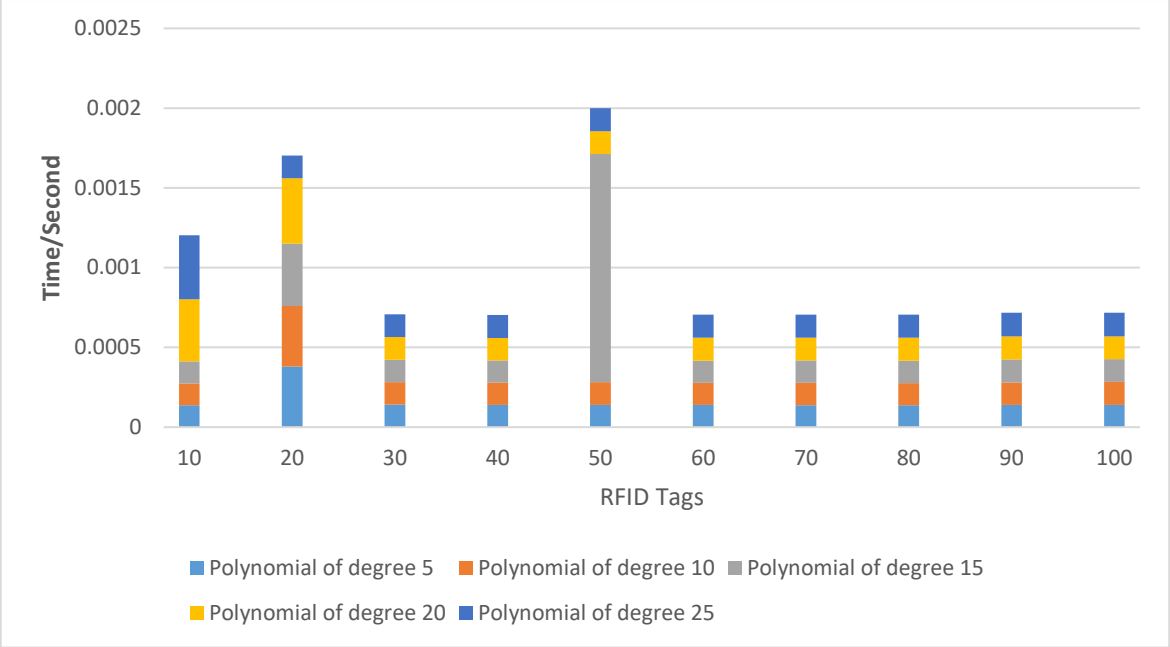

Figure 6. 8: timing analysis for generating secret shares of the anti-cloned quorum RFID based system

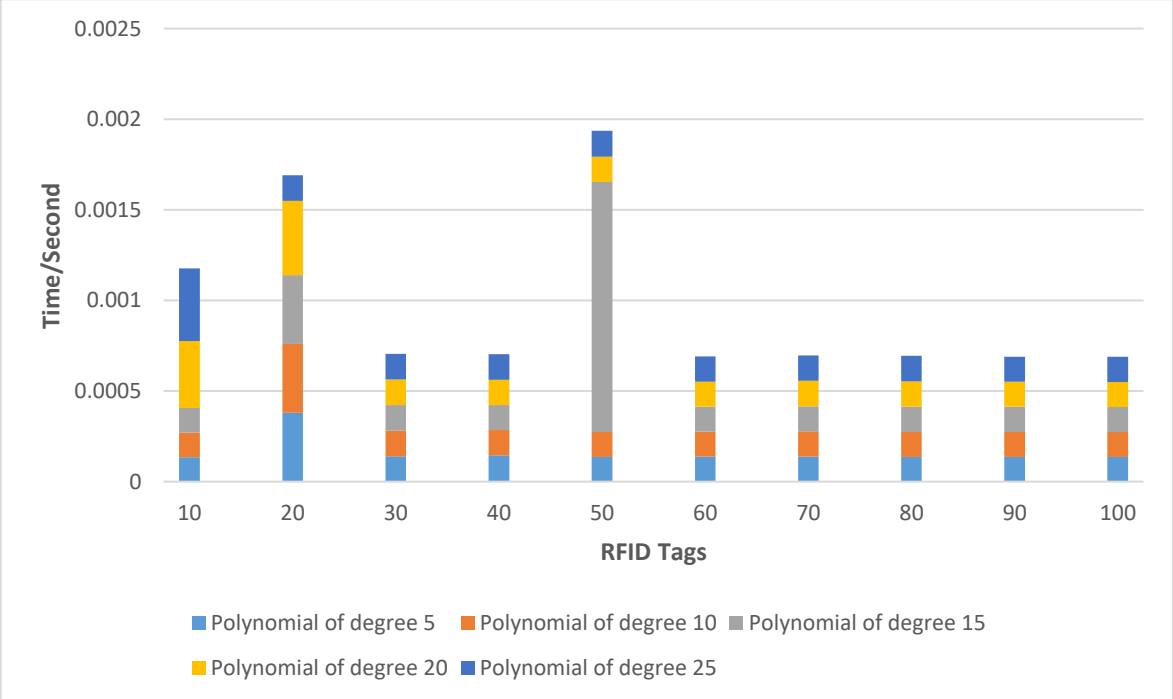

Figure 6. 9: Timing analysis for the anti-cloned quorum RFID based system

6.6 Summery

In this chapter, two RFID systems have been introduced in term of supporting a package of multi RFID tags to be securely authenticated with one reader. Both protocols are designed to be used for the passive RFID tags.

This is done by allowing the logistic information to be securely distributed secret key among multiple tags such that not all tags have to be utilized in a threshold cryptosystem. The first system is based on using elliptical curve version of ElGamal threshold cryptosystem. Each RFID tag has a shared key that is generated from a polynomial that contains the encrypted secret information. A reader is used to send tags shared key information to the cipher-text reconstruction for verification stage while the final verification is to check the original secret information after authentication and decryption process. The security of the system relies on the security of the elliptical curve version of ElGamal cryptosystem and Shamir secret sharing scheme. As a result, this system provides security and privacy for the RFID system by using distributed Shamir cipher-text secret amongst several RFID tags. Thus, an adversary cannot reveal the secret key or learn any information from the RFID tags. However, the system is vulnerable to cloning attack by an adversary.

The second system is based on using the elliptical curve version of the lite scheme of Cramer-Shoup with Shamir secret sharing for distributing n RFID tags. To overcome the cloning attack in the first system, a mutual authentication scheme between RFID tags and a reader is achieved by using AES with Keccak hash function. Thus, an adversary cannot clone tags and cannot reach the privacy property for the RFID tags.

The second system can be also implemented by using the elliptical cuve version of the full version Cramer-Shoup scheme as it achieves the IND-CCA2 property. The full version involves of using a one –way hash function in the key generation process and the encryption process. A particular addition to the Cramer – Shoup cryptosystem can be achieved by using the Keccak hash function. This will result in ensuring further security and privacy for the system.

7. Conclusions and Future Work

7.1 Conclusions

The aim of this thesis is to investigate the security and privacy concerns of RFID system. These concerns come from the fact that wireless channel increases the demand for possible threats to affect the security and privacy of the RFID system. Although traditional cryptographic solutions can improve the immunity against most of these threats, they are still impractical due to computation limitation of passive RFID tags. The problem of satisfying security and privacy properties for RFID system and minimising the computation in RFID tags at minimum computation has increased research for developing several RFID authentication protocols. However, several RFID authentication protocols still lack provision of enough requirements for security and privacy of RFID system. Designing secure and private authentication protocol that supports low computation ability of RFID tags is considered to be difficult to achieve. Combining high computation cryptographic protocols with RFID system leads to affecting the scalability of the system or even cost effectiveness of the tags. Therefore, this thesis focuses on implementing RFID protocols that can enhance security and privacy levels for RFID system. This is achieved by classifying the ability of an adversary to threaten the security and privacy level of the system. Then, design a mutual authentication protocol that can ensure good security and privacy levels and can be applied to single RFID tag to reader authentication. Additionally, this thesis also considers designing a mutual authentication protocol that can support multi-RFID tags to be authenticated. Furthermore, the thesis targets are to meet the following investigations:

The first investigation is how to design an efficient mutual authentication protocol that can be used to overcome the security and privacy threats in related work.

The authentication protocol is based on using a zero-knowledge proof technique as a challenge-response authentication. Thus, the tag's secret cannot be revealed until the final verification process is completed. The protocol combines using Schnorr identification protocol that can run over the elliptical curve of prime numbers and Keccak hash function. The confidentiality of the exchanged message is ensured by using the zero-knowledge protocol and ensuring the integrity of the message is achieved by using Keccak hash function. This combination assures that the mutual authentication is run through legitimate parties that need to complete the challenge otherwise, the authentication fails to process. The authentication process requires each party to compute the challenge in each step then the final stage requires the verifier to compute the final challenge and then get the secret information about the tag. Although, the computations are run through RFID tag is considered to be a high computation, the security and privacy achievement is promising. Therefore, this chapter discusses how to apply secure mutual authentication process and can be done with passive RFID tags but with computations ability.

The possible security threats were discussed and showed that only legitimate tags could participate in the mutual authentication phase. The privacy threats are also discussed and it can be proved that an adversary cannot affect the privacy of tag by tracing tag or accessing future transactions. The security of the protocol relies on using Keccak hash function in random oracle to validate the overall model design of the protocol.

The second investigation is to design and implement a low-cost RFID system that supports multi-RFID tags. In chapter six, two RFID protocols were introduced. The aim of these protocols is to consider the scenario of including multi-RFID tags in a package, and during the transmission, there is a possibility of losing one or more tags. Therefore, Shamir secret sharing scheme is used to ensure there is no problem to complete the verification process if one or more tags are damaged or lost. The idea behind these systems is to store shared keys that are generated from the secret information in each RFID tag instead of using tag's secret in the authentication process. Both protocols establish a combination of encrypting process and distributing process of RFID tags that can be used in supply chain management. These protocols rely on using servers to generate shares, reconstruct shares and then decrypt the secret information. The first protocol is an efficient and compact protocol that can be implemented in storage only tags that do not perform any computation. The secret information is encrypted by using ElGamal cryptosystem with elliptical curve operations. The distribution process is done by using Shamir secret sharing scheme that hides these secrets into two polynomial and generated shared keys that can be stored in each tag. The way of distributing shared keys are varied as they are generated through encrypted procedures and depending on the complexity of the polynomials. Tags only require storing encrypted shared keys; therefore, a reader will only send tag shared keys to the cipher-text reconstruction for verifying the shared keys reconstructing them to the original shares without revealing any secret information. The final stage is to decrypt the shares and return them to the original message after specific decryption process. In addition, a specific TLS authentication is illustrated by

using Keccak hash function to verify and ensure the integrity of the message exchanged between servers and to provide forward privacy.

The combination of the system ensures security and privacy, and only legitimate parties can participate in the system. However, since the RFID tags cannot perform any computation, there is a possible vulnerability of the tag to be cloned. Therefore, the second system is introduced as the anti-cloned quorum RFID based system.

The same procedure is followed in the second system, but additionally, it includes a mutual authentication phase that performs computations on each RFID tag. The mutual authentication phase involves a combination of AES encryption and Keccak hash function in order to satisfy security against the cloning attack. This procedure can be applied to NTAG 413 DNA tag which uses AES primitive as a basic authentication scheme. Moreover, the secret information is encrypted by using the elliptical curve version of the lite Cramer-Shoup scheme to ensure IND-CCA property for the system. The authentication phase has two rounds of verification in order to send the shared keys to a reader. A reader is required to complete the authentication process by using the authentication keys then verify the validity of the tags and send tag's shared key to the decryption server. The security of the system relies on the fact that only legitimate tags and reader can complete the authentication process. Therefore, the system considers most of security and privacy threats that can affect an RFID system and designed to enhance the immunity against these threats. Possible threats have been analysed and timing performance has been achieved by implementing 256 bits of both protocols.

In summary, both protocols combine different cryptographic techniques and can be used in low-cost RFID tags.

7.2 Future works

Possible research directions can be investigated in further research are summarised as follows:

- Investigate possible quantum cryptography in RFID system by implementing 512 bits of elliptic curve authentication protocol.
- Investigate the possibility of using a lattice-based authentication in low-cost RFID tags.
- Investigate the possibility to design an efficient distance bounding protocol.
- Investigate the possibility of designing an authentication protocol that is based on error correction code by using McEleese cryptosystem.
- Design an RFID system that is based on using multi-party computation and can support multi-RFID tags combine with active RFID tags.

References

1. Abughazalah, S., Markantonakis, K. and Mayes, K., 2014, May. Enhancing the key distribution model in the RFID-enabled supply chains. In *Advanced Information Networking and Applications Workshops (WAINA), 2014 28th International Conference on* (pp. 871-878). IEEE.
2. Adams. 2014. *Adams1*. [ONLINE] Available at: http://www.adams1.com/stack.html. [Accessed 1 December 17]
3. Ahson, S. A., & Ilyas, M. (2008). *RFID handbook: applications, technology, security, and privacy*. CRC press.
4. Al Kattan, I., & Al-Khudairi, T. (2007, July). Improving supply chain management effectiveness using RFID. In *Engineering Management Conference, 2007 IEEE International* (pp. 191-198). IEEE.
5. Alien technology. 2014. *ALR-9680*. [ONLINE] Available at: http://www.alientechnology.com/products/readers/commercial-4-port/. [Accessed 1 December 17].
6. Avoine, G., & Oechslin, P. (2005, March). A scalable and provably secure hash-based RFID protocol. In *Pervasive Computing and Communications Workshops, 2005. PerCom 2005 Workshops. Third IEEE International Conference on* (pp. 110-114). IEEE.
7. Avoine, G., 2005. Adversarial Model for Radio Frequency Identification. *IACR Cryptology ePrint Archive*, *2005*, p.49.
8. Babaheidarian, P., Delavar, M., & Mohajeri, J. (2012, September). On the security of an ECC based RFID authentication protocol. In

Information Security and Cryptology (ISCISC), 2012 9th International ISC Conference on (pp. 111-114). IEEE.

9. Bagchi, U., Guiffrida, A., O'Neill, L., Zeng, A., & Hayya, J. (2007). The effect of RFID on inventory management and control. In *TRends in supply chain design and management* (pp. 71-92). Springer London.
10. Banks J., David Hanny, Manuel A. Pachano, and Les G (2007). Thompson, *RFID APPLIED.* NJ: John Wiley & Sons, Inc.
11. Batina, L., Guajardo, J., Kerins, T., Mentens, N., Tuyls, P., & Verbauwhede, I. (2007, March). Public-key cryptography for RFID-tags. In *Pervasive Computing and Communications Workshops, 2007. PerCom Workshops' 07. Fifth Annual IEEE International Conference on* (pp. 217-222). IEEE.
12. Batina, L., Lee, Y.K., Seys, S., Singelée, D. and Verbauwhede, I., 2011, October. Privacy-Preserving ECC-Based Grouping Proofs for RFID. In *ISC* (pp. 159-165).
13. Bellare, M., Desai, A., Pointcheval, D. and Rogaway, P., 1998. Relations among notions of security for public-key encryption schemes. In *Advances in Cryptology—CRYPTO'98* (pp. 26-45). Springer Berlin/Heidelberg.
14. Bertoni, G., Daemen, J., Peeters, M. and Van Assche, G., 2009. Keccak sponge function family main document. *Submission to NIST (Round 2)*, *3*, p.30.
15. Bhuptani, M., & Moradpour, S. (2005). *RFID field guide: deploying radio frequency identification systems*. Prentice Hall PTR.
16. Blahut, R. E. (2014). *Cryptography and Secure Communication*. Cambridge University Press.

17. Blake, I., Seroussi, G. and Smart, N., 1999. *Elliptic curves in cryptography* (Vol. 265). Cambridge university press.
18. Blakley, G.R., 1979, June. Safeguarding cryptographic keys. In *Proceedings of the national computer conference* (Vol. 48, pp. 313-317).
19. Bolic, M., Simplot-Ryl, D. and Stojmenovic, I. eds., 2010. *RFID systems: research trends and challenges*. John Wiley & Sons.
20. Boneh, D. and Franklin, M., 2001. Identity-based encryption from the Weil pairing. In *Advances in Cryptology—CRYPTO 2001* (pp. 213-229). Springer Berlin/Heidelberg.
21. Bos, J.W., Costello, C., Longa, P. and Naehrig, M., 2016. Selecting elliptic curves for cryptography: an efficiency and security analysis. *Journal of Cryptographic Engineering*, *6*(4), pp.259-286.
22. Bringer, J., Chabanne, H., & Icart, T. (2008). Cryptanalysis of EC-RAC, a RFID identification protocol. In *Cryptology and Network Security* (pp. 149-161). Springer Berlin Heidelberg.
23. Burmester, M., De Medeiros, B., & Motta, R. (2008). Provably secure grouping-proofs for RFID tags. In *Smart Card Research and Advanced Applications* (pp. 176-190). Springer Berlin Heidelberg.
24. Cai, S., Li, T., Ma, C., Li, Y. and Deng, R.H., 2009, December. Enabling secure secret updating for unidirectional key distribution in RFID-enabled supply chains. In *International Conference on Information and Communications Security* (pp. 150-164). Springer, Berlin, Heidelberg.
25. Cai, S., Li, Y., Li, T., & Deng, R. H. (2009, March). Attacks and improvements to an RIFD mutual authentication protocol and its extensions. In *Proceedings of the second ACM conference on Wireless network security* (pp. 51-58). ACM.

26. Canard, S., Coisel, I., Etrog, J. and Girault, M., 2010. Privacy-Preserving RFID Systems: Model and Constructions. *IACR Cryptology ePrint Archive*, *2010*, p.405.
27. Chen, Y., Chou, J.S., Lin, C.F. and Wu, C.L., 2011. A Novel RFID Authentication Protocol based on Elliptic Curve Cryptosystem. *IACR Cryptology ePrint Archive*, *2011*, p.381.
28. Cheng, S., Varadharajan, V., Mu, Y. and Susilo, W., 2017. An efficient and provably secure RFID grouping proof protocol.
29. Chia-Hui Wei, Min-Shiang Hwang, and Augustin Yeh-hao Chin. "A Mutual Authentication Protocol for RFID". *IT Professional*, pp. 20-24, 2011.
30. Chien, H. Y., & Chen, C. H. (2007). Mutual authentication protocol for RFID conforming to EPC Class 1 Generation 2 standards. *Computer Standards & Interfaces*, *29*(2), 254-259.
31. Chien, H. Y., & Liu, S. B. (2009, April). Tree-based RFID yoking proof. In *Networks Security, Wireless Communications and Trusted Computing, 2009. NSWCTC'09. International Conference on* (Vol. 1, pp. 550-553). IEEE.
32. Chopra, S., & Meindl, P. (2007). *Supply chain management. Strategy, planning & operation* (pp. 265-275). Gabler.
33. Chou, J.S., 2014. An efficient mutual authentication RFID scheme based on elliptic curve cryptography. *The Journal of Supercomputing*, *70*(1), pp.75-94.
34. Cisco. 2014. *RFID tag consideration*. [ONLINE] Available at: http://www.cisco.com/c/en/us/td/docs/solutions/Enterprise/Mobility/WiFiLBS-DG/wifich6.html. [Accessed 1 December 17].

35. Cramer, R. and Shoup, V., 1998. A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. In *Advances in Cryptology—CRYPTO'98* (pp. 13-25). Springer Berlin/Heidelberg.
36. Diffie, W. and Hellman, M., 1976. New directions in cryptography. *IEEE transactions on Information Theory*, *22*(6), pp.644-654.
37. Dimitriou, T. (2005, September). A lightweight RFID protocol to protect against traceability and cloning attacks. In *Security and Privacy for Emerging Areas in Communications Networks, 2005. SecureComm 2005. First International Conference on* (pp. 59-66). IEEE.
38. Dimitriou, T. (2005, September). A lightweight RFID protocol to protect against traceability and cloning attacks. In *Security and Privacy for Emerging Areas in Communications Networks, 2005. SecureComm 2005. First International Conference on* (pp. 59-66). IEEE.
39. ElGamal, T., 1985. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE transactions on information theory*, *31*(4), pp.469-472.
40. EPC-RFID. 2014. *Epc-RFID*. [ONLINE] Available at: http://www.epc-rfid.info/rfid. [Accessed 1 December 17]
41. Evsen Korkmaz and Alp Ustundag, "Standards, Security & Privacy Issues about Radio Frequency Identification (RFID)" *RFID Eurasia 1st Annual,* pp. 1-10, Sept. 2007.
42. Fan, J., Hermans, J., & Vercauteren, F. (2010). On the claimed privacy of EC-RAC III. In *Radio Frequency Identification: Security and Privacy Issues* (pp. 66-74). Springer Berlin Heidelberg.
43. Ferguson, N., & Schneier, B. (2003). *Practical cryptography* (Vol. 23). New York: Wiley.

44. FINKENZELLER, K. 2010. *RFID handbook, Fundemental and applications in contactless smart cards, radio frequency Identification and near feald communication.* John Wiley & Sons Ltd.
45. Garcia-Alfaro, J., Barbeau, M., & Kranakis, E. (2008, April). Security threats on EPC based RFID systems. In *Fifth international conference on information technology: new generations* (pp. 1242-1244). IEEE.
46. Garfinkel S., Rosenberg B. (2006) RFID: Applications, Security, and Privacy, Addison-Wesley, Reading, MA.
47. GARFINKEL, S. L., JUELS, A., & PAPPU, R. (2005). RFID privacy: An overview of problems and proposed solutions. *IEEE security & privacy*, *3*(3), 34-43.
48. Giusto, D., Iera, A., Morabito, G., & Atzori, L. (Eds.). (2010). *The internet of things: 20th Tyrrhenian workshop on digital communications*. Springer Science & Business Media.
49. GMP. (2017). *The GNU Multiple Precision Arithmetic Library .* Available: https://gmplib.org/. Last accessed 30/10/2017.
50. Goldreich, O. (2004). *Foundations of cryptography: volume 2, basic applications*. Cambridge university press.
51. Goldwasser, S., Micali, S., & Rackoff, C. (1989). The Knowledge Complexity of Interactive Proof Systems. *SIAM Journal on Computing*, *18*(1), 186-208.
52. Hankerson, D., Menezes, A.J. and Vanstone, S., 2006. *Guide to elliptic curve cryptography*. Springer Science & Business Media.
53. He, D. and Zeadally, S., 2015. An analysis of rfid authentication schemes for internet of things in healthcare environment using elliptic curve cryptography. *IEEE internet of things journal*, *2*(1), pp.72-83.

54. Henrici, D. (2008). *RFID security and privacy: concepts, protocols, and architectures* (Vol. 17). Springer Science & Business Media.

55. Henrici, D., & Müller, P. (2004, March). Hash-based enhancement of location privacy for radio-frequency identification devices using varying identifiers. In *Pervasive Computing and Communications Workshops, 2004. Proceedings of the Second IEEE Annual Conference on* (pp. 149-153). IEEE.

56. Hermans, J. and Peeters, R., 2012, July. Private Yoking Proofs: Attacks, Models and New Provable Constructions. In *RFIDSec* (pp. 96-108).

57. Hermans, J., Peeters, R. and Preneel, B., 2014. Proper RFID privacy: model and protocols. *IEEE Transactions on Mobile Computing*, *13*(12), pp.2888-2902.

58. Hong-yan, K., 2015. Design of a Mutual Authentication Protocol for RFID Based on ECC. *Open Automation and Control Systems Journal*, *7*, pp.1532-1536.

59. Huang, H. H., & Ku, C. Y. (2009). A RFID grouping proof protocol for medication safety of inpatient. *Journal of medical systems*, *33*(6), 467-474.

60. Huang, H. H., & Ku, C. Y. (2009). A RFID grouping proof protocol for medication safety of inpatient. *Journal of medical systems*, *33*(6), 467-474.

61. Huang, P. and Mu, H., 2015. A High-security RFID Grouping Proof Protocol. *International Journal of Security and Its Applications*, *9*(1), pp.35-44.

62. Hunt, V. D., Puglia, A., & Puglia, M. (2007). *RFID: a guide to radio frequency identification*. John Wiley & Sons.

63. Hutto, J., & Atkinson, R. D. (2004). Radio Frequency Identification.International Organization for Standardisation, Genève, Switzerland. ISO7498-2: 1989, Information processing systems | Open systems Interconnection | Basic reference model | Part 2: Security architecture, 1989.
64. Jesic-tech. 2014. *RFID tag*. [ONLINE] Available at: http://www.jesic-tech.com/RFID_tag.html. [Accessed 1 December 17].
65. Juels, A. (2004, March). "Yoking-proofs" for RFID tags. In *Pervasive Computing and Communications Workshops, 2004. Proceedings of the Second IEEE Annual Conference on* (pp. 138-143). IEEE.
66. Juels, A. (2006). RFID security and privacy: A research survey. *Selected Areas in Communications, IEEE Journal on*, *24*(2), 381-394.
67. Juels, A., Pappu, R. and Parno, B., 2008, July. Unidirectional Key Distribution Across Time and Space with Applications to RFID Security. In *USENIX Security Symposium* (pp. 75-90).
68. Kang, Y., & Gershwin, S. B. (2005). Information inaccuracy in inventory systems: stock loss and stockout. *IIE transactions*, *37*(9), 843-859.
69. Knorr, K., & Röhrig, S. (2001). Security requirements of e-business processes. In *Towards the E-Society* (pp. 72-86). Springer US.
70. Ko, W.T., Chiou, S.Y., Lu, E.H. and Chang, H.K.C., 2014. Modifying the ECC-based grouping-proof RFID system to increase inpatient medication safety. *Journal of medical systems*, *38*(9), p.66.
71. Korsh, J. F., & Garrett, L. J. (1989). *Data structures, algorithms, and program style using C*. PMS/King Publishing Co.
72. Langheinrich, M. (2009). A survey of RFID privacy approaches. *Personal and Ubiquitous Computing*, *13*(6), 413-421.

73. Langheinrich, M. and Marti, R., 2007. Practical minimalist cryptography for RFID privacy. *IEEE Systems Journal*, *1*(2), pp.115-128.

74. Langheinrich, M. and Marti, R., 2007. RFID privacy using spatially distributed shared secrets. *Ubiquitous Computing Systems*, pp.1-16.

75. Layton, T. P. (2006). *Information Security: Design, Implementation, Measurement, and Compliance*. CRC Press.

76. Lee, H. C., & Yi, J. H. (2011, September). Development of privacy-preserving RFID authentication system using mobile devices. In *ICT Convergence (ICTC), 2011 International Conference on* (pp. 760-765). IEEE.

77. Lee, Y. K., Batina, L., & Verbauwhede, I. (2008, April). EC-RAC (ECDLP based randomized access control): Provably secure RFID authentication protocol. In *RFID, 2008 IEEE International Conference on* (pp. 97-104). IEEE.

78. Lee, Y. K., Batina, L., Singelée, D., & Verbauwhede, I. (2010, March). Low-cost untraceable authentication protocols for RFID. In *Proceedings of the third ACM conference on Wireless network security* (pp. 55-64). ACM.

79. Lee, Y. K., Sakiyama, K., Batina, L., & Verbauwhede, I. (2008). Elliptic-curve-based security processor for RFID. *Computers, IEEE Transactions on*, *57*(11), 1514-1527.

80. Lim, C. H., & Kwon, T. (2006). Strong and robust RFID authentication enabling perfect ownership transfer. In *Information and Communications Security* (pp. 1-20). Springer Berlin Heidelberg.

81. Lin, C. C., Lai, Y. C., Tygar, J. D., Yang, C. K., & Chiang, C. L. (2007). Coexistence proof using chain of timestamps for multiple RFID tags. In

Advances in Web and Network Technologies, and Information Management (pp. 634-643). Springer Berlin Heidelberg.

82. Lin, Q. and Zhang, F., 2012. ECC-based grouping-proof RFID for inpatient medication safety. *Journal of Medical Systems*, *36*(6), pp.3527-3531.

83. Liu, L., Chen, Z., Yan, D., Lu, Y., & Wang, H. (2010, May). RFID in Supply Chain Management. In *E-Business and E-Government (ICEE), 2010 International Conference on* (pp. 3279-3282). IEEE.

84. Liu, Y.L., Qin, X.L., Wang, C. and Li, B.H., 2013. A lightweight RFID authentication protocol based on elliptic curve cryptography. *Journal of computers*, *8*(11).

85. Lv, C., Jia, X., Lin, J., Jing, J. and Tian, L., 2011, May. An efficient group-based secret sharing scheme. In *International Conference on Information Security Practice and Experience* (pp. 288-301). Springer, Berlin, Heidelberg.

86. Lv, C., Jia, X., Lin, J., Jing, J., Tian, L. and Sun, M., 2011. Efficient secret sharing schemes. *Secure and Trust Computing, Data Management and Applications, Communications in Computer and Information Science*, *186*, pp.114-121.

87. Lv, C., Li, H., Ma, J., Niu, B. and Jiang, H., 2011. Security analysis of a privacy-preserving ECC-based grouping-proof protocol. *Journal of Convergence Information Technology*, *6*(3), pp.113-119.

88. Lv, C., Li, H., Ma, J., Niu, B., & Jiang, H. (2011). Security Analysis of a Privacy-preserving ECC-based Grouping-proof Protocol. *Journal of Convergence Information Technology*, *6*(3), 113-119.

89. Malek, B., & Miri, A. (2012, June). Lightweight mutual RFID authentication. In *Communications (ICC), 2012 IEEE International Conference on* (pp. 868-872). IEEE.
90. Martínez, S., Valls, M., Roig, C., Miret, J. M., & Giné, F. (2009). A secure elliptic curve-based RFID protocol. *Journal of Computer Science and Technology*, *24*(2), 309-318.
91. Masum, K. M., & Bhuiyan, F. (2013). Impact of Radio Frequency Identification (RFID) Technology on Supply Chain Efficiency: An Extensive Study. *Global Journal of Researches In Engineering*, *13*(4).
92. Mathur, S. N (2010). *Inventory management.*Saurashta University.
93. McLoone, M., & Robshaw, M. J. (2006). Public key cryptography and RFID tags. In *Topics in Cryptology–CT-RSA 2007* (pp. 372-384). Springer Berlin Heidelberg.
94. Menezes, A. J., Van Oorschot, P. C., & Vanstone, S. A. (1996). *Handbook of applied cryptography*. CRC press.
95. Michael, K., & McCathie, L. (2005, July). The pros and cons of RFID in supply chain management. In *Mobile Business, 2005. ICMB 2005. International Conference on* (pp. 623-629). IEEE.
96. Nájera, P., & Lopez, J. (2008). RFID: Technological issues and privacy concerns. *Digital Privacy: Theory, Technologies and Practices*, 285-306.
97. Nance, D. W., & Naps, T. L. (1992). *Introduction to Computer Science: Programming, Problem Solving and Data Structures*. West Publishing Co..
98. NXP. (2017). *NTAG 413 DNA .* Available: https://www.nxp.com/products/identification-and-security/smart-label-

and-tag-ics/ntag/ntag-413-dna-secure-unique-nfc-message-for-direct-access-to-web-services:NT4H1321G0DUF. Last accessed 30/10/2017.

99. Ohkubo, M., Suzuki, K., & Kinoshita, S. (2003, November). Cryptographic approach to "privacy-friendly" tags. In *RFID privacy workshop* (Vol. 82).

100. Oppliger, R., 2016. *SSL and TLS: Theory and Practice*. Artech House.

101. Osaka, K., Takagi, T., Yamazaki, K., & Takahashi, O. (2006, November). An efficient and secure RFID security method with ownership transfer. In *Computational Intelligence and Security, 2006 International Conference on* (Vol. 2, pp. 1090-1095). IEEE.

102. Ouafi, K., & Phan, R. C. W. (2008, January). Traceable privacy of recent provably-secure RFID protocols. In *Applied Cryptography and Network Security* (pp. 479-489). Springer Berlin Heidelberg.

103. Paise, R.I. and Vaudenay, S., 2008, March. Mutuaauthentication in RFID: security and privacy. In *Proceedings of the 2008 ACM symposium on Information, computer and communications security* (pp. 292-299). ACM.

104. Peris-Lopez, P., Hernandez-Castro, J. C., Estevez-Tapiador, J. M., & Ribagorda, A. (2006, January). RFID systems: A survey on security threats and proposed solutions. In *Personal Wireless Communications* (pp. 159-170). Springer Berlin Heidelberg.

105. Peris-Lopez, P., Hernandez-Castro, J. C., Estevez-Tapiador, J. M., & Ribagorda, A. (2007, July). Solving the simultaneous scanning problem anonymously: clumping proofs for RFID tags. In *Security,*

Privacy and Trust in Pervasive and Ubiquitous Computing, 2007. SECPerU 2007. Third International Workshop on (pp. 55-60). IEEE.

106. Peris-Lopez, P., Orfila, A., Hernandez-Castro, J. C., & Van der Lubbe, J. C. (2011). Flaws on RFID grouping-proofs. Guidelines for future sound protocols. *Journal of Network and Computer Applications*, *34*(3), 833-845. Pervasive Computing | SPC 2003, volume 2802 of Lecture Notes in Computer.

107. Piramuthu, S. (2006, June). On existence proofs for multiple RFID tags. In *Pervasive Services, 2006 ACS/IEEE International Conference on* (pp. 317-320). IEEE.

108. Rekik, Y., Sahin, E., & Dallery, Y. (2009). Inventory inaccuracy in retail stores due to theft: An analysis of the benefits of RFID. *International Journal of Production Economics*, *118*(1), 189-198.

109. Rieback, M. R., Crispo, B., & Tanenbaum, A. S. (2006). The evolution of RFID security. *IEEE Pervasive Computing*, (1), 62-69.

110. Ryu, E.K., Kim, D.S. and Yoo, K.Y., 2015, June. On elliptic curve based untraceable RFID authentication protocols. In *Proceedings of the 3rd ACM Workshop on Information Hiding and Multimedia Security* (pp. 147-153). ACM.

111. Saito, J., & Sakurai, K. (2005, March). Grouping proof for RFID tags. In *Advanced Information Networking and Applications, 2005. AINA 2005. 19th International Conference on* (Vol. 2, pp. 621-624). IEEE.

112. Sandhya, M. and Rangaswamy, T. R. “A Forward Secured Authentication Protocol For Mobile”, *International Journal of Information Technology and Knowledge Management vol. 4, no. 2,* pp. 549-553. 2011

113. Sato, T., Toyoda, K. and Sasase, I., 2016, August. Practical key distribution scheme with less dummy tags in RFID-enabled supply chains. In *Communications (APCC), 2016 22nd Asia-Pacific Conference on* (pp. 476-480). IEEE.

114. Saygin, C. (2007). Adaptive inventory management using RFID data. *The International Journal of Advanced Manufacturing Technology*, *32*(9-10), 1045-1051.

115. Schnorr, C. (1990). Efficient Identification and Signatures for Smart Cards. In *Advances in Cryptology—CRYPTO'89 Proceedings* (pp. 239-252). Springer Berlin/Heidelberg.

116. Shamir, A., 1979. How to share a secret. *Communications of the ACM*, *22*(11), pp.612-613.

117. Shanmugam, R., 2001. Elliptic Curves and Their Applications to Cryptography: An Introduction: Andreas Enge, Kluwer Academic Press, Norwell, MA, 1999, pp. 164. ISBN 0-7923-8589-6.

118. Shen, J., Tan, H., Ren, Y., Liu, Q. and Wang, B., 2016, January. A practical RFID grouping authentication protocol in multiple-tag arrangement with adequate security assurance. In *Advanced Communication Technology (ICACT), 2016 18th International Conference on* (pp. 693-699). IEEE.

119. Shen, J., Tan, H., Wang, Y., Ji, S. and Wang, J., 2014. An enhanced grouping proof for multiple RFID readers and tag groups. *International Journal of Control and Automation*, *7*(12), pp.239-246.

120. Song, B., & Mitchell, C. J. (2008, March). RFID authentication protocol for low-cost tags. In *Proceedings of the first ACM conference on Wireless network security* (pp. 140-147). ACM.

121. Songhela, R. and Das, M.L., 2014, October. Yet another strong privacy-preserving RFID mutual authentication protocol. In *International Conference on Security, Privacy, and Applied Cryptography Engineering* (pp. 171-182). Springer, Cham.

122. Stinson, D. R. (2006). *Cryptography: theory and practice*. CRC press.

123. Trappe, Wade, and Lawrence C. Washington. *Introduction to cryptography with coding theory*. Pearson Education India, 2006

124. Tuyls, P., & Batina, L. (2006). RFID-tags for Anti-Counterfeiting. In *Topics in cryptology-CT-RSA 2006* (pp. 115-131). Springer Berlin Heidelberg.

125. Vaidya, B., Makrakis, D., & Mouftah, H. T. (2012, September). Robust RFID Authentication for Supply Chain Management. In *Vehicular Technology Conference (VTC Fall), 2012 IEEE* (pp. 1-5). IEEE.

126. Van Deursen, T., & Radomirovic, S. (2008). Attacks on RFID Protocols. *IACR Cryptology ePrint Archive*, *2008*, 310.

127. Van Deursen, T., & Radomirovic, S. (2009). Untraceable RFID protocols are not trivially composable: Attacks on the revision of EC-RAC. *IACR Cryptology ePrint Archive*, *2009*, 332.

128. Vaudenay, S., 2007. On privacy models for RFID. *Advances in Cryptology-ASIACRYPT 2007*, pp.68-87.

129. Violino. 2005. *RFID journal*. [ONLINE] Available at: http://www.rfidjournal.com/articles/view?1337. [Accessed 1 December 17]

130. Walczyk, D. 2009. RFID and Its Message. *Library Journal,* Vol. 134 Issue 1, S4-S7 2009.

131. Wang, S.H., Liu, S. and Chen, D.W., 2014. Analysis and Construction of Efficient RFID Authentication Protocol with Backward Privacy. *IACR Cryptology ePrint Archive*, *2012*, p.391.

132. Washington, L.C., 2008. *Elliptic curves: number theory and cryptography*. CRC press

133. Wei, C. H., Hwang, M. S., & Chin, A. Y. H. (2011). A mutual authentication protocol for RFID. *IT Professional*, (2), 20-24.

134. Weis, S. A. (2003). *Security and privacy in radio-frequency identification devices* (Doctoral dissertation, Massachusetts Institute of Technology).

135. Weis, S. A. (2007). Rfid (radio frequency identification): Principles and applications. *System*, *2*, 3Principles.

136. Weis, S. A., Sarma, S. E., Rivest, R. L., & Engels, D. W. (2004). Security and privacy aspects of low-cost radio frequency identification systems. In *Security in pervasive computing* (pp. 201-212). Springer Berlin Heidelberg.

137. Wolkerstorfer, J. (2005, July). Is elliptic-curve cryptography suitable to secure RFID tags. In *Workshop on RFID and Lightweight Cryptography, Graz-August*.

138. Zhang, X., Li, L., Wu, Y., & Zhang, Q. (2011, May). An ECDLP-Based Randomized Key RFID Authentication Protocol. In *Network Computing and Information Security (NCIS), 2011 International Conference on* (Vol. 2, pp. 146-149). IEEE.

139. Zhang, Z. and Qi, Q., 2014. An efficient RFID authentication protocol to enhance patient medication safety using elliptic curve cryptography. *Journal of medical systems*, *38*(5), p.47