2011

Security and authentication schemes in RFID

Shu Cheng
University of Wollongong

Security and Authentication Schemes in RFID

A thesis submitted in fulfillment of the
requirements for the award of the degree

Master of Computer Science by Research

from

UNIVERSITY OF WOLLONGONG

by

Shu Cheng

School of Computer Science and Software Engineering
December 2011

Dedicated to
My parents and my grandmother

Declaration

This is to certify that the work reported in this thesis was done by the author, unless specified otherwise, and that no part of it has been submitted in a thesis to any other university or similar institution.

Shu Cheng
December 12, 2011

Abstract

Radio Frequency Identification (RFID) has been widely applied in various applications in the modern world. For example, it can be used in supply chain management, automated payment systems and other daily applications as an essential technology to enhance lives of human beings. However, RFID systems are vulnerable to many malicious attacks against security and privacy. To solve these problems, cryptography must be applied. Our research work in this thesis mainly focuses on designing secure RFID authentication schemes with untraceability. We review a number of recent proposed RFID authentication protocols as well as related cryptographic techniques, and then define the security and privacy requirements for our RFID systems. Our main contributions in this thesis consist of two proposed RFID authentication schemes.

The first scheme is a symmetric-key-based authentication scheme for low-cost RFID tags. RFID systems used in this scheme conform to EPCglobal Class-1 Generation-2 RFID Specification. The work is an improvement of the protocol proposed by Yeh, Wang, Kuo and Wang (YWKW) in 2010. We investigate the YWKW protocol and present the man-in-the-middle attack and the strong tracing attack on their protocol. Our scheme successfully overcomes these drawbacks without impacting the performance advantage. Besides, our scheme achieves both backward untraceability and forward untraceability.

In last few years, some basic operations on elliptic curves have been proved applicable for low-cost RFID tags, which make elliptic-curve-based cryptography possible for RFID protocols. We proposed our second scheme constructed on elliptic curves using public-key cryptography. We prove the unforgeability for our scheme in the random oracle model. The security of our scheme is based on the hardness of the Gap Diffie-Hellman problem. We provide a rigorous privacy proof for our scheme based on the Vaudenay's privacy model. To our knowledge, it is the first secure

elliptic-curve-based authentication protocol that achieves both narrow-destructive and wide-forward privacy. Our scheme is also scalable for large-scale deployment in practice.

Acknowledgement

I would like to express my sincere gratitude to my supervisors, Professor Yi Mu and Professor Minjie Zhang, for giving me an opportunity to do research and directing me into RFID security. Thank you for your patient guidance, indispensable instructions and excellent comments during my study.

I would like to thank Professor Yong Yu for his comments on my research. I also want to thank all my fellow students and staff in School of Computer Science and Software Engineering, University of Wollongong.

I am supremely grateful to my parents and my grandmother. You always support me whenever I need. This thesis would never be possible without your encouragements.

Publications

Shu Cheng, Yi Mu and Minjie Zhang. Ananlysis and Improvement of A Mutual Authentication Protocol for RFID, submitted to Journal of Computer Standards & Interfaces. (under revision)

Shu Cheng, Yi Mu, Minjie Zhang and Yu Yong. A Public-Key-Based RFID Authentication Protocol with Untraceability. (draft)

Contents

List of Tables

List of Figures

Chapter 1

Introduction

This chapter provides an overview of the thesis. We introduce the background of RFID technology in Section 1.1. Section 1.2 presents several commonly used applications using RFID technology. We describe the research issues in RFID systems in Section 1.3, and explain the motivations of our research in Section 1.4. The contributions of this thesis are highlighted in Section 1.5 and the organisation of the thesis is given in Section 1.6.

1.1 RFID Technology

Radio Frequency Identification (RFID) is a technology which can identify and track objects automatically using radio waves. It has been considered as a substitute of barcode and offers some attractive features. An RFID system is composed of three primary parts: tags, readers and a back-end server [THD+06, GB06], which are illustrated in Figure 1.1.

A typical RFID tag is a radio transponder designed to receive and send radio-frequency signals, which combines a microchip and an antenna [Fin03]. The microchip is designed for storing and processing data, while the antenna is used to send and receive radio frequency signals. The tag is usually attached to an item. Depending on the method of its power supply, a tag can be either a passive tag, an active tag or a semi-passive tag [Jue06].

Passive tags do not have internal power source, such as batteries, so they are cheap and small. The microchip inside the tag is able to gain power from the reader's interrogation radio signals. Consequently, a communication session can only be initiated by a reader and the range between a passive tag and a reader is limited [Jue06, THD+06]. The computational ability of a passive tag is weak.

Both active tags and semi-passive tags contain batteries. The difference between

them is that, active tags are able to power their memory circuitry and radio circuits while semi-passive tags still need the reader's radio-frequency signals to power their radio circuits. Since they have their own power source, they can communicate with readers in a greater distance compared to passive tags [Jue06, THD+06].

An RFID reader is a radio transceiver. It contains an antenna to query a tag via radio signal from distance. A reader is normally connected to a back-end server, which contains a database about the information of items, through a secure channel. Upon receiving the response of a tag, the reader can access the database and retrieve the information of the item to which the tag is attached [THD+06]. The communication between a reader and a database server is assumed to be secure. For convenience, the reader and the database server are always considered as a single entity.

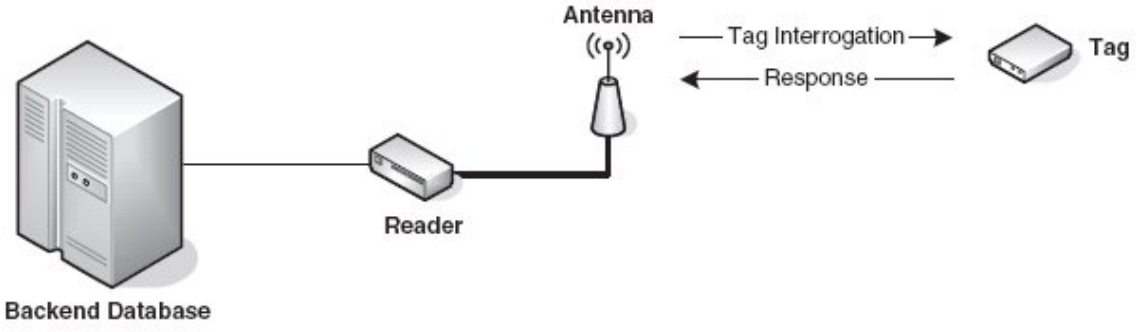

Figure 1.1: RFID System Components [THD+06]

A number of organisations, including International Organisation for Standardisation (ISO), the International Electrotechnical Commission (IEC) and EPCglobal, have been working on standards for RFID technology [RFI]. Among all these standards, EPCglobal UHF Class 1 Generation 2 (EPC Class-1 Generation-2) [EPC08], proposed by EPCglobal, is designed for passive RFID tags in the supply chain [RFI]. Every tag conforming to the EPC Class-1 Generation-2 contains a unique Electronic Product Code (EPC), which is used as an identity for goods in the supply chain.

1.2 RFID Applications

The RFID technology has become widespread due to its low cost and easy deployment. This section briefly introduces several applications using RFID technology that are widely deployed in our daily life.

The widest application about RFID is the supply chain and logistics [KS09, LD07, RCT06, Rob05, DOD03]. Products attached with RFID tags can be monitored in the whole supply chain. For each RFID supply chain procedure, numerous RFID tags are functioned as a quantity of goods that are involved. Thus, passive tags with low price and recyclability will reduce the overall cost and improve the efficiency of inventory tracking and management.

Access control is another typical application of RFID tags [GJP05, Jue06]. They can be used as an access key to pass the security entrance of confidential department [RCT06, Rot08]. They can also be used as a security device to automatically identify vehicles [GJP05]. An RFID tag is attached to a car key so that the automobile can be launched only when the key is closed enough, i.e., the reader receives a responding signal from the tag. This application is reported to effectively reduce the auto theft [AC01].

Animal Tracking is one of the oldest applications of RFID technology. Animals have been implanted with RFID tags to help tracking, management and scientific research. Lost animals can be easily found and returned to their owners by tracking the tags on them [GJP05, PG07]. Moreover, scientists have used RFID-based animal tracking to observe and control the outburst of animal diseases such as mad cow disease and bird flu [RCT06].

RFID technology is adopted in the passports of some countries [Jue06, Lau07, Rot08]. The RFID tag embedded in a passport records the information of the holder. These RFID-enabled passports are difficult to forge in comparison with the traditional passports. Therefore, RFID technology can expedite exit and entry formalities and improve national security.

It has been realised that the traditional payment methods with credit card or cash are quite inefficient. Paying by cards requires customers to sign a receipt or entering a personal identification number to confirm the payment, while paying by cash needs shop assistants to collect the money and give the change. Recently, some credit card companies begin to offer a contactless payment system with RFID

technology integrated [O'C05]. The point-of-sale terminals in this kind of RFID-enabled systems can make the check-out procedure more efficient and convenient.

RFID can be embedded into smart appliances. For example, a washing machine with an RFID reader can read the tag of a clothes and then run a specific washing process [Mer03]. A smart oven is also capable of reading the instruction from the tag and deciding how to cook the food [Mer03].

RFID technology facilitates healthcare as well [WVL06]. It can be used to efficiently search treatment record for a patient, monitor patient's drug treatment and locate patients. The RFID system also can provide automated processes to reduce the high cost of hospitals and decrease mistakes during in order to improve safety for patients.

1.3 Research Issues

Low-cost RFID tags are used in most of the RFID systems due to inexpensive cost and easy deployment [Sys]. These tags have a limited memory and weak computational capability. In consequence, the security and privacy issues have become growing concerns for RFID systems. We describe these issues in RFID systems as follows.

- The communication between a reader and a tag is wireless via radio frequency [THD+06, GB06]. An attacker can easily eavesdrop, interrupt, modify and counterfeit the messages of the communication.

- Each RFID tag contains a unique ID (for example, Electronic Product Code), which can be used to identify it. Most tags either broadcast their unique IDs or blend their IDs into the communication messages. As a result, the adversary may possibly track a tag using the unique ID captured in the communication [Jue06], which may lead to the leakage of location information of the tag and corresponding object. Moreover, some tags may also contain the information about the objects that they are attached to, such as information of books borrowed by a person [MW04], even personal confidential in an e-passport [JMW05]. Leakage of this type of information may compromise personal privacy.

- RFID tags with tight memory and restricted processing ability are vulnerable to compromise [MRW04]. As it has already been mentioned above, an adversary can eavesdrop communication sessions. It can, in addition, either extract the secret of a tag from the communication messages and digitally forge a tag, or simply raise active attacks such as man-in-the-middle attacks to impersonate a valid tag [Jue05]. A strong adversary is also able to breach the back-end server and obtain the secret key to raise cloning attacks [Jue05]. The other approach to clone legitimate tags is physical counterfeiting. An adversary can reverse-engineer a tag and then attain all the secret key of the tag. In this case, the cloning is perfect [Jue05].

1.4 Motivations

To address all these issues mentioned above, the research should concern several properties regarding RFID security and privacy.

- Authentication is an act that one entity proves its validity to another entity in a communication session. Tag authentication means a tag proves its validity to a reader, while reader authentication is in the opposite way. Mutual authentication is referred to both tag and reader authentication. In particular, tag authentication is more important since tags are much easier to counterfeit than readers. Authentication is an effective approach to prevent impersonation attacks.

- Untraceability means that a tag is both anonymous and indistinguishable to an adversary. Although an RFID tag is vulnerable to compromise, an malicious reader should not identify the communication records of this tag in the past or future session. Thus, a protocol should not leak the identity or internal secret of RFID tags in communication processes.

- Performance requirements should be satisfied. Considering the limited storage and computational ability of low-cost tags, cryptographic techniques used in the schemes should be basic and restricted [WSRE04]. Besides, a back-end server may deal with a plenty of tags in one RFID system in real world scenarios. Thus, an RFID scheme should also be scalable [AO05] and especially easy for a server to search the information of a tag in the database.

For these reasons, the thesis focuses on designing secure and untraceable authentication schemes for light-weight RFID systems.

1.5 Contributions of the Thesis

This thesis concentrates on the security and privacy issues in RFID systems. In order to clarify the security and privacy requirements, we review a number of existing authentication protocols proposed to counter various attacks for RFID systems. Two authentication schemes based on symmetric-key cryptography and public-key cryptography have been proposed respectively to deal with these issues.

The main contributions of the thesis include:

- We investigate and study a cryptanalysis of a symmetric-key-based RFID authentication protocol recently proposed by Yeh, Wang, Kuo and Wang (YWKW) in 2010 [YWKW10]. Their protocol is established based on the EPC Class-1 Generation-2 specification. We analyse the strengths and weakness of the YWKW protocol and show that although it provides low database loading, it is still vulnerable to several attacks such as man-in-the-middle attacks and tracing attacks. As a result, an adversary can forge messages and identify past and future interactions between a tag and a reader.

- We propose a light-weight RFID authentication scheme using symmetric-key cryptography. The scheme is an improvement of the YWKW protocol and also complies with the EPC Class-1 Generation-2 specification. We show that our proposed scheme overcomes the drawbacks of the YWKW protocol without losing the performance advantage. Moreover, our scheme can achieve both backward untraceability and forward untraceability.

- We develop a novel public-key-based RFID authentication scheme with untraceability. Our scheme is constructed on elliptic curves. We prove the security of our scheme by deducing the unforgeability to the hardness of Gap Diffie-Hellman problem in the random oracle model. Our scheme is also provably privacy-preserved in the Vaudenay's model. We compare our scheme to other elliptic-curve-based schemes for RFID systems and show that it is the first secure elliptic-curve-based RFID authentication scheme that can achieve

both narrow-destructive and wide-forward privacy. Our scheme is scalable for large-scale deployment.

1.6 Organisation of the Thesis

The rest of this thesis is organised as follows.

- Chapter 2 presents the cryptographic background knowledge for the thesis. Firstly, we introduce the cryptographic primitives, including cryptographic hash function, pseudorandom number generator, symmetric-key cryptography, public-key cryptography and digital signature. Secondly, we define the complexity problems. Finally, we briefly describe the random oracle model and Vaudenay's security and privacy model.

- Chapter 3 reviews related RFID authentication protocols based on both symmetric-key and public-key cryptography.

- Chapter 4 presents the cryptanalysis of the YWKW protocol and proposes an improvement that also complies with the EPC Class-1 Generation-2 specification. We also analyse our symmetric-key-based RFID authentication scheme and show that it overcomes the drawbacks of the YWKW protocol.

- Chapter 5 proposes a public-key-based RFID authentication scheme with untraceability. The rigorous security and privacy proofs for this scheme are also provided.

- Chapter 6 concludes the thesis by highlighting the contributions of this study, and points out the future work.

Chapter 2

Background

This chapter introduces related background knowledge for this thesis. In Section 2.1, we give a briefly description of some cryptography primitives that will be used in Chapters 3, 4 and 5 of this thesis, including hash function, pseudorandom number generator, symmetric-key cryptography, public-key cryptography as well as digital signature. In Section 2.2, we present several computational hard problems such as the discrete logarithm problem and the diffie-hellman problem. In Section 2.3, the random oracle model and Vaudenay's privacy model are visited. Section 2.4 summarises this chapter.

2.1 Cryptographic Primitives

2.1.1 Cryptographic Hash Function

Hash function is one of the fundamental cryptographic primitives. A hash function is a mathematical function which maps an input of arbitrary length to a output with some fixed length, which is called *hash value* or *hash value* [MvOV97, Mit04]. Hash value can be considered as a digital fingerprint of an input message. Hence, hash function is often used to protect the integrity of data [Sti06]. A formal definition of hash function is described as follows.

Definition 2.1 *Let l denote a fixed number. A function $H : \{0,1\}^* \rightarrow \{0,1\}^l$ is a hash function, if it satisfies the following properties:*

- *Efficiency: Given any $x \in \{0,1\}^*$ as an input of H, it is easy to compute $H(x)$.*
- *Deterministic function: Given a same value x as an input, H always outputs the same $H(x)$.*

- *Preimage resistance: Given* $y \in \{0,1\}^l$*, it is computationally hard to generate a value* x *such that* $H(x) = y$*.*
- *Second Preimage resistance: Given* $x \in \{0,1\}^*$*, it is computationally hard to generate a value* $x^{'} \neq x$ *such that* $H(x) = H(x^{'})$*.*
- *Collision resistance: It is computationally hard to find* $x, x^{'} \in \{0,1\}^*$ *such that* $x \neq x^{'}$ *and* $H(x) = H(x^{'})$*.*

There are two types of hash functions. Unkeyed hash functions and keyed hash functions [Sti06]. The difference between these two types are that, a keyed hash function takes both a secret key k and a message x as inputs, while a unkeyed hash function takes only the message x. Keyed hash functions can be used to guarantee the hash value is not counterfeit as only the parties that own the shared secret key can generate the hash value of the original message. Message Authentication Code (MAC) can be implemented using keyed hash function.

The most widely used cryptographic hash functions are MD5 and SHA-1. MD5 is designed by Ron Rivest [Riv92a] in 1992. SHA-1 is developed by National Security Agency (NSA) and published by National Institute of Standards and Technology (NIST). Although they are applied in an abundance of security applications and communication protocols, both of them are provable not secure. MD5 can not resist preimage attack and collision attack [WY05, XF09, SA09], while SHA-1 is vulnerable to collision attack [WYY05, Man11].

2.1.2 Pseudorandom Number Generator

Random numbers are used in almost all the cryptographic protocols. However, it is not practical to generate a number that is truly random since the generating procedure is very inefficient. For this reason, pseudorandom number generator (PRNG), also known as pseudorandom bit generator, is introduced to be applied in practice. A pseudorandom number generator is an algorithm that takes an arbitrary sequence as a seed to produce sequences that looks random [MvOV97]. The output sequences actually have a period. When the end of the period is reached, generator will repeat the results from the beginning. PRNG is also a deterministic function so that when the same seed is inputed, it will output the same results. The robustness of a PRNG depends on the period and probability distribution of the output

sequence [DPLK08]. In the rest of this thesis, the term "random" is referred to pseudorandom.

2.1.3 Symmetric-key Cryptography

Symmetric-key cryptography is a type of cryptographic systems that uses only one key for both encryption and decryption [MvOV97, Mao04]. In symmetric-key cryptosystems, the sender, Alice, and the receiver, Bob, share an identical secret key. When a message is sending to Bob, Alice uses the shared secret key to encrypt the message into the ciphertext. Upon receiving the ciphertext, Bob decrypts it into plaintext using the same secret key.

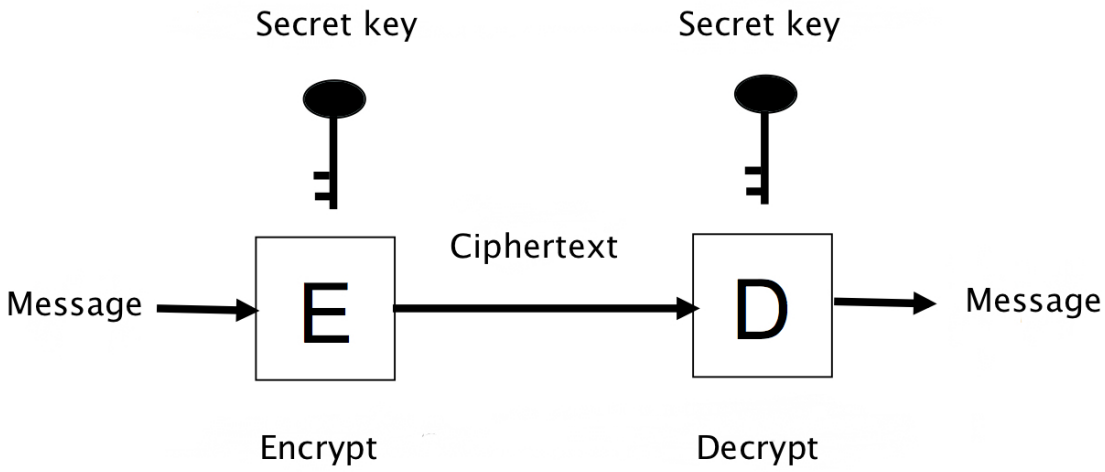

Figure 2.1: Symmetric-key Encryption and Decryption

Symmetric key cipher can be split into two types according to different encryption methods: stream cipher and block cipher. A stream cipher algorithm encrypts a message by combining a digit or bit of the plaintext with a key stream each time, while a block cipher algorithm encrypts a fixed length block of the plaintext into a block of the output ciphertext [MvOV97]. RC4 proposed by Ron Rivest [Riv92b] is the most famous stream cipher that is generally implemented into several cryptographic protocols such as Secure Sockets Layer (SSL). Notable block ciphers include Data Encryption Standard (DES) [Cop94], Advanced Encryption Standard (AES) [DR00], RC5 [Riv95], International Data Encryption Algorithm (IDEA) [LM06],

etc. Block ciphers can be used to construct many cryptographic primitives such as cryptographic hash functions, message authentication codes, pseudorandom number generator and stream cipher.

Symmetric-key cryptography has a number of features [MvOV97]. Firstly, encryption and decryption processes are fast and efficient in symmetric-key cryptosystems in terms of implementations of hardware. Secondly, key sizes for secret keys are short. On the other hand, the same key for two entities in one communication needs to be secret. Thus it comes to the problem how to securely exchange the secret key. Currently the effective solution is to use trusted third party. The other problem is that an entity may need to store enormous amounts of keys in a large-scale communication network in certain scenarios.

2.1.4 Public-key Cryptography

Unlike symmetric-key cryptography, public-key cryptography uses a key pair to process encryption and decryption. It was proposed by Whitfield Diffie and Martin Hellman in 1976 [DH76a]. A communication entity intended to receive messages chooses a key pair that composed of a public key and a private key. He keeps the private key secret and broadcasts the public key publicly so that any entity that wants to send a message to him in secure channel can use the public key to encrypt the message. Upon receiving the ciphertext, the recipient decrypts the ciphertext using the private key, which is only known by himself.

Definition 2.2 *A encryption scheme is called public-key encryption scheme, if it comprises three algorithms:*

- $Gen(n) \rightarrow (sk, pk)$*: Given a secure parameter* n *as input, the randomized algorithm outputs a key pair where* sk *is the private key and* pk *is the public key.*
- $Encrypt(pk, m) \rightarrow c$*: Given the public key* pk *and a message* m*, the algorithm outputs a ciphertext* c*.*
- $Decrypt(sk, c) \rightarrow m$*: Given the private key* sk *and a ciphertext* c*, the algorithm outputs a message* m*.*

Many public-key cryptosystems have been published and applied in the real world. The RSA encryption algorithm [RSA78], ElGamal algorithm[ElG85] and

Cramer-Shoup algorithm [CS00] are the most famous encryption algorithms among all the cryptosystems.

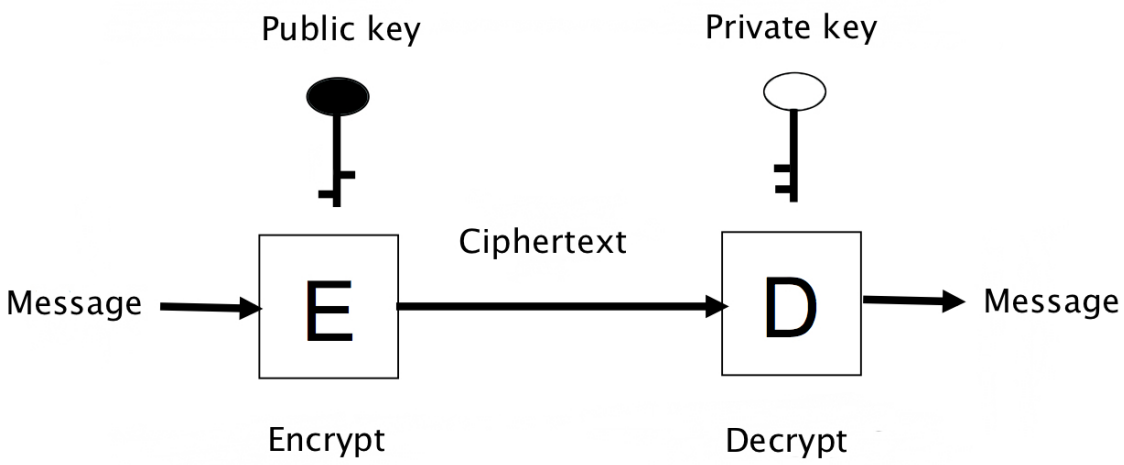

Figure 2.2: Public-key Encryption and Decryption

Compared with the symmetric-key cryptography, public-key encryption has a variety of strengths and weaknesses [MvOV97]. In public-key cryptosystems, only private keys are required to be secret. Key exchange is simpler than symmetric-key cryptography as public keys can be publicly broadcast. In addition, public-key cryptography can be developed into digital signature, which we will introduce in the next section. In a communication network, public-key cryptosystems may require less keys than symmetric-key cryptosystems. However, public-key encryption is less efficient than symmetric-key cryptography and the required sizes of keys are larger.

2.1.5 Digital Signatures

In an open communication network, messages can be easily forged by an adversary. It is necessary to protect the data integrity. There are two methods that be employed in practice, which are Message Authentication Code (MAC) and Digital Signature [Mao04]. As we mentioned above, MAC can be implemented using cryptographic hash functions or block ciphers. Now we briefly describe the digital signature technique.

Digital signature is the most significant application of public-key cryptography

that is introduced by Diffie and Hellman in 1976 [DH76a, DH76b]. In a digital signature algorithm, a signer uses his private key to generate a signature of a message. Any other entity can use the public key of the signer and the message to verify the signature afterwards. As the private key is only known by the signer himself, the authenticity of the message is guaranteed by the validity of the signature.

Definition 2.3 *A scheme is called digital signature scheme, if it consists of three algorithms:*

- $Gen(n) \rightarrow (sk, pk)$*: Given a secure parameter* n *as input, the randomized algorithm outputs a key pair where* sk *is the private key and* pk *is the public key.*
- $Sign(sk, m) \rightarrow \sigma$*: Given the private key* sk *and a message* m*, the algorithm outputs a signature* σ*.*
- $Verify(pk, m, \sigma)$*: Given the public key* pk*, the message* m *and a signature* σ*, the algorithm outputs true if* $Sign(sk, m) \rightarrow \sigma$*; otherwise outputs* $false$*.*

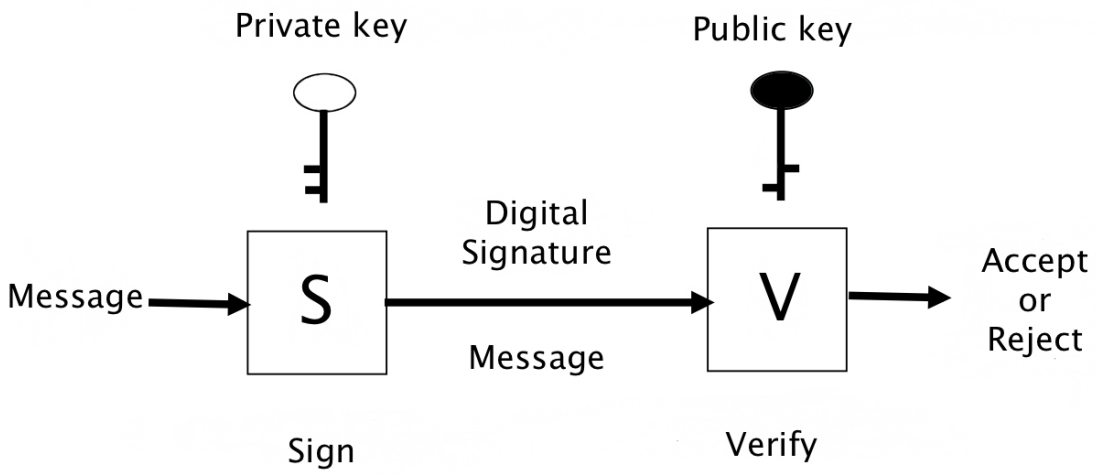

Figure 2.3: Digital Signature

There are various digital signature schemes, such as RSA signature scheme [RSA78], ElGamal signature scheme [ElG85], Rabin signature scheme [Rab79], Digital Signature Standard and Schnorr signature [Sch90]. We will briefly explain the Schnorr signature as it is related to our work in Chapter 5.

Schnorr Signature, proposed by Schnorr in 1990, is derived from ElGamal signature. The security of this scheme is constructed on the hardness of Discrete Logarithm problem on finite fields. It have been proved secure in the random oracle model [BR93]. The signature generated from a message is relative small so that it is very practical for devices with restricted computational abilities, for example, smart card.

The details of the scheme are specified as follows.

- Parameters Setup:
  Randomly choose two primes p and q such that $q|p-1$;
  choose $g \in \mathbb{Z}_p^*$ such that $g^q = 1 \ (mod\ p)$ and $g \neq 1$;
  choose a cryptographic hash function $H : \{0,1\}^* \rightarrow \mathbb{Z}_q$.
  p, q, g, H are the system parameters published for all the users.

- Key Generation:
  The signer chooses a random number $x \in \mathbb{Z}_q$ as his private key. He computes $y = g^{-x} \ (mod\ p)$ and publishes y as the corresponding public key.

- Signing:
  To sign a message $m \in \{0,1\}^*$, the signer chooses a random number $l \in \mathbb{Z}_q$ and computes

$$\begin{aligned} r &= g^l \ (mod\ p), \\ e &= H(m \| r), \\ s &= l + xe \ (mod\ p). \end{aligned}$$

  (e, s) is the signature pair that will be sent with the message.

- Verifying:
  Upon receiving the message m and the associated signature (e, s), a verifier computes

$$\begin{aligned} r' &= g^s y^e \ (mod\ p), \\ e' &= H(m \| r'). \end{aligned}$$

  The result is true if $e' = e$; otherwise the result is false.

2.2 Computational Hard Problems

A mathematical problem is said to be *hard* if there is no algorithm that can solve it in polynomial time. Most public-key cryptosystems in modern cryptography are either directly or indirectly relevant to computational hard problems as the security of each cryptographic scheme is established on the hardness of a computational hard problem [MvOV97]. There are a number of computational hard problems used in cryptography. In this section, we review some computational hard problems that related to our proposed schemes.

2.2.1 Discrete Logarithm Problem

The discrete logarithm problem is one of the most basic computational hard problems in cryptography. Computational Diffie-Hellman problem and all its variants involve the discrete logarithm as a basis. Discrete logarithms are similar to the conventional logarithms except they effect on finite fields. There is no efficient algorithm solving the discrete logarithm problem in polynomial time so far. Hence, it has been considered as one of the computational hard problems in mathematics and cryptography [COS86, McC90]. The definition of the discrete logarithm problem is shown below.

Definition 2.4 Discrete Logarithm Problem
Given a cyclic group $\mathbb{G}$ *and a generator* $g \in \mathbb{G}$*, for any unknown randomly chosen* $h \in \mathbb{G}$*, computes the unique* a *such that* $h = g^a$*.*

2.2.2 Computational Diffie-Hellman Problem

Computational Diffie-Hellman problem, first proposed by Whitfield Diffie and Martin Hellman [DH76a] in 1976, is another intractable problem along with discrete logarithm problem. The relationship between computational Diffie-Hellman problem and discrete logarithm problem is significant as an algorithm that solves discrete logarithm problem in polynomial time can certainly solve computational Diffie-Hellman problem.

Definition 2.5 Computational Diffie-Hellman (CDH) Problem

Let $\mathbb{G}$ be a cyclic group with order p, where p is a prime. Given a randomly chosen generator $g \in \mathbb{G}$, as well as g^a, g^b for unknown randomly chosen a, $b \in \mathbb{Z}_p^*$, compute g^{ab}.

2.2.3 Decisional Diffie-Hellman Problem

The CDH problem has some variants. Decisional Diffie-Hellman (DDH) problem [Bon98] is among the most notable variants. It has been used to prove the security of ElGamal crytosystem [ElG85] and Cramer-Shoup crytosystem [CS00]. The definition of DDH problem is as follows.

Definition 2.6 Decisional Diffie-Hellman (DDH) Problem
Let $\mathbb{G}$ be a cyclic group with order p, where p is a prime. Given a randomly chosen generator $g \in \mathbb{G}$, g^a, g^b, g^c for unknown randomly chosen a, b, $c \in \mathbb{Z}_p^*$, decide whether $g^c = g^{ab}$.

From the definitions of the CDH and the DDH problems, it is obviously that a algorithm that is able to solve CDH problem can solve DDH problem as well.

2.2.4 Gap Diffie-Hellman Problem

The Gap Diffie-Hellman (GDH) Problem is another variant of the Diffie-Hellman problem introduced in [OP01]. It has been used in the security proofs of some undeniable signatures and designated verifier signatures, such as Chaum's signature [CvA90].

Definition 2.7 Gap Diffie-Hellman (GDH) Problem
Let $\mathbb{G}$ be a cyclic group with order p, where p is a prime. Assume there exists an efficient polynomial-time algorithm $\mathcal{O}$ that solves the DDH problem in $\mathbb{G}$ and there is no probabilistic polynomial-time algorithm that solves the CDH problem with non-negligible probability. Given a randomly chosen generator $g \in \mathbb{G}$, and g^a, g^b for unknown a, $b \in \mathbb{Z}_p^*$, compute g^{ab} with the help of the DDH oracle $\mathcal{O}$.

2.3 Provable Security

2.3.1 Random Oracle Model

We have already mentioned the concept of cryptography hash functions in the first section of this chapter. In the definition of hash functions, the term "computationally hard" means in polynomial time it is not able to compute the result. However, in the security proof, it often requires the output of a hash function to be truly uniform. As a result, random oracle is introduced to modelling cryptographic hash function. A random oracle is a deterministic function as a hash function that for the same input, it always returns the same result. Moreover, the outputs of random oracles are truly uniform. It is an ideal function that cannot be implemented in the real world according to Shannon's theory [Sha48].

In a security proof on the random oracle model, a simulator simulates all the random oracles for all the entities using a list to record the results of random oracle queries [BR93]. As long as the simulator keeps the deterministic and uniform properties of a random oracle, the simulation is perfect. Cryptographic schemes, in some cases, can be deduced to a computational hard problem more easily than in the standard model.

2.3.2 Vaudenay's Privacy Model

It is essential to study formal RFID security and privacy models because they are necessary for designing and analysing robust RFID protocols. There are several security models proposed for RFID [DLYZ10, HMZH08, JW07, MLDL09, Vau07]. Among these frameworks, Vaudenay's model is one of the most systematic frameworks and cited in a wide variety of RFID privacy analysis. Here, we briefly introduce Vaudenay's model, which will be detailed in Chapter 5.

Vaudenay's model defines eight oracles to classify the adversary's attack power, which are

- $\mathsf{CreateTag}^{\mathsf{b}}(\mathsf{ID})$ creates a free tag with an identity ID and a bit b. The tag is legitimate if $b = 1$ and not legitimate if $b = 0$;

- $\mathsf{DrawTag}() \rightarrow (\mathsf{vtag}_1, \mathsf{b}_1, \ldots, \mathsf{vtag}_n, \mathsf{b}_n)$ makes several free tags become drawn. $vtag$ is the virtual ID of a tag and b indicates whether the tag is legitimate or

not;

- Free(vtag) makes a drawn become free, which means an adversary can no longer access the tag with a virtual ID $vtag$.
- Launch $\rightarrow \pi$ starts a session π of the protocol;
- SendTag(m, vtag) $\rightarrow$ m′ sends the message m to the virtual tag $vtag$ and returns a message m';
- SendReader(m) $\rightarrow$ m′ sends the message m to the reader and returns m';
- Result(π) $\rightarrow$ b returns a bit b to indicate whether the protocol session π is completed successfully or not;
- Corrupt(vtag) $\rightarrow$ S returns the secret state S of the tag.

A Wide-Weak adversary is allowed to access all the oracles except Corrupt; a Wide-Forward adversary can access no oracles but Corrupt oracle once Corrupt has been accessed for the first time; a Wide-Destructive adversary will not access a tag any more if it has Corrupted the tag; a Wide-Strong adversary is the strongest adversary that is allowed to access any oracles at any time. Every Wide adversary also has a corresponding Narrow adversary, who can query the same oracles except the Result oracle.

Vaudenay also introduces the *Blinder* to simulate protocol messages for adversaries. An adversary is trivial if it does not use the communication messages to compromise a tag; that is to say, the adversary cannot distinguish between the real communication and the messages simulated by a blinder. If an RFID system is secure against a class of adversaries, i.e. all the adversaries from the some class are trivial, the system achieves the corresponding privacy class.

2.4 Summary

This chapter introduced the preliminaries of cryptography for the thesis. We reviewed cryptographic primitives in Section 2.1 which will be used in analysing existing protocols and designing our robust schemes. Computational hard problems and security models for the security proof of our schemes were also presented in Section 2.2 and 2.3 respectively.

Chapter 3

Literature Review

The RFID authentication schemes proposed in this thesis are constructed on symmetric-key cryptography and public-key cryptography respectively. In this chapter, we first review the related work regarding symmetric-key cryptography in Section 3.1, and then we investigate recent proposed public-key-based RFID authentication protocols in Section 3.2. Section 3.3 summarises this chapter.

3.1 Symmetric-Key-Based Authentication Schemes

Although a lot of approaches attempt to address the security and privacy issues for RFID, few in practice on account of either security, privacy, or performance drawbacks. In this section, we review several previous symmetric-key-based authentication protocols for RFID, especially those conforming to EPC Class-1 Generation-2 specification. We summarise their security properties and weaknesses.

- **Hash-Based Access Control and Randomized Access Control**

  Weis, Sarma, Rivest and Engels [WSRE04] proposed two RFID authentication protocols called Hash-based Access Control (HAC) and Randomized Access Control (RAC) respectively in 2003. Their protocols are both built on one-way hash functions to control the access to tags.

  In HAC, an RFID tag is normally "locked" so that it only answers a temporary ID called $metaID$ to all the queries from random readers. Only the reader with the possession of the tag's secret key k can unlock the tag and get access to its private information. This protocol requires only a hash function implemented in a light-weight tag. The key for unlocking is only stored in the back-end server so that corrupt the tag will not leak the secret key. However, since the

fixed $metaID$ is repeatedly used in different sessions, the locked tag can be tracked [WSRE04].

In order to prevent the tag from tracing, the RAC protocol embeds a random number with the tag ID in the hash function to replace $metaID$. Nevertheless, it does not provide backward untraceability because the random number can be replayed and the tag ID is also fixed.

- **OSK Protocol**

  Ohkubo, Suzuki and Kinoshita (OSK) [OSK03] presented a privacy protection scheme for RFID relying on a low-cost hash chain approach in 2003. The secret key of an RFID tag is renewed using a one-way hash function after the tag sends the response to the reader's query. They also proposed a new security requirement called backward untraceability, which prevents a tag to be identified in the past communication sessions. The OSK protocol is proved to achieve backward untraceability. Even if a strong adversary compromises the tag and acquires the knowledge of the current secret key, it is impossible for it to reveal the past secret key and then identify the tag due to the property of hash chain. However, it was found that the protocol does not protect the system against replay attacks [ADO06]. An adversary can reuse the communication messages to pretend to be a valid tag without knowing the secret key.

- **MW Protocol**

  Molnar and Wagner's (MW) scheme [MW04] was designed for library RFID. In their basic mutual authentication protocol, the tag and the reader share a secret ID. The authors implemented a pseudo-random function and the exclusive or operation to protect the communication messages. They extended the protocol to a tree-based private authentication protocol. This expansion protocol is scalable for large-scale RFID applications such as library RFID. It reduces the complexity of identifying tags from $O(n)$ to $O(\log n)$. However, a tag can still be identified once it is compromised [CC07]. Therefore, it cannot achieve backward untraceability.

- **HM Protocol and Dimitriou Protocol**

  In [HM04], Henrici and Müller (HM) proposed an mutual authentication protocol. The protocol uses a one-way hash function and a conjunction operation to protect the privacy for tags. The back-end server stores the hash value of tags' ID in order to speed up the search process. The tag and the server update their shared secret once the authentication succeeds. In spite of that, the adversary can still trace the tag before the next successful authentication, since the tag always responses the same hash value of tag's ID during this period [CC07].

  Dimitriou's protocol [Dim05] is a mutual authentication providing backward untraceability and scalability. A one-way hash function can guarantee the untraceability of past communication session and the server stores the hash value of tags' ID to make the identification process efficient. However, it has the same problem as the HM protocol. Although it uses a challenge-response mechanism to ensure user privacy and updates the tag's ID in each session, the ID remains the same between valid session. An adversary can randomly query the tag to reveal the fixed hash value of the ID [SM08].

- **LK Protocol**

  Forward untraceability was first proposed in Lim and Kwon's work [LK06]. The concept is related to backward untraceability in [OSK03] but focuses on tag identification issues in future communication sessions. Lim and Kwon also proposed a mutual authentication scheme to provide both backward untraceability and forward untraceability. Three pseudorandom number functions and a extract function are implemented in the protocol. When reader authentication is successfully completed, both the reader and the tag update the share secret key using the old secret key blended by two random numbers exchanged in this authentication session. Backward untraceability is guaranteed by the one-wayness of pseudorandom number functions. Forward untraceability can be achieved if only the adversary misses one message from a successful authentication session. However, the intention of preventing denial-of-service attacks lead to another attack that allows an adversary to identify the tag without corrupt the tag [OP08].

- **YA-TRAP**

  YA-TRAP (Yet Another Trivial RFID Authentication Protocol) is a challenge-response scheme proposed by Tsudik in 2006 [Tsu06]. It is designed to provide tag authentication for RFID systems. The most notable advantage of this scheme is the fast tag identification process. Hash table is adopted in the protocol for tag lookup. It is more efficient compared to Molnar and Wagner's scheme. However, YA-TRAP is vulnerable to denial-of-service attacks and tracing attacks [Tsu06].

- **SM Protocol**

  Song and Mitchell defined several novel security threats for RFID systems, such as tag impersonation and server impersonation at WISEC '08 [SM08]. They also introduced an mutual authentication protocol claimed to be privacy-preserved and secure against these threats. It is the first RFID authentication protocol that employs bit operation. Keyed hash functions are also utilized in the scheme. The reader stores both the old and the new secret information for the tag. Cai *et al.* [CLLD09] discovered that Song and Mitchell's scheme cannot resist either tag impersonate attacks or sever impersonate attacks.

- **EPC Class-1 Generation-2 Specification Compliant Protocols**

  According to EPC Class-1 Generation-2 RFID specification [EPC08], an RFID tag is only capable of processing basic operations, such as Pseudo-Random Number Generator and Cyclic Redundancy Code (CRC). There are several EPC Class-1 Generation-2 specification compliant authentication protocols described below.

  1. Karthikeyan and Nesterenko [KN05] described an RFID tag identification algorithm, which is the first RFID authentication protocol based on simple matrix multiplication. Both the reader and the tag store a matrix as a shared secret key. Messages in the protocol are protected by matrix multiplication and exclusive or operations, which can be handled by EPC Class-1 Generation-2 tags [EPC08]. However, the scheme is vulnerable to denial of service attacks and brute-force matrix or key guessing attacks [KN05], moreover, replay attacks and individual tracing as argued in [CC07].

2. In 2007, Chien and Chen proposed a mutual authentication protocol in compliance with EPC Class-1 Generation-2 standards to improve the security performance. The back-end server stores both the old and new session keys to prevent denial-of-service attacks. It is also designed to defend denial of service attacks and provide backward untraceability as well. However, it has already been proven that this protocol is vulnerable to denial-of-service attacks, forward tracing attacks as well as tag and server impersonation attacks due to the linearity of Cyclic Redundancy Code (CRC) [PLHCETR09].

3. In 2008, Burmester and Medeiros [BdM08] proposed an mutual authentication protocol called TRAP-3 complying with EPC Class-1 Generation-2 standard. They constructed a cryptographic pseudorandom number function using the 16-bit pseudorandom number generator supported by the standard. The function aims to provide backward untraceability and tag anonymity. In 2010, Yeh and Lo [YL10] successfully performed denial-of-service attacks on this protocol so that the shared secret between a tag and a reader can easily be desynchronised.

4. Chen and Deng [CD09] presented an authentication and encryption protocol that is also compliant with EPC Class-1 Generation-2 specification. In this protocol, tags and readers need to register with the back-end server before further communication. Messages transmit between tags and readers are encrypted using cyclic redundancy code function. In 2011, Peris-Lopez, Hernandez-Castro, Tapiador and van der Lubbe [PLHCTvdL11] showed that an adversary is able to trace a certain tag by impersonating either a reader or a tag. In addition, the protocol is vulnerable to denial-of-service attacks if the reader receives an tampered identifier of the tag. They also concluded that cyclic redundancy code is a linear function so that it should never be used as a one-way function in cryptographic protocols.

3.2 Public-Key-Based Authentication Schemes

Public-key cryptography can achieve higher security levels compared to symmetric-key cryptography [Vau07], while it has the drawback of higher computational overhead. However, for computationally capable tags, it seems necessary to consider public-key cryptography in RFID systems. In recent years, the considerable amount of work [BGK+07, HWF09, LBSV10a, LSBV08, LBSV10b, LBV08, LBV09, MR06, TB06] have shown that public-key cryptographic techniques can be implemented into low-cost RFID tags. RFID tags are able to process modular additions, modular multiplications and elliptic curve scalar multiplications.

- **Conventional Signature Protocol**

  It has been observed that the Schnorr [Sch90] and Okamoto [Oka93] signature schemes of conventional cryptography have implemented in elliptic curve cryptography for RFID in [TB06] and [BGK+07], respectively. However, it was found that they do not meet the basic privacy requirements of RFID systems [LBV08]. An adversary can recover the public key of a tag in an authentication phase. Thus, the tag can be easily tracked in any communication session.

- **EC-RAC and the Improvement Protocols**

  In RFID'08, Lee, Batina and Verbauwhede [LBV08] described a new authentication protocol, ECDLP-based Randomized Access Control (EC-RAC), in order to fulfil all the security requirements for RFID systems. The EC-RAC protocol is based on the elliptic curve discrete logarithm problem (ECDLP). The protocol is one of the first practical RFID protocols using public-key cryptography. In 2009 [LBV09] and 2010 [LBSV10a, LBSV10b], three improvements for EC-RAC were proposed to solve different drawbacks. The aim of the EC-RAC set of RFID protocols is to provide stronger privacy for RFID systems than all symmetric-key-based protocols, as well as efficient tag lookup mechanism. Due to the design flaw, however, all versions of EC-RAC protocols are not secure. They are vulnerable to tracking attacks [BCI08, FHV10, vDR08]. Moreover, an adversary can collect information from several authentication sessions and raise man-in-the-middle attacks to impersonate a legitimate tag [BCI08, VDR10]. van Deursen [VDR10] analysed the EC-RAC protocols and showed that all the EC-RAC protocols are vulnerable to compositional attacks and none of them is secure against any wide adversary.

- **Randomized Schnorr Protocol**

  The randomized Schnorr protocol [BCI08] proposed by Bringer, Chabanne and Icart aims to solve tracking attacks and impersonation attacks against EC-RAC. It is proved to be narrow-strong in Vaudenay's privacy model [Vau07] as well as Zero-Knowledgeness. However, it has been shown that a wide adversary can still perform man-in-the-middle attacks to track the tag [LBSV10b].

3.3 Summary

This chapter reviewed a number of recently proposed protocols for RFID and briefly presented the state of the art of RFID authentication. The advantages and drawbacks of both symmetric-key-based and public-key-based protocols were summarised in Section 3.1 and Section 3.2 respectively. Symmetric-key-based RFID authentication protocols are vulnerable to tracing attacks and research work towards public-key-based authentication protocols are limited. In the following two chapters of the thesis, we will present our solutions.

Chapter 4

Symmetric-Key-Based Mutual Authentication Scheme for RFID

Passive RFID tags are widely used in supply chains and other systems [Rob05]. Since these tags have a weak computational capability, communications between RFID tags and readers are vulnerable to attacks. A tag could illegally be traced, eavesdropped, blocked and manipulated. There have been a considerable number of research attempts towards practical RFID authentication. Unfortunately, many of them have been proven insecure.

In 2010, Yeh, Wang, Kuo and Wang (YWKW) introduced a mutual authentication protocol [YWKW10], which is an improvement to Chien and Chen's protocol [CC07]. It conforms to EPC Class-1 Generation-2 standards [EPC08] and is claimed to overcome the drawbacks of Chien and Chen's protocol; besides, it reduces the loading of the database.

In this chapter, we investigate the YWKW protocol and show that the YWKW protocol is still vulnerable to several attacks such as the man-in-the-middle attack, and tracing attacks. As a result, an adversary can forge messages and identify past and future interactions of the tag. We propose an improvement to the scheme so that the new scheme can meet the desired security requirements.

We first revisit a previous protocol presented by Yeh *et al.* in Section 4.1. The analysis and attacks of the YWKW scheme are introduced in Section 4.2. A general privacy model is described in detail in Section 4.3. Our improvement to the protocol is presented in Section 4.4. Section 4.5 analyses the security and performance of our protocol. Finally, the conclusion of this chapter is presented in the Section 4.6.

4.1 Revisit of the YWKW Protocol

In 2010, Yeh, Wang, Kuo and Wang presented an RFID mutual authentication scheme [YWKW10] complied with EPC Class-1 Generation-2 specification, which improves Chien and Chen's scheme. This protocol is aimed to prevent replay attacks and denial-of-service attacks, moreover, to provide backward untraceability and low database loading. Before reviewing it, we give all the notations used in the YWKW protocol in Table 4.1.

Table 4.1: Notations of the YWKW protocols

Notation	Interpretation
$\mathcal{T}$	a tag in our protocol
$\mathcal{R}$	a reader in our protocol
EPC	electronic product code
K_i	the authentication key for ith authentication session
K_{old}	the old authentication key stored in the back-end server
K_{new}	the new authentication key stored in the back-end server
P_i	the access key for ith authentication session
P_{old}	the old access key stored in the back-end server
P_{new}	the new access key stored in the back-end server
C_i	the database index for ith authentication session
C_{old}	the old database index stored in the back-end server
C_{new}	the new database index stored in the back-end server
N_T	the nonce generated by the tag
N_R	the nonce generated by the reader
$\oplus$	exclusive or
$PRNG$	pseudorandom number generator
$\mathcal{A} \rightarrow \mathcal{B}$: m	$\mathcal{A}$ sends $\mathcal{B}$ a message m

Now we briefly describe the YWKW protocol.

4.1.1 Initialisation Phase

In the initialisation phase, tags and readers of a protocol instance are set up as follows.

1. A unique initial authentication key K_0 is chosen for a tag. The authentication key is updated in the end of each successful authentication. In the end of $(i+1)th$ authentication phase, the updated new key $K_{i+1} = PRNG(K_i)$, where

$PRNG$ is a pseudorandom bit generator and K_i is the old authentication key used in $(i+1)th$ authentication session.

2. A unique initial access key P_0 is chosen for a tag. The access key is updated in the end of each successful authentication. In the end of the $(i+1)th$ authentication phase, the updated new key $P_{i+1} = PRNG(P_i)$, where P_i is the old access key used in the $(i+1)th$ authentication session.

3. The database index $C_0 = 0$ is assigned for all tags. The database index is used to find corresponding record for the tag efficiently. The initial value 0 means the tag has not been authenticated before. The database index is updated in the end of each successful authentication. In the end of the $(i+1)th$ authentication phase, the updated new index $C_{i+1} = PRNG(N_T \oplus N_R)$, where N_T and N_R are two nonces used in the $(i+1)th$ authentication session.

4. For each tag, the reader stores (1) Electronic Product Code EPC; (2) The old authentication key K_{old}, originally set to K_0; (3) The new authentication key K_{new}, originally set to K_0; (4) The old access key P_{old}, initially set to P_0; (5) The new access key P_{new}, initially set to P_0; (6) The old database index C_{old}; (7) The new database index C_{new}.

5. Every tag stores its own EPC, the initial authentication key K_0, the initial access key P_0 and the initial database index C_0.

4.1.2 The $(i+1)th$ Authentication Phase

The general $(i+1)th$ authentication session is depicted in Figure 4.1. The details are described as follows.

1. $\mathcal{R} \rightarrow \mathcal{T} : N_R$

   The reader chooses a random number N_R and sends it to the tag as a challenge.

2. $\mathcal{T} \rightarrow \mathcal{R} : M_1, D, E, C_i$

   Upon receiving the challenge, the tag randomly chooses a number N_T and

computes

$$M_1 = PRNG(EPC \oplus N_R) \oplus K_i,$$
$$D = N_T \oplus K_i,$$
$$E = N_T \oplus PRNG(C_i \oplus K_i).$$

The tag then sends (M_1, D, E, C_i) as an response to the reader.

3. When the reader receives (M_1, D, E, C_i), it first investigates the value of C_i.

   (a) $C_i = 0$: The reader picks up every record $(EPC, K_{old}, K_{new}, P_{old}, P_{new})$ from the back-end server and computes

   $$I_{old} = M_1 \oplus K_{old},$$
   $$I_{new} = M_1 \oplus K_{new},$$

   The read then checks if either I_{old} or I_{new} equals to $PRNG(EPC \oplus N_R)$. If a match is found, the reader will stop searching the back-end server and set x as *old* or *new* in terms of the comparison result.

   (b) $C_i \neq 0$: The reader searches the back-end server with the database index C_i and gets the corresponding record $(EPC, K_{old}, K_{new}, P_{old}, P_{new}, C_{old}, C_{new})$. Then let $x = old$ or new according to $C_i = C_{old}$ or C_{new}. Then, the reader verifies whether M_1 equals $PRNG(EPC \oplus N_R) \oplus K_x$.

   After the step mentioned above, the reader checks the validity of D and E by examining whether $D \oplus K_x \oplus PRNG(C_x \oplus K_x)$ equals to E.

4. $\mathcal{R} \rightarrow \mathcal{T} : M_2$

   The reader computes $M_2 = PRNG(EPC \oplus N_T) \oplus P_x$ and forwards M_2 to the tag for reader authentication. Then it updates the storing record of the tag, including the authentication key, the access key and the database index.

(a) $x = new$: The reader updates

$$K_{old} = K_{new},$$
$$P_{old} = P_{new},$$
$$C_{old} = C_{new},$$
$$K_{new} = PRNG(K_{new}),$$
$$P_{new} = PRNG(P_{new}),$$
$$C_{new} = PRNG(N_T \oplus N_R),$$

(b) $x = old$: The reader updates $C_{new} = PRNG(N_T \oplus N_R)$.

5. The tag computes $M_2 \oplus P_i$ and checks whether the value equals to $PRNG(EPC \oplus N_T)$. The reader will be authenticated if the result is true, and the tag then updates the shared authentication and access keys as well as the database index

$$K_{i+1} = PRNG(K_i),$$
$$P_{i+1} = PRNG(P_i),$$
$$C_{i+1} = PRNG(N_T \oplus N_R).$$

4.2 Model and Assumptions

RFID systems are typically composed of three main components: tags, readers, and a database. In the security model of the YWKW protocol, tags are assumed to have limited computing power. Readers are connected to the database server via a secure channel, so we will consider the reader and the database server as a single entity.

A tag is only capable of processing the operations of Pseudo-Random Number Generator (PRNG) and Cyclic Redundancy Code (CRC) according to EPC Class-1 Generation-2 RFID specification [EPC08]. We have already described the concept of PRNG in Chapter 2. EPC Class-1 Generation-2 compliant RFID tags have the ability to generate 16-bit PRNG with the following criteria [EPC08]:

- The probability that any 16-bit random number is drawn from the generator shall be bounded by $\frac{0.8}{2^{16}}$ and $\frac{1.25}{2^{16}}$.

Reader		*Tag*
EPC		EPC
K_{old}, K_{new}		K_i
P_{old}, P_{new}		P_i
C_{old}, C_{new}		C_i
Generate N_R		
	$\xrightarrow{N_R}$	
		Generate N_T
		$M_1 = PRNG(EPC \oplus N_R) \oplus K_i$
		$D = N_T \oplus K_i$
		$E = N_T \oplus PRNG(C_i \oplus K_i)$
	$\xleftarrow{M_1,D,E,C_i}$	
if $C_i = 0$:		
$I_{old} = M_1 \oplus K_{old}$		
$I_{new} = M_1 \oplus K_{new}$		
$PRNG(EPC \oplus N_R) \stackrel{?}{=} I_{old}$ or I_{new}		
$x = old$ or new		
else if $C_i \neq 0$:		
$C_i \stackrel{?}{=} C_{old}$ or C_{new}		
$x = old$ or new		
$M_1 \stackrel{?}{=} PRNG(EPC \oplus N_R) \oplus K_x$		
$D \oplus K_x \oplus PRNG(C_x \oplus K_x) \stackrel{?}{=} E$		
$M_2 = PRNG(EPC \oplus N_T) \oplus P_x$	$\xrightarrow{M_2}$	
		$M_2 \oplus P_i \stackrel{?}{=} PRNG(EPC \oplus N_T)$
update		update
if $x = new$:		$K_{i+1} = PRNG(K_i)$
$K_{old} \leftarrow K_{new} \leftarrow PRNG(K_{new})$		$P_{i+1} = PRNG(P_i)$
$P_{old} \leftarrow P_{new} \leftarrow PRNG(P_{new})$		$C_{i+1} = PRNG(N_T \oplus N_R)$
$C_{old} \leftarrow C_{new} \leftarrow PRNG(N_T \oplus N_R)$		
if $x = old$:		
$C_{new} = PRNG(N_T \oplus N_R)$		

Figure 4.1: The $(i+1)th$ authentication phase of the YWKW scheme.

- The probability that any two or more tags among up to $10,000$ tags generate the same sequence of 16-bit random numbers shall not be greater than 0.1%.
- The probability of predicting a 16-bit random number generated by a certain tag shall be less than 0.025% even if the previous outputs from the generator are given.

CRC is a checksum code used to detect errors of the original message. It was invented by Peterson and Brown in 1961 [PB61]. A binary message is represented by a polynomial so that the checksum of this message can be generated by a CRC polynomial using polynomial divisions. The computational procedure can be very efficient by using the logical operation exclusive or (XOR). Thus it is suitable to be implement in low-cost RFID tags. EPC Class-1 Generation-2 standard uses 16-bit CRC polynomial, which is $x^{16} + x^{12} + x^5 + 1$.

An adversary is assumed to have complete control over the communication channel between tags and readers. Namely, it can observe, block and tamper all exchanged messages, and counterfeit new messages.

We now define our attack model and security requirement for RFID system that will be used in the cryptanalysis of the YWKW protocol and our improved protocol.

4.2.1 Attacks on RFID Systems

Definition 4.1 Replay Attack

Replay attack states an adversary maliciously repeats previous communication messages between a tag and a reader to perform a successful authentication.

Definition 4.2 Man-in-the-Middle attack

Man-in-the-middle attack states an adversary inserts or tampers the communication messages sent between a tag and a reader without being detected.

Definition 4.3 Denial-of-Service Attack

Man-in-the-middle attack states an adversary blocks or modifies one or some communication messages sent between a tag and a reader so that the tag and the reader cannot communicate any more.

Definition 4.4 Weak Tracing Attack

Given only the communication records between a tag T_i and a reader R at time t_0, a probabilistic polynomial-time adversary $\mathcal{A}$ identifies T_i using communication

messages sent between T_i and R at time t. That is to say, an attacker performing weak tracing attacks does not compromise the tag.

Definition 4.5 Strong Tracing Attack

Given the internal secret of a tag T_i and the communication records between T_i and a reader R at time t_0, a probabilistic polynomial-time adversary $\mathcal{A}$ identifies T_i using communication messages sent between T_i and R at time t.

It is obvious that a protocol vulnerable to Weak Tracing Attack is also vulnerable to Strong Tracing Attack. We say a tracing attack is a **Backward Tracing Attack** if $\mathcal{A}$ identifies T_i for all the sessions at time t, where $t < t_0$; a tracing attack is a **Forward Tracing Attack** if $\mathcal{A}$ identifies T_i for all the sessions at time t, where $t > t_0$;

4.2.2 Security Requirements

Definition 4.6 Resistance to Replay Attack

An adversary is not able to replay the previous messages and perform a successful authentication, even if he eavesdrops the communications between a tag and a reader.

Definition 4.7 Resistance to Man-in-the-Middle Attack

Given all the messages transferred between a tag and a reader, an adversary is not able to tamper messages that can be accepted by either of the entities.

Definition 4.8 Resistance to Denial-of-Service Attack

A valid tag and a legitimate server can always communicate in a new protocol instance even if an adversary blocks or modifies communication messages transferred between the tag and the reader.

Definition 4.9 Backward Untraceability [1] [OSK03]

Given the internal secret of a tag T_i at time t_0, there is no probabilistic polynomial-time adversary that is able to identify T_i at the time t for all $t < t_0$.

A protocol that achieves Backward Untraceability can resist Strong Backward Tracing attack.

[1] In some papers [OSK03, CC07, DPLK08, YWKW10], backward untraceability is described as forward security. We hereby use the terms backward untraceability defined in [LK06] to distinguish it from forward untraceability.

Definition 4.10 Forward Untraceability [LK06]

Given the internal secret of a tag T_i at time t_0, there is no probabilistic polynomial-time adversary that is able to identify T_i at the time t for all $t > t_0$.

A protocol that achieves Forward Untraceability can resist Strong Forward Tracing attack.

4.3 Analysis of the YWKW Protocol

4.3.1 Man-in-the-Middle Attack

We first show that an adversary can perform a man-in-the-middle attack on the YWKW protocol. Suppose the adversary eavesdrops a valid session π. In this session the tag generates random number N_T, computes $M_1 = PRNG(EPC \oplus N_R) \oplus K_i$, $D = N_T \oplus K_i$ and $E = N_T \oplus PRNG(C_i \oplus K_i)$ after receiving the challenge N_R from the reader. The adversary intercepts the messages (M_1, D, C_i, E) sent by the tag to the reader and calculates a new message (M_1', D', C_i', E'), using the following equations

$$\begin{aligned} M_1' &= M_1, \\ D' &= D \oplus N_A, \\ C_i' &= C_i, \\ E' &= E \oplus N_A, \end{aligned}$$

where N_A is a random number chosen by the adversary. The adversary sends (M_1', D', C_i', E') to the reader. We now prove that the reader will accept the forged message. Since M_1' and C_i' are the same as the valid message, we only need to prove that D' and E' will be accepted by the reader.

After M_1 is verified, the reader obtains $N_T' = D' \oplus K_i$ and calculates

$$\begin{aligned} & N_T' \oplus PRNG(C_i \oplus K_i) \\ &= D' \oplus K_i \oplus PRNG(C_i \oplus K_i) \\ &= D \oplus N_A \oplus K_i \oplus PRNG(C_i \oplus K_i) \\ &= (N_T \oplus PRNG(C_i \oplus K_i)) \oplus N_A \\ &= E \oplus N_A, \end{aligned}$$

which is E'. As a result, the reader accepts the message (M_1', D', C_i', E') manipulated by the adversary.

Note that when the reader replies $M_2 = PRNG(EPC \oplus N_T) \oplus P_i$ to the tag, there is no way that the adversary can forge an acceptable message as it has no knowledge of the tag's secret EPC and P_i. Also N_T is protected by the one-way function $PRNG$. Therefore, the YWKW protocol is vulnerable to the partial man-in-the-middle attack.

4.3.2 Strong Tracing Attacks

Assume an adversary compromises the tag T_x at time t_0 and gets the secret EPC. He also observes the transaction at t_0. Since N_R, M_1, D, E, C_i are public, by examining the relationship between

$$\begin{aligned} M_1 &= PRNG(EPC \oplus N_R) \oplus K_i, \\ D &= N_T \oplus K_i, \\ E &= N_T \oplus PRNG(C_i \oplus K_i), \end{aligned}$$

the adversary can easily calculate

$$M_1 \oplus D \oplus E = PRNG(EPC \oplus N_R) \oplus PRNG(C_i \oplus K_i).$$

Since $K_i = M_1 \oplus PRNG(EPC \oplus N_R)$, we have

$$\begin{aligned} M_1 \oplus D \oplus E \;\; = \;\; & PRNG(EPC \oplus N_R) \\ & \oplus PRNG(C_i \oplus (M_1 \oplus PRNG(EPC \oplus N_R))). \end{aligned}$$

The adversary can obtain this equation with only the long-term secret EPC and other public information by cancelling K_i and N_T. Therefore, the adversary can identify the tag T_x in all the past and future transactions by simply comparing the values of both sides of the equation. Hence, YWKW protocol is vulnerable to both backward and forward strong tracing attacks. Note that this forward tracing attack is more efficient than simply compromising K_{t_0} at time t_0, and computing every K_i using the equation $K_{i+1} = PRNG(K_i)$, where $i > t_0$.

4.4 Our Improved Protocol

In this section we propose an improved protocol in order to get rid of the existing security vulnerabilities of the YWKW protocol.

4.4.1 Notations and Initialisation phase

First, we illustrate the initialisation phase of our scheme, which sets up the system parameters and initial states for both tags and readers. The notations used in our scheme is illustrated in Table 4.2.

Table 4.2: Notations of the our symmetric-key-based protocols

Notation	Interpretation
$\mathcal{T}$	a tag in our protocol
$\mathcal{R}$	a reader in our protocol
EPC	electronic product code
K_i	the secret key for ith authentication session
K_{old}	the old secret key stored in the back-end server
K_{new}	the new secret key stored in the back-end server
C_i	the database index for ith authentication session
C_{old}	the old database index stored in the back-end server
C_{new}	the new database index stored in the back-end server
N_1, N_3	the nonce generated by the reader
N_2	the nonce generated by the tag
$\oplus$	exclusive or
$\|$	concatenation
$PRNG$	pseudorandom number generator
$\mathcal{A} \rightarrow \mathcal{B}$: m	$\mathcal{A}$ sends $\mathcal{B}$ a message m

1. Each tag has a unique initial secret key K_0 saved in itself. Once an authentication session is successfully complete, the secret key will be updated as $K_{i+1} = PRNG(K_i \oplus N_1 \oplus N_2)$, where K_i is the old secret key used in $(i+1)th$ authentication session, N_1 and N_2 are two nonces used in the $(i+1)th$ authentication session..

2. All the tags store $C_0 = 0$ as the database indices to efficiently search records of tags in the database. The database index is updated in the end of each successful authentication. The index C_{i+1} is updated as $C_{i+1} = PRNG(N_2 \oplus N_3)$ in the end of the $(i+1)th$ authentication phase, where N_2 and N_3 are two nonces used in the $(i+1)th$ authentication session.

3. For each tag, the reader stores (1) Electronic Product Code EPC; (2) The old shared secret key K_{old}, originally set to K_0; (3) The new shared key K_{new},

originally set to K_0; (4) The old database index C_{old}; (7) The new database index C_{new}.

4. Every tag stores its own EPC, the initial shared secret key K_0 and the initial database index C_0.

4.4.2 The $(i+1)th$ Authentication Phase

The detail of a general $(i+1)th$ protocol session is illustrated in Figure 4.2.

1. $\mathcal{R} \rightarrow \mathcal{T} : N_1$

   The reader queries the tag with a non-zero nonce N_1, where $N_1 \neq EPC$.

2. $\mathcal{T} \rightarrow \mathcal{R} : M_1, M_2, N_2, C_i$

   Tag rejects the request if $N_1 = 0$ or $N_1 = EPC$; otherwise, the tag randomly generates a number N_2 and computes

   $$M_1 = PRNG(EPC \oplus K_i \oplus N_1 \oplus N_2),$$
   $$M_2 = PRNG(C_i \oplus K_i \oplus N_2).$$

   Then it forwards the message (M_1, M_2, N_2, C_i) to the reader.

3. Upon receiving (M_1, M_2, N_2, C_i), the reader first investigates the value of C_i.

   (a) $C_i = 0$: The reader retrieves the records (EPC, K_{old}, K_{new}) from the back-end server iteratively and computes

   $$I_{old} = PRNG(EPC \oplus K_{old} \oplus N_1 \oplus N_2),$$
   $$I_{new} = PRNG(EPC \oplus K_{new} \oplus N_1 \oplus N_2),$$

   until either I_{old} or I_{new} equals to M_1. Let $x = old$ or new according to the result.

   (b) $C_i \neq 0$: The reader searches the back-end server with the database index C_i and gets the corresponding record $(EPC, K_{old}, K_{new}, C_{old}, C_{new})$. Then let $x = old$ or new depending on whether $C_i = C_{old}$ or C_{new}.

   In addition, the reader verifies the correctness of C_i by examining whether $PRNG(C_x \oplus K_x \oplus N_2) = M_2$.

4. $\mathcal{R} \rightarrow \mathcal{T} : M_3, M_4$

   The reader generates a random number N_3, computers

   $$M_3 = PRNG(CRC(K_x \| N_3)),$$
   $$M_4 = K_x \oplus N_3,$$

   and sends them to the tag. Then it updates the record considering the value of x.

   (a) $x = new$: The reader updates

   $$K_{old} = K_{new},$$
   $$C_{old} = C_{new},$$
   $$K_{new} = PRNG(K_{new} \oplus N_1 \oplus N_2),$$
   $$C_{new} = PRNG(N_2 \oplus N_3).$$

   (b) $x = old$: The reader updates $C_{new} = PRNG(N_2 \oplus N_3)$.

5. The tag recovers the secret N_3 from M_4 using the shared key K_i and checks whether M_3 equals to $PRNG(CRC(K_i \| N_3))$. If the authentication to the reader is complete, the tag will update the shared key and the database index

   $$K_{i+1} = PRNG(K_i \oplus N_1 \oplus N_2),$$
   $$C_{i+1} = PRNG(N_2 \oplus N_3).$$

4.5 Analysis

The mutual authentication YWKW protocol is vulnerable to the man-in-the-middle attack, and strong backward and forward tracing attacks. Our protocol eliminates the vulnerabilities without affecting any other security properties.

4.5.1 Resistance to Replay Attack

Our protocol uses a challenge-response mechanism in the authentication phase. The reader and the tag generate random nonces N_1, N_2 and N_3 in every communication session so that every message is distinctive. Therefore our scheme is resistant to replay attacks.

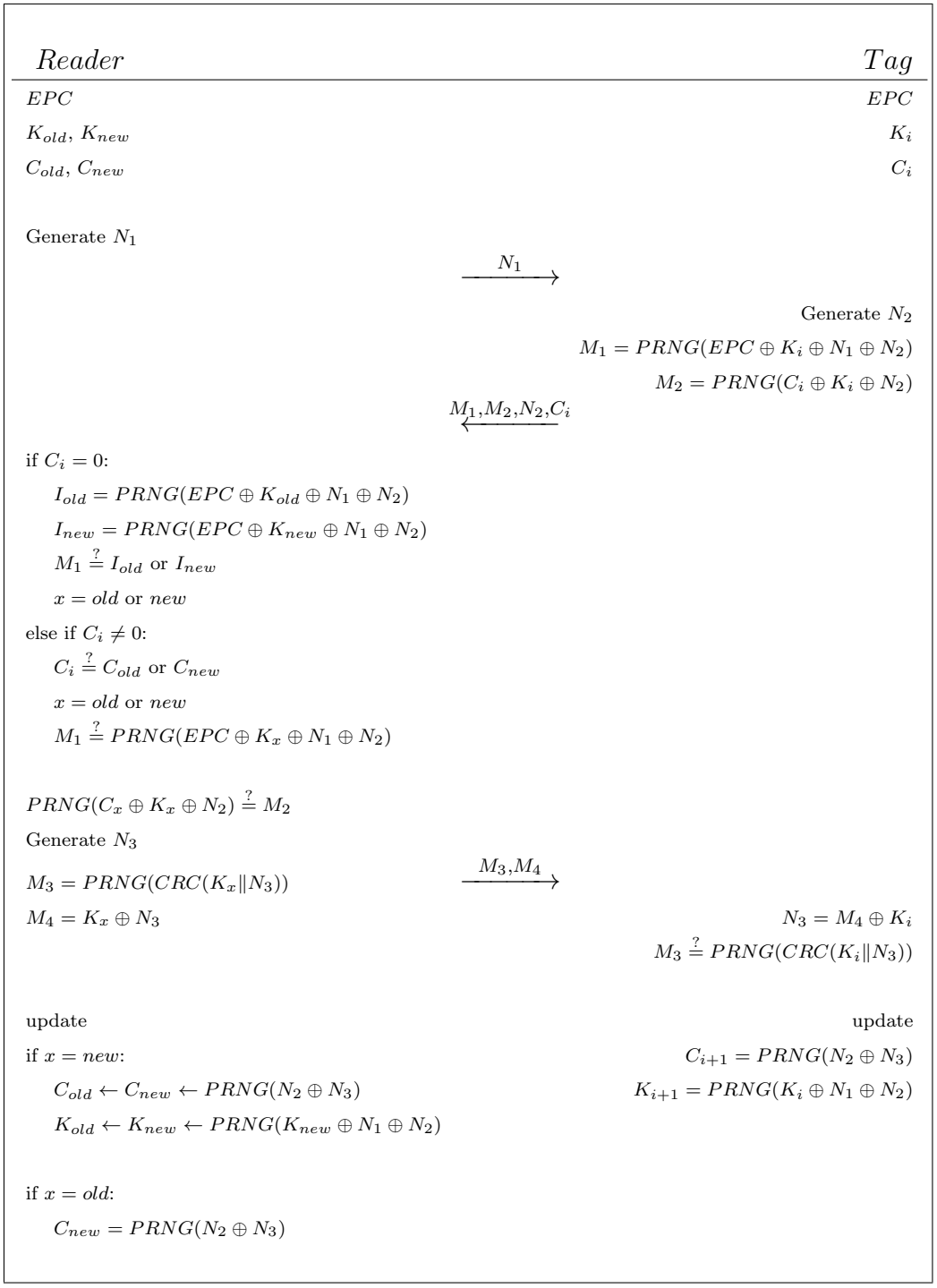

Figure 4.2: The $(i+1)th$ authentication phase of our symmetric-key-based scheme.

4.5.2 Resistance to Man-in-the-Middle Attack

In the YWKW protocol, the tag sends (M_1, D, C_i, E) as a response to the server to authenticate itself. The validity and freshness of this message is dependent on the nonce N_T and the session key K_i. The weakness of the YWKW protocol is that an adversary can construct a valid D and E with a forged nonce N'_T. This is on account of the property of the XOR operation. Our scheme uses $PRNG$ to provide the correctness of the nonces N_1, N_2 so that the adversary cannot counterfeit a message (M_1, M_2, C_i, N_2) that can be accepted by the reader. The security is based on the robustness of the pseudo-random number generator. A PRNG is a deterministic function so that without knowing the secret key of the tag, the adversary cannot use the input to get the same result as the tag. The output generated by the adversary will be rejected by the reader. Therefore, our scheme can prevent the man-in-the-middle attack.

4.5.3 Resistance to Denial-of-Service Attack

The tag updates its secret key K_i and the database index C_i once the reader authentication is complete, i.e., the valid message (M_3, M_4) reaches the tag. A denial-of-service attack will succeed if and only if the message (M_3, M_4) is either blocked or tampered by the adversary. In that case, the reader updates the secret key while the tag does not. Hence, they cannot communicate in the future. However, the reader in our protocol also stores the past secrets K_{old} and C_{old}. In case the tag does not update its secrets, when failing to verify the message using the new key, the reader still will use the old key and index to authenticate the tag if the current information does not work. For this reason, even if the tag does not update its secrets, it can still communicate with the reader in future protocol instance. Thus, our protocol can resist denial-of-service attacks.

4.5.4 Resistance to Tracing Attacks

The traceable drawback of the YWKW protocol is caused by the use of XOR operation. The anonymity of the fixed ID, EPC, relies on the session key K_i. K_i is updated in the end of each session, so that the adversary is unable to identify the message M_1 without the knowledge of the secret K_i. Nonetheless, the adversary can exploit the relationship between the value of M_1, M_2 and form an equation to

identify the tag without the session secret.

In our scheme, an adversary cannot trace a tag in past or future communication even if it compromises the tag to get EPC. These securities property of untraceability is provided by the robustness of pseudo-random number generator.

Suppose the adversary compromise the tag T_0 to acquire knowledge of EPC and the session key K_{t_0} at time t_0. Then, in time t, it eavesdrops a communication session between a tag T_b and the reader and get the public information N_1, N_2, N_3, C_t, M_1, M_2, M_3. We now show that our protocol can achieve backward untraceability, as well as forward untraceability under certain assumptions.

- $t < t_0$

  Because of the correctness of the chosen nonces N_1, N_2, N_3 and the confidentiality of the session key K_t provided by pseudo-random number generator $PRNG$, the adversary is unlikely to calculate or bypass the session key K_t from the equations

  $$\begin{aligned} M_1 &= PRNG(EPC \oplus K_t \oplus N_1 \oplus N_2), \\ M_2 &= PRNG(C_i \oplus K_t \oplus N_2), \\ M_3 &= PRNG(CRC(K_t \| N_3)), \\ M_4 &= K_i \oplus N_3, \end{aligned}$$

  Moreover, K_i is updated based on a one-way hash chain so that the compromise of K_{t_0} will not lead to the leakage of the past session key K_t. Therefore, the fixed ID, EPC, is protected with $PRNG$ and the session key K_t. The adversary cannot generate a identifiable value from M_1, M_2, M_3 without K_t, in past sessions.

- $t > t_0$

  It is difficult to achieve perfect forward untraceability since an adversary can always reckon the secret session key K_t using the equation $K_t = PRNG^n(K_{t_0} \oplus N_1 \oplus N_2)$ once K_{t_0} has been compromised. However, our protocol can still achieve restricted forward untraceability if only the adversary misses one successful authentication session after time t_0. That is because, although N_1 and N_2 are publicly broadcasted, missing one successful authentication session leads the adversary fail to update the obtained session key K_{t_0}. In future

authentication sessions, the tag and the reader will use new shared session keys that the adversary does not have access to. On the other hand, as we already demonstrated above, the fixed identifier EPC is as hard to reveal from the communication messages as past sessions. Therefore, an adversary cannot identify a compromised tag in all the sessions at time t under certain assumptions.

Therefore, our scheme can resist strong tracing attacks and achieve backward untraceability as well as forward untraceability.

4.5.5 Performance

The performance of the YWKW protocol is enhanced with the database index C_i. The index is retained in our protocol. As a result, our protocol does not add the database workload. Moreover, our protocol reduce the number of the keys using in communication sessions, which saves the storage for low-cost RFID tags with limited memory.

4.6 Summary

In this chapter, we analysed a recently proposed mutual authentication for EPC Class-1 Generation-2 RFID systems based on symmetric-key cryptography. We found that the protocol is vulnerable to man-in-the-middle attacks and tracing attacks, which allow an adversary to forge a communication message as well as identify the tag in past and future sessions. We proposed an improved protocol to eliminate the flaws without losing the performance advantage. Our scheme achieves both backward untraceability and forward untraceability. Our new protocol sets a new security benchmark for RFID security.

Chapter 5

Public-Key-Based Authentication Scheme with Untraceability

Currently, as we have already reviewed in Chapter 3, most of the proposed authentication schemes use symmetric-key cryptography such as hash functions and pseudo-random number generators because of the simplicity compared to asymmetric-key cryptography. They all have either security and privacy problems or scalability drawbacks, however. Public-key cryptography can achieve higher security levels compared to symmetric-key cryptography [Vau07], while it has the weakness of higher computational overhead. However, for computationally capable tags, it seems necessary to consider public-key cryptography in RFID systems. In recent years, considerable work has shown that public-key cryptographic techniques can be implemented into low-cost RFID tags [BGK+07, HWF09, LBSV10a, LSBV08, LBSV10b, LBV08, LBV09, MR06, TB06]. RFID tags are able to process modular additions, modular multiplications and elliptic curve scalar multiplications.

In this chapter, we propose a novel RFID authentication scheme with untraceability based on elliptic curve cryptography. We provide rigorous security and privacy proofs for our scheme in the random oracle model and the widely-used Vaudenay's privacy model [Vau07] respectively. It is the first elliptic-curve-based RFID authentication scheme that can achieve both narrow-destructive and wide-forward privacy in the Vaudenay's model. Our scheme is also scalable for a large-scale deployment.

This chapter is organised as follows. We describe the complexity assumptions and privacy model for our scheme in Section 5.1. Our elliptic-curve-based RFID authentication protocol is proposed in Section 5.2. In Section 5.3, we analyse our scheme and prove its security and privacy. Section 5.4 summarises this chapter.

5.1 Preliminaries

In this section, we briefly describe the complexity assumptions and the privacy model that will be used throughout this chapter.

5.1.1 Complexity Assumptions

Let $\mathbb{G}$ be a cyclic additive group with order p, where p is a prime. P is a generator of $\mathbb{G}$.

Definition 5.1 Computational Diffie-Hellman (CDH) Assumption

The CDH problem is ϵ-hard, if given a tuple (P, aP, bP) for a, $b \in_R \mathbb{Z}_p^*$, there is no probabilistic polynomial-time adversary $\mathcal{A}$ that can solve the CDH problem with a probability $Succ_{CDH} > \epsilon$.

Definition 5.2 Decisional Diffie-Hellman (DDH) Oracle

The DDH oracle is a probabilistic polynomial-time algorithm that solves DDH problem on $\mathbb{G}$. Given an tuple (P, aP, bP) and $C \in \mathbb{G}$ as an input, it replies whether $C = abP$. The output is true if $C = abP$; otherwise the output is false.

Definition 5.3 Gap Diffie-Hellman (GDH) Assumption

The GDH problem is ϵ-hard, if given a tuple (P, aP, bP) for a, $b \in_R \mathbb{Z}_p^*$ and the DDH oracle $\mathcal{O}_{\mathcal{DDH}}$, there is no probabilistic polynomial-time adversary $\mathcal{A}$ that can solve the CDH problem with a probability $Succ_{GDH} > \epsilon$.

5.1.2 Vaudenay's Privacy Model

We use the Vaudenay's privacy model [Vau07] to analyse our scheme throughout this chapter. In this section, we give a detailed description of the Vaudenay's model.

Adversary

In the Vaudenay's model, a tag is either *free* or *drawn* in this model. A drawn tag is available for an adversary to interact while a free tag cannot be reached by an adversary. Every tag has a session handle in a communication session called *virtual ID*.

An adversary is defined as eight oracles.

- $\mathsf{CreateTag}^{\mathsf{b}}(\mathsf{ID})$ creates a free tag with an identity ID and a bit b. The tag is legitimate if $b = 1$ and not legitimate if $b = 0$;

- $\mathsf{DrawTag}() \rightarrow (\mathsf{vtag}_1, \mathsf{b}_1, \ldots, \mathsf{vtag}_n, \mathsf{b}_n)$ makes several free tags become drawn. $vtag$ is the virtual ID of a tag and b indicates whether the tag is legitimate or not;

- $\mathsf{Free}(\mathsf{vtag})$ makes a drawn become free, which means an adversary can no longer access the tag with a virtual ID $vtag$.

- $\mathsf{Launch} \rightarrow \pi$ starts a session π of the protocol;

- $\mathsf{SendTag}(\mathsf{m}, \mathsf{vtag}) \rightarrow \mathsf{m}'$ sends the message m to the virtual tag $vtag$ and returns a message m';

- $\mathsf{SendReader}(\mathsf{m}) \rightarrow \mathsf{m}'$ sends the message m to the reader and returns m';

- $\mathsf{Result}(\pi) \rightarrow \mathsf{b}$ returns a bit b to indicate whether the protocol session π is completed successfully or not;

- $\mathsf{Corrupt}(\mathsf{vtag}) \rightarrow \mathsf{S}$ returns the secret state S of the tag.

An adversary is classified into eight classes: Wide-Weak, Narrow-Weak , Wide-Forward, Narrow-Forward, Wide-Destructive, Narrow-Destructive, Wide-Strong, Narrow-Strong.

A Wide-Weak adversary is allowed to access all the oracles except Corrupt; a Wide-Forward adversary can access no oracles but Corrupt oracle once Corrupt has been accessed for the first time; a Wide-Destructive adversary will not access a tag any more if it has Corrupted the tag; a Wide-Strong adversary is the strongest adversary that is allowed to access any oracles at any time. Every Wide adversary also has a corresponding Narrow adversary, who can query the same oracles except the Result oracle.

An adversary wins the privacy game if it recognises the real ID of a virtual tag after accessing all the allowed oracles.

Blinder

Blinder is also introduced to simulate protocol messages for adversaries. Specifically, it can only observe all the communication messages and simulate Launch, SendTag, SendReader and Result oracles to the adversary. Note that a blinder can never corrupt any tag hence it has no access to the secret state of a tag.

An adversary $\mathcal{A}$ with the present of a blinder B is denoted by $\mathcal{A}^B$. $\mathcal{A}$ is a trivial adversary if $\left|\Pr[\mathcal{A}\ wins] - \Pr[\mathcal{A}^B\ wins]\right| < \epsilon$, where ϵ is a negligible number. Informally, an adversary is trivial if it does not use the communication messages to compromise an RFID system; that is to say, the adversary cannot distinguish between the real communication and the messages simulated by a blinder.

Privacy

If an RFID system is secure against a class of adversaries, i.e. all the adversaries from the some class are trivial, the system achieves the corresponding privacy class. The links between these privacy classes are shown below. We refer the reader to [Vau07] for more details.

$$\begin{array}{ccccccc} \text{Wide-Strong} & \Rightarrow & \text{Wide-Destructive} & \Rightarrow & \text{Wide-Forward} & \Rightarrow & \text{Wide-Weak} \\ \Downarrow & & \Downarrow & & \Downarrow & & \Downarrow \\ \text{Narrow-Strong} & \Rightarrow & \text{Narrow-Destructive} & \Rightarrow & \text{Narrow-Forward} & \Rightarrow & \text{Narrow-Weak} \end{array}$$

5.2 Our authentication scheme

We propose an RFID authentication protocol based on elliptic curve cryptography, which is shown in Figure 5.1. The steps of our elliptic-curve-based RFID authentication protocol is illustrated below.

5.2.1 Notations and Initialisation

Let E be an elliptic curve defined over a field $\mathbb{Z}_p^*$, where p is an n-bit prime number. Assume a point P is a generator of $\mathbb{G}$, which is the group of points on the elliptic cure E. Let x and y denote the private keys of the tag and the reader, respectively.

$X = xP$ and $Y = yP$ denote the corresponding public keys. $H : \mathbb{G} \times \mathbb{G} \to \{0,1\}^k$ is a collision-resistant hash function from $\mathbb{G} \times \mathbb{G}$ to an k-bit number.

Table 5.1: Notations of the our public-key-based protocols

Notation	Interpretation
$\mathcal{T}$	a tag in our protocol
$\mathcal{R}$	a reader in our protocol
p	a prime
E	an elliptic curve over the finite field $\mathbb{Z}_p^*$
$\mathbb{G}$	the cyclic additive group of the points on E
P	a generator of $\mathbb{G}$
x	the private key of the tag
X	the public key of the tag
y	the private key of the reader
Y	the public key of the reader
$H : \mathbb{G} \times \mathbb{G} \to \{0,1\}^k$	a cryptographic hash function mapping two points of E to a k-bit binary string
$\mathcal{A} \to \mathcal{B}$: m	$\mathcal{A}$ sends $\mathcal{B}$ a message m

Initially, every tag stores its own private key x and the public key Y of the reader; the reader stores its private key y and each tag's public key X. The prime p, the elliptic curve E and the hash function H are system parameters shared between tags and the reader.

5.2.2 Authentication Phase

Our scheme is a typical challenge-response authentication protocol with steps as follows.

1. $\mathcal{R} \to \mathcal{T} : A$
   The reader randomly chooses a number $r_s \in \mathbb{Z}_p^*$ and sends the challenge $A = r_s P$ to the tag.

2. $\mathcal{T} \to \mathcal{R} : T, S, v$
   Upon receiving A, the tag randomly chooses $r_t \in \mathbb{Z}_p^*$ and sends $T = r_t Y$, $S = xA + r_t P$ and $v = H(xY, S)$ as a response to the reader.

3. The reader computes $X' = r_s^{-1}(S - y^{-1}T)$ and looks for X' in the database. If X' is in the database, the reader calculates the value of $H(yX', S)$ and

compares it to v. If the result is equal, the tag is authenticated as a legitimate one.

The detail of our protocol is illustrated in Figure 5.1.

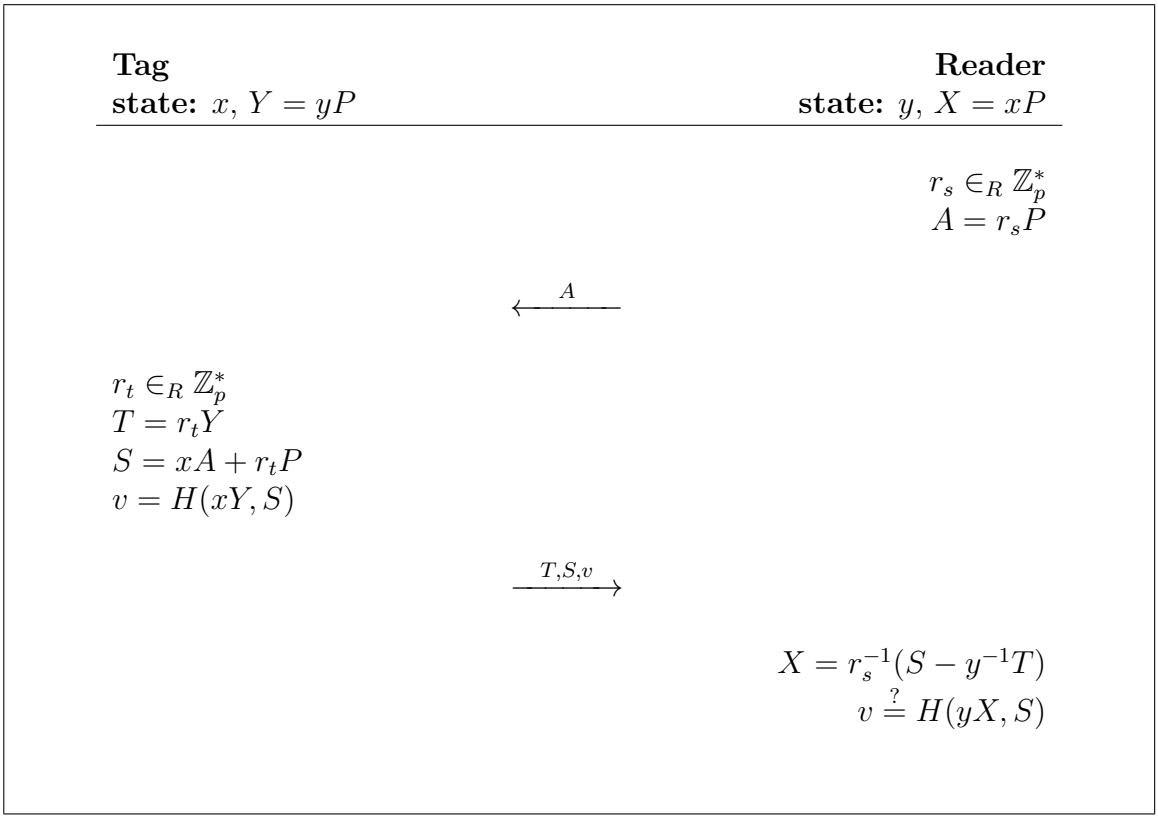

Figure 5.1: Our public-key-based authentication protocol flow

5.3 Protocol analysis

We analyse our scheme in three steps. Firstly we prove the correctness of our scheme; secondly we investigate the security of our scheme; finally we classify the privacy level of our scheme in the Vaudenay's privacy model [Vau07] .

5.3.1 Correctness

Our proposed scheme is correct because the following equations hold.

$$\begin{aligned} X &= r_s^{-1}(S - y^{-1}T) \\ &= r_s^{-1}(xA + r_tP - y^{-1}r_tY) \\ &= r_s^{-1}(xr_sP + r_tP - y^{-1}r_tyP) \\ &= r_s^{-1}xr_sP \\ &= xP, \end{aligned}$$

and

$$\begin{aligned} v &= H(xY, S) \\ &= H(yX, S). \end{aligned}$$

Since the reader can directly recover the public key from S and T and use the key as the tag's ID, it is not possible that the tag is identified as a different one. It is only possible that given S, S', where $S \neq S'$, $H(xY, S) = H(yX, S')$, which occurs with the probability 2^{-n} under the random oracle model and is negligible.

5.3.2 Security

Security refers to soundness and unforgeability of a scheme. That is to say, an adversary $\mathcal{A}$ cannot impersonate a legitimate tag and be authenticated by a reader without corrupting tags.

Theorem 5.1 *If the GDH problem is ϵ-hard in a cyclic additive group $\mathbb{G}$, then our scheme is secure in the random oracle model.*

Proof. In our scheme, T is used for looking up RFID tags only. A proof of a legitimate tag fully depends on the validity of v, which has no connection with T. Hence, w.l.o.g., we do not consider T and the lookup process in our proof for convenience.

Assume $\mathcal{A}$ is an adversary against our scheme with success probability $Succ_0$. He makes q_H queries to $H : \mathbb{G} \times \mathbb{G} \to \{0,1\}^k$, q_T queries to SendTag oracles and q_R queries to SendReader oracles. Since the security proof consider the scenario with only one tag and one reader in a protocol instance. W.l.o.g. the input, vtag, of the SendTag oracle can be omitted. Moreover, note that in our scheme, SendReader

oracle will not return any message that is meant to send to the tag. So we let the returning m' be the result x of Result oracle to represent the result of authentication.

Suppose a simulator $\mathcal{B}$ simulates all the oracles to answer $\mathcal{A}$'s queries. $\mathcal{B}$ maintains a list called H-List to record the inputs of the hash function and the corresponding values in the simulation. $\mathcal{B}$ sets $X = xP$ and $Y = yP$ where (P, xP, yP) is the tuple of the GDH assumption defined in Section 5.1. We consider the cryptographic hash function H as the random oracle H.

Hash Queries: In this game, $\mathcal{B}$ simulates the random oracle H to $\mathcal{A}$. To answer the queries, $\mathcal{B}$ maintains a list called H-List to record the hash queries and the corresponding values. The list is composed of the tuples (K_i, S_i, v_i, b_i), where K_i and S_i represent the input of the hash function, v_i represents the output of the hash function, $b_i = 1$ if $K_i = xyP$ and $b_i = 0$ if $K_i \neq xyP$. The list is initially empty. When $\mathcal{A}$ queries the oracle H with (K_i, S_i), $\mathcal{B}$ will use the DDH oracle to determine whether $K_i = xyP$.

1. $K_i = xyP$, $\mathcal{B}$ checks the H-List.

   - If there exists a tuple $(\perp, S_i, v_i, 1)$ in the list (this can be added in SendTag queries), $\mathcal{B}$ returns v_i to $\mathcal{A}$;
   - Otherwise, $\mathcal{B}$ chooses $v_i \in_R \mathbb{Z}_p$ such that there is no tuple $(\cdot, \cdot, v_i, \cdot)$ in the H-list, and adds $(K_i, S_i, v_i, 1)$ to the list. $\mathcal{B}$ then returns v_i to $\mathcal{A}$. Here, $\perp$ denotes the simulator $\mathcal{B}$ has no idea of the value.

2. $K_i \neq xyP$, $\mathcal{B}$ chooses $v_i \in_R \mathbb{Z}_p$ such that there is no tuple $(\cdot, \cdot, v_i, \cdot)$ in the H-list and adds $(K_i, S_i, v_i, 0)$ to the list. $\mathcal{B}$ then outputs v_i to $\mathcal{A}$.

SendTag Queries: In this game, $\mathcal{B}$ simulates the SendTag oracle to $\mathcal{A}$. Upon receiving the query A_i from the adversary $\mathcal{A}$, $\mathcal{B}$ chooses $r_i \in_R \mathbb{Z}_p$ and calculates $S_i = r_iP$ (Note that r_iP and $xA + r_iP$ are indistinguishable to $\mathcal{A}$). $\mathcal{B}$ then checks the List.

1. If there exists a tuple $(\cdot, S_i, v_i, 1)$ in the H-list, $\mathcal{B}$ answers $\mathcal{A}$'s query with (S_i, v_i).

2. Otherwise, $\mathcal{B}$ picks up $v_i \in_R \mathbb{Z}_p$ such that there does not exist a tuple $(\cdot, \cdot, v_i, \cdot)$ in the H-list, and adds $(\perp, S_i, v_i, 1)$ to the list. $\mathcal{B}$ then answers (S_i, v_i) to $\mathcal{A}$'s

queries.

SendReader Queries: In this game, $\mathcal{B}$ simulates the SendReader oracle for the adversary $\mathcal{A}$. Upon receiving the request (S_i, v_i), $\mathcal{B}$ checks the H-List.

1. If there exists a tuple $(\cdot, S_i, v_i, 1)$ in the H-list, $\mathcal{B}$ accepts it and outputs 1.

2. Otherwise, $\mathcal{B}$ sends 0 to rejects the request.

This simulation is perfect except (S_i, v_i) is valid but v_i is not obtained from the oracle H. Since H is uniformly distributed, this case only occurs with the probability less than $\frac{q_V}{2^k - q_H - q_S}$, which is negligible. Therefore, $\epsilon \geq Succ_0 - \frac{q_V}{2^k - q_H - q_S}$. □

5.3.3 Narrow-Destructive and Wide-Forward privacy

Privacy addresses the resistance against unauthorised identification or tracking tags. Hereby, we prove that all of the Narrow-Destructive adversaries against our scheme are trivial. Then we deduce Wide-Forward privacy of our scheme based on that result.

Theorem 5.2 *Assume $\frac{q}{2^k - q}$ is negligible, then our scheme can achieve Narrow-Destructive and Wide-Forward privacy in the Vaudenay's privacy model.*

Proof. Firstly, we will prove our scheme is Narrow-Destructive private in the Vaudenay's Model. It is obvious that all Launch and SendReader queries does not need to be simulated since a narrow-destructive adversary is not allowed to access Result query. We now prove that for any narrow-destructive adversary $\mathcal{A}$, there exists a blinder B such that $\left|\Pr[\mathcal{A}\ wins] - \Pr[\mathcal{A}^B\ wins]\right| < \epsilon$, where ϵ is negligible.

Let $\mathcal{B}$ be a simulator simulating all the oracles for $\mathcal{A}$ and $\mathcal{A}^B$. Suppose $\mathcal{B}$ maintains a list recording the inputs and outputs of the random oracle H, denoted by H-List. The H-List contains the items (A, K, h), where A and K are the inputs to the random oracle H and h is the corresponding hash values. $\mathcal{B}$ also maintains a list for the SendTag oracle, called S-List. The S-List contains the items $(A, vtag, v)$, where A and $vtag$ are the inputs to the SendTag oracle, and v is the corresponding output.

Since the blinder B has no knowledge of the secret state of $vtag$, B will return a randomly chosen c to simulate SendTag oracles to $\mathcal{A}^B$. Suppose B maintains a list, denoted by S'-List, which contains the items $(A, vtag, c')$, where A and $vtag$ are the inputs and c' is the output.

Let E denote the event that there exists at least one A that is in both H-list and S-list, which means A is sent to $\mathcal{A}$ for both the hash queries H and SendTag query. Let E' denote the event that there exists at least one A in both H-list and S'-list, which means A is sent to $\mathcal{A}$, for the hash queries H, and B, for the SendTag query as an input.

Note that $\mathcal{A}$ is a narrow-destructive adversary, so $\mathcal{A}$ cannot query corrupted tags. As a result, $\mathcal{A}$ cannot distinguish between v, c and c'. Hence the simulation is perfect as long as E and E' never happen, i.e. $\Pr[\mathcal{A}\ wins|\neg E] = \Pr[\mathcal{A}^B\ wins|\neg E']$ and $\Pr[E] = \Pr[E']$. Thus we have $\left|\Pr[\mathcal{A}\ wins] - \Pr[\mathcal{A}^B\ wins]\right| < \Pr[E]$.

Suppose $\mathcal{A}$ makes q queries to the random oracle H. Since H is uniformly distributed, the probability of E occurring is $\frac{q}{2^k - q}$, which is negligible. Our scheme is Narrow-Destructive private. Clearly, narrow-destructive privacy implies narrow-forward privacy. Therefore, our scheme is also Wide-Forward private following the Vaudenay's Lemma 8 [Vau07]. □

5.3.4 Performance

Our scheme is scalable because the reader can compute the tag's public key and use it as the ID to efficiently look up the tag in the back-end database. On the other hand, a tag computes only four scalar multiplications and one points addition on the elliptic curve in our scheme. Therefore, our scheme is low-cost and suitable for light-weight RFID tags. The comparisons among related RFID authentication schemes are shown in Table 5.2.

5.4 Summary

In this chapter, we presented a novel RFID authentication protocol for low-cost RFID tags.

We first defined the complexity assumptions and reviewed the privacy definition of the Vaudenay's privacy model. We proposed an tag-authentication protocol using public-key cryptography on elliptic curve. We then proved our scheme is secure based

Table 5.2: Comparisons among related public-key-based RFID authentication schemes

	Unforgeability	Privacy-Preserved
Schnorr	Yes	No
Okamoto	Yes	No
EC-RAC	No	Narrow-Weak
Randomized Schnorr	Yes	Narrow-Weak
EC-RAC II	No	Narrow-Weak
EC-RAC III	No	Narrow-Weak
EC-RAC IV	No	Narrow-Weak
Proposed scheme	Yes	Narrow-Destruction and Wide-Forward

on the Gap Diffie-Hellman problem in the random oracle model. Our scheme is also privacy-preserved in the Vaudenay's model. We compared our scheme with other public-key-based RFID authentication protocol and showed that our scheme is the first elliptic-curve-based RFID authentication scheme that achieves both narrow-destructive and wide-forward privacy. Our scheme is also scalable for large-scale deployment.

Chapter 6

Conclusion

This chapter concludes the thesis from two aspects. The contributions of our work are emphasised in Section 6.1. Section 6.2 indicates the possible directions for RFID security.

6.1 Contributions

This thesis concentrates on the security and privacy issues for RFID authentication protocols. We reviewed related work and presented two RFID authentication schemes using symmetric-key cryptography and public-key cryptography respectively.

The major contributions of this thesis can be summarised as follows.

- In Chapter 3, we gave an analysis of the YWKW protocol, which is a recently proposed EPC Class-1 Generation-2 compliant authentication protocol. We showed that it is vulnerable to some attacks that may cause privacy compromised, such as man-in-the-middle attacks and strong tracing attacks. We proposed an improved scheme based on the EPC Class-1 Generation-2 specification as well. Our scheme overcomes all the drawbacks without compromise the performance advantage of the YWKW protocol. In addition, the proposed scheme achieves both backward untraceability and forward untraceability.

- In Chapter 4, we proposed a novel tag-authentication scheme constructed on elliptic curve cryptography. It is, to our knowledge, the first elliptic-curve based RFID authentication scheme that achieves both narrow-destructive and wide-forward privacy. The security of our proposed protocol can be deduced to the hardness of Gap Diffie-Hellman problem in the random oracle model. We

proved the privacy of our scheme in the Vaudenay's privacy model. We compared our scheme with other related public-key based authentication protocols for RFID. Since it is efficient for a back-end server to look up the information of a tag, our scheme is scalable for large-scale RFID systems.

6.2 Future Work

Although a considerable amount of research have focused on RFID security, there still are some research topics that were not mentioned in this thesis. Further study is needed for them.

- Currently all the public-key based protocols are designed for tag authentication. However, readers need to be authenticated in the communication as well. Applying public-key cryptography in mutual authentication protocols should be studied further.

- Juels introduced the concept of yoking-proof for RFID in 2004 [Jue04], which was extended to grouping-proof by Saito and Sakurai in 2005 [SS05]. It is proposed to address the problem of scanning multiple tags simultaneously by generating a co-existence proof of them. Very little work focuses on this topic, thus further research would be interesting.

- Tag ownership transfer is another open topic proposed by Molnar, Soppera and Wagner in 2006 [MSW06]. An RFID tag may be transferred to another owner in real-world applications, for example, supply chain. How to pass the information of tags from old owners to new owners without compromising the privacy of tags and both owners is an interesting work for further research.

Bibliography

[AC01] Allianz Canada encourages customers to fight auto theft. July 2001. http://www.insurance-canada.ca/market/canada/Allianz200107.php.

[ADO06] Gildas Avoine, Etienne Dysli, and Philippe Oechslin. Reducing time complexity in RFID systems. In Bart Preneel and Stafford Tavares, editors, *Selected Areas in Cryptography*, volume 3897 of *Lecture Notes in Computer Science*, pages 291–306. Springer Berlin / Heidelberg, 2006.

[AO05] G. Avoine and P. Oechslin. A scalable and provably secure hash-based rfid protocol. In *Pervasive Computing and Communications Workshops, 2005. PerCom 2005 Workshops. Third IEEE International Conference on*, pages 110 – 114, Mar. 2005.

[BCI08] Julien Bringer, Herv Chabanne, and Thomas Icart. Cryptanalysis of ec-rac, a rfid identification protocol. In Matthew Franklin, Lucas Hui, and Duncan Wong, editors, *Cryptology and Network Security*, volume 5339 of *Lecture Notes in Computer Science*, pages 149–161. Springer Berlin / Heidelberg, 2008.

[BdM08] Mike Burmester and Breno de Medeiros. The security of EPC Gen2 compliant RFID protocols. In *Proceedings of the 6th international conference on Applied cryptography and network security*, ACNS'08, pages 490–506, Berlin, Heidelberg, 2008. Springer-Verlag.

[BGK+07] L. Batina, J. Guajardo, T. Kerins, N. Mentens, P. Tuyls, and I. Verbauwhede. Public-key cryptography for RFID-tags. In *Proceedings*

of the Fifth IEEE International Conference on Pervasive Computing and Communications Workshops, PERCOMW '07, pages 217–222, Washington, DC, USA, 2007. IEEE Computer Society.

[Bon98] Dan Boneh. The decision diffie-hellman problem. In Joe Buhler, editor, *Algorithmic Number Theory*, volume 1423 of *Lecture Notes in Computer Science*, pages 48–63. Springer Berlin / Heidelberg, 1998.

[BR93] Mihir Bellare and Phillip Rogaway. Random oracles are practical: a paradigm for designing efficient protocols. In *Proceedings of the 1st ACM conference on Computer and communications security*, CCS '93, pages 62–73, New York, NY, USA, 1993. ACM.

[CC07] Hung-Yu Chien and Che-Hao Chen. Mutual authentication protocol for RFID conforming to EPC Class 1 Generation 2 standards. *Computer Standards & Interfaces*, 29(2):254 – 259, 2007.

[CD09] Chin-Ling Chen and Yong-Yuan Deng. Conformation of epc class 1 generation 2 standards rfid system with mutual authentication and privacy protection. *Engineering Applications of Artificial Intelligence*, 22(8):1284 – 1291, 2009.

[CLLD09] Shaoying Cai, Yingjiu Li, Tieyan Li, and Robert H. Deng. Attacks and improvements to an rifd mutual authentication protocol and its extensions. In *Proceedings of the second ACM conference on Wireless network security*, WiSec '09, pages 51–58, New York, NY, USA, 2009. ACM.

[Cop94] D. Coppersmith. The data encryption standard (DES) and its strength against attacks. *IBM Journal of Research and Development*, 38(3):243 –250, May 1994.

[COS86] Don Coppersmith, Andrew Odlzyko, and Richard Schroeppel. Discrete logarithms in GF(p). *Algorithmica*, 1:1–15, 1986.

[CS00] Ronald Cramer and Victor Shoup. Signature schemes based on the strong rsa assumption. *ACM Trans. Inf. Syst. Secur.*, 3:161–185, August 2000.

[CvA90] David Chaum and Hans van Antwerpen. Undeniable signatures. In Gilles Brassard, editor, *Advances in Cryptology – CRYPTO – 89 Proceedings*, volume 435 of *Lecture Notes in Computer Science*, pages 212–216. Springer Berlin / Heidelberg, 1990.

[DH76a] W. Diffie and M. Hellman. New directions in cryptography. *Information Theory, IEEE Transactions on*, 22(6):644 – 654, Nov. 1976.

[DH76b] Whitfield Diffie and Martin E. Hellman. Multiuser cryptographic techniques. In *Proceedings of the June 7-10, 1976, national computer conference and exposition*, AFIPS '76, pages 109–112, New York, NY, USA, 1976. ACM.

[Dim05] Tassos Dimitriou. A lightweight RFID protocol to protect against traceability and cloning attacks. In *Proceedings of the First International Conference on Security and Privacy for Emerging Areas in Communications Networks*, pages 59–66, Washington, DC, USA, 2005. IEEE Computer Society.

[DLYZ10] Robert Deng, Yingjiu Li, Moti Yung, and Yunlei Zhao. A new framework for RFID privacy. In Dimitris Gritzalis, Bart Preneel, and Marianthi Theoharidou, editors, *Computer Security – ESORICS 2010*, volume 6345 of *Lecture Notes in Computer Science*, pages 1–18. Springer Berlin / Heidelberg, 2010.

[DOD03] U.S. military to issue RFID mandate. *RFID Journal,*, Sept. 2003. http://www.rfidjournal.com/article/articleview/576/1/1.

[DPLK08] Dang Nguyen Duc, Jaemin Park, Hyunrok Lee, and Kwangjo Kim. Enhancing security of Class i Generation 2 RFID against traceability and cloning. In Damith C. Ranasinghe and Peter H. Cole, editors, *Networked RFID Systems and Lightweight Cryptography*, pages 269–277. Springer Berlin Heidelberg, 2008.

[DR00] Joan Daemen and Vincent Rijmen. The block cipher rijndael. In Jean-Jacques Quisquater and Bruce Schneier, editors, *Smart Card Research and Applications*, Lecture Notes in Computer Science, pages 277–284. Springer Berlin / Heidelberg, 2000.

[ElG85] Taher ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. In George Blakley and David Chaum, editors, *Advances in Cryptology*, volume 196 of *Lecture Notes in Computer Science*, pages 10–18. Springer Berlin / Heidelberg, 1985.

[EPC08] EPC^{TM} Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860 MHz - 960 MHz Version 1.2.0, Oct 2008. http://www.gs1.org/gsmp/kc/epcglobal/uhfc1g2/uhfc1g2_1_2_0-standard-20080511.pdf.

[FHV10] Junfeng Fan, Jens Hermans, and Frederik Vercauteren. On the claimed privacy of ec-rac iii. In Siddika Ors Yalcin, editor, *Radio Frequency Identification: Security and Privacy Issues*, volume 6370 of *Lecture Notes in Computer Science*, pages 66–74. Springer Berlin / Heidelberg, 2010.

[Fin03] K. Finkenzeller. *RFID Handbook: Fundamentals and Applications in Contactless Smart Cards and Identification*. Wiley, 2003.

[GB06] Bill Glover and Himanshu Bhatt. *RFID Essentials (Theory in Practice (O'Reilly))*. O'Reilly Media, Inc., 2006.

[GJP05] S.L. Garfinkel, A. Juels, and R. Pappu. RFID privacy: an overview of problems and proposed solutions. *Security Privacy, IEEE*, 3(3):34 – 43, May-June 2005.

[HM04] D. Henrici and P. Müller. Hash-based enhancement of location privacy for radio-frequency identification devices using varying identifiers. *Pervasive Computing and Communications Workshops, IEEE International Conference on*, 0:149, 2004.

[HMZH08] Junghoon Ha, Sangjae Moon, Jianying Zhou, and Jaecheol Ha. A new formal proof model for RFID location privacy. In *Proceedings of the 13th European Symposium on Research in Computer Security: Computer Security*, ESORICS '08, pages 267–281, Berlin, Heidelberg, 2008. Springer-Verlag.

[HWF09] Daniel Hein, Johannes Wolkerstorfer, and Norbert Felber. ECC is ready for RFID – a proof in silicon. In Roberto Avanzi, Liam Keliher, and Francesco Sica, editors, *Selected Areas in Cryptography*, volume 5381 of *Lecture Notes in Computer Science*, pages 401–413. Springer Berlin / Heidelberg, 2009.

[JMW05] Ari Juels, David Molnar, and David Wagner. Security and privacy issues in e-passports. In *Proceedings of the First International Conference on Security and Privacy for Emerging Areas in Communications Networks*, pages 74–88, Washington, DC, USA, 2005. IEEE Computer Society.

[Jue04] A. Juels. "yoking-proofs" for RFID tags. In *Pervasive Computing and Communications Workshops, 2004. Proceedings of the Second IEEE Annual Conference on*, pages 138 – 143, march 2004.

[Jue05] Ari Juels. Strengthening EPC tags against cloning. In *Proceedings of the 4th ACM workshop on Wireless security*, WiSe '05, pages 67–76, New York, NY, USA, 2005. ACM.

[Jue06] A. Juels. RFID security and privacy: a research survey. *Selected Areas in Communications, IEEE Journal on*, 24(2):381 – 394, Feb. 2006.

[JW07] Ari Juels and Stephen A. Weis. Defining strong privacy for RFID. In *Pervasive Computing and Communications Workshops, 2007. PerCom Workshops '07. Fifth Annual IEEE International Conference on*, pages 342 –347, Mar. 2007.

[KN05] Sindhu Karthikeyan and Mikhail Nesterenko. RFID security without extensive cryptography. In *Proceedings of the 3rd ACM workshop on Security of ad hoc and sensor networks*, SASN '05, pages 63–67, New York, NY, USA, 2005. ACM.

[KS09] Florian Kerschbaum and Alessandro Sorniotti. RFID-based supply chain partner authentication and key agreement. In *Proceedings of the second ACM conference on Wireless network security*, WiSec '09, pages 41–50, New York, NY, USA, 2009. ACM.

[Lau07] Adam Laurie. Practical attacks against RFID. *Network Security*, 2007(9):4 – 7, 2007.

[LBSV10a] YK Lee, L Batina, D Singelée, and I Verbauwhede. Wide–weak privacy–preserving authentication protocols. In *The 2nd International Conference on Mobile Lightweight Wireless Systems, Mobilight*. Springer-Verlag, 2010.

[LBSV10b] Yong Ki Lee, Lejla Batina, Dave Singelée, and Ingrid Verbauwhede. Low-cost untraceable authentication protocols for RFID. In *Proceedings of the third ACM conference on Wireless network security*, WiSec '10, pages 55–64, New York, NY, USA, 2010. ACM.

[LBV08] Yong Ki Lee, L. Batina, and I. Verbauwhede. EC-RAC (ECDLP based randomized access control): Provably secure rfid authentication protocol. In *RFID, 2008 IEEE International Conference on*, pages 97 –104, April 2008.

[LBV09] Yong Ki Lee, L. Batina, and I. Verbauwhede. Untraceable RFID authentication protocols: Revision of EC-RAC. In *RFID, 2009 IEEE International Conference on*, pages 178 –185, April 2009.

[LD07] Yingjiu Li and Xuhua Ding. Protecting RFID communications in supply chains. In *Proceedings of the 2nd ACM symposium on Information, computer and communications security*, ASIACCS '07, pages 234–241, New York, NY, USA, 2007. ACM.

[LK06] Chae Lim and Taekyoung Kwon. Strong and robust RFID authentication enabling perfect ownership transfer. In Peng Ning, Sihan Qing, and Ninghui Li, editors, *Information and Communications Security*, volume 4307 of *Lecture Notes in Computer Science*, pages 1–20. Springer Berlin / Heidelberg, 2006.

[LM06] Xuejia Lai and James Massey. A proposal for a new block encryption standard. In Ivan Damgrd, editor, *Advances in Cryptology – EUROCRYPT '90*, volume 473 of *Lecture Notes in Computer Science*, pages 389–404. Springer Berlin / Heidelberg, 2006.

[LSBV08] Yong Ki Lee, Kazuo Sakiyama, Lejla Batina, and Ingrid Verbauwhede. Elliptic-curve-based security processor for RFID. *IEEE Transactions on Computers*, 57:1514–1527, 2008.

[Man11] Stéphane Manuel. Classification and generation of disturbance vectors for collision attacks against SHA-1. *Designs, Codes and Cryptography*, 59:247–263, 2011.

[Mao04] Wenbo Mao. *Modern Cryptography: Theory and Practice*. Prentice Hall PTR, 2004.

[McC90] Kevin McCurley. The discrete logarithm problem. *In Cryptology and computation number theory*, 42:49–74, 1990.

[Mer03] Merloni unveils RFID appliances. *RFID Journal,*, April 2003. http://www.rfidjournal.com/article/articleview/369/1/1/.

[Mit04] Chris Mitchell. *Security for mobility*. IET, 2004.

[MLDL09] Changshe Ma, Yingjiu Li, Robert H. Deng, and Tieyan Li. RFID privacy: relation between two notions, minimal condition, and efficient construction. In *Proceedings of the 16th ACM conference on Computer and communications security*, CCS '09, pages 54–65, New York, NY, USA, 2009. ACM.

[MR06] M. McLoone and M. Robshaw. Public key cryptography and RFID tags. In Masayuki Abe, editor, *Topics in Cryptology – CT-RSA 2007*, volume 4377 of *Lecture Notes in Computer Science*, pages 372–384. Springer Berlin / Heidelberg, 2006.

[MRW04] Josh Mandel, Austin Roach, and Keith Winstein. MIT proximity card vulnerabilities. Mar. 2004. http://web.mit.edu/keithw/Public/MIT-Card-Vulnerabilities-March31.pdf.

[MSW06] David Molnar, Andrea Soppera, and David Wagner. A scalable, delegatable pseudonym protocol enabling ownership transfer of RFID tags. In Bart Preneel and Stafford Tavares, editors, *Selected Areas in Cryptography*, volume 3897 of *Lecture Notes in Computer Science*, pages 276–290. Springer Berlin / Heidelberg, 2006.

[MvOV97] Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone. *Handbook of Applied Cryptography*. CRC Press, 1997.

[MW04] David Molnar and David Wagner. Privacy and security in library RFID: issues, practices, and architectures. In *Proceedings of the 11th ACM conference on Computer and communications security*, CCS '04, pages 210–219, New York, NY, USA, 2004. ACM.

[O'C05] Mary Catherine O'Connor. MasterCard PayPass beefs up NFL lineup. *RFID Journal,*, Feb. 2005. http://www.rfidjournal.com/article/view/1420.

[Oka93] Tatsuaki Okamoto. Provably secure and practical identification schemes and corresponding signature schemes. In Ernest Brickell, editor, *Advances in Cryptology—CRYPTO'92*, volume 740 of *Lecture Notes in Computer Science*, pages 31–53. Springer Berlin / Heidelberg, 1993.

[OP01] Tatsuaki Okamoto and David Pointcheval. The gap-problems: A new class of problems for the security of cryptographic schemes. In Kwangjo Kim, editor, *Public Key Cryptography*, volume 1992 of *Lecture Notes in Computer Science*, pages 104–118. Springer Berlin / Heidelberg, 2001.

[OP08] Khaled Ouafi and Raphael Phan. Traceable privacy of recent provably-secure RFID protocols. In Steven Bellovin, Rosario Gennaro, Angelos Keromytis, and Moti Yung, editors, *Applied Cryptography and Network Security*, volume 5037 of *Lecture Notes in Computer Science*, pages 479–489. Springer Berlin / Heidelberg, 2008.

[OSK03] Miyako Ohkubo, Koutarou Suzuki, and Shingo Kinoshita. Cryptographic approach to "privacy-friendly" tags. In *RFID Privacy Workshop*, November 2003.

[PB61] W.W. Peterson and D.T. Brown. Cyclic codes for error detection. *Proceedings of the IRE*, 49(1):228 –235, jan. 1961.

[PG07] William Pentland and David E. Gumpert. USDA Bets the Farm on Animal ID Program. *The Nation,*, Dec. 2007. http://www.thenation.com/article/usda-bets-farm-animal-id-program.

[PLHCETR09] Pedro Peris-Lopez, Julio Cesar Hernandez-Castro, Juan M. Estevez-Tapiador, and Arturo Ribagorda. Cryptanalysis of a novel authentication protocol conforming to EPC-C1G2 standard. *Computer Standards & Interfaces*, 31(2):372 – 380, 2009.

[PLHCTvdL11] P Peris-Lopez, Julio C. Hernandez-Castro, J. M. E. Tapiado, and Jan C. A. van der Lubbe. Cryptanalysis of an EPC Class-1 Generation-2 Standard Compliant Authentication Protocol. *Engineering Applications of Artificial Intelligence*, 24(6):1061–1069, 2011.

[Rab79] M. O. Rabin. Digitalized signatures and public-key functions as intractable as factorization. Technical report, Cambridge, MA, USA, 1979.

[RCT06] Melanie R. Rieback, Bruno Crispo, and Andrew S. Tanenbaum. The evolution of RFID security. *IEEE Pervasive Computing*, 5:62–69, 2006.

[RFI] A summary of RFID standards. *RFID Journal,*. http://www.rfidjournal.com/article/view/1335/1.

[Riv92a] R. Rivest. The MD5 message-digest algorithm. *RFC Editor*, 1992.

[Riv92b] Ron L. Rivest. The RC4 encryption algorithm. *RSA Data Security, Inc*, Mar. 1992.

[Riv95] Ronald Rivest. The RC5 encryption algorithm. In Bart Preneel, editor, *Fast Software Encryption*, volume 1008 of *Lecture Notes in Computer Science*, pages 86–96. Springer Berlin / Heidelberg, 1995.

[Rob05] Mark Roberti. Target, Wal-Mart share epc data. *RFID Journal,*, Oct. 2005. http://www.rfidjournal.com/article/view/1928/1/1.

[Rot08] P. Rotter. A framework for assessing RFID system security and privacy risks. *Pervasive Computing, IEEE*, 7(2):70 –77, April-June 2008.

[RSA78] R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Commun. ACM*, 21:120–126, February 1978.

[SA09] Yu Sasaki and Kazumaro Aoki. Finding preimages in full MD5 faster than exhaustive search. In Antoine Joux, editor, *Advances in Cryptology – EUROCRYPT 2009*, volume 5479 of *Lecture Notes in Computer Science*, pages 134–152. Springer Berlin / Heidelberg, 2009.

[Sch90] C. Schnorr. Efficient identification and signatures for smart cards. In Gilles Brassard, editor, *Advances in Cryptology—CRYPTO'89 Proceedings*, volume 435 of *Lecture Notes in Computer Science*, pages 239–252. Springer Berlin / Heidelberg, 1990.

[Sha48] C.E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423, 623–656, 1948.

[SM08] Boyeon Song and Chris J. Mitchell. RFID authentication protocol for low-cost tags. In *Proceedings of the first ACM conference on Wireless network security*, WiSec '08, pages 140–147, New York, NY, USA, 2008. ACM.

[SS05] J. Saito and K. Sakurai. Grouping proof for RFID tags. In *Advanced Information Networking and Applications, 2005. AINA 2005. 19th International Conference on*, volume 2, pages 621 – 624 vol.2, march 2005.

[Sti06] Douglas Robert Stinson. *Cryptography: theory and practice*. Chapman & Hall/CRC, 2006.

[Sys] RFID system components and costs. *RFID Journal,*. http://www.rfidjournal.com/article/view/1336/1.

[TB06] Pim Tuyls and Lejla Batina. RFID-tags for anti-counterfeiting. In David Pointcheval, editor, *Topics in Cryptology – CT-RSA 2006*, volume 3860 of *Lecture Notes in Computer Science*, pages 115–131. Springer Berlin / Heidelberg, 2006.

[THD+06] F Thornton, B Haines, A.M. Das, H Bhargava, A Campbell, and J Kleinschmidt. *RFID Security*. Syngress, Massachusetts, USA, 2006.

[Tsu06] Gene Tsudik. YA-TRAP: Yet another trivial RFID authentication protocol. In *Proceedings of the 4th annual IEEE international conference on Pervasive Computing and Communications Workshops*, PERCOMW '06, pages 640–, Washington, DC, USA, 2006. IEEE Computer Society.

[Vau07] Serge Vaudenay. On privacy models for RFID. In *Proceedings of the Advances in Crypotology 13th international conference on Theory and application of cryptology and information security*, ASIACRYPT'07, pages 68–87, Berlin, Heidelberg, 2007. Springer-Verlag.

[vDR08] T. van Deursen and S. Radomirovic. Attacks on rfid protocols. Cryptology ePrint Archive, Report 2008/310, 2008.

[VDR10] Ton Van Deursen and Saša Radomirović. EC-RAC: enriching a capacious RFID attack collection. In *Proceedings of the 6th international conference on Radio frequency identification: security and privacy issues*, RFIDSec'10, pages 75–90, Berlin, Heidelberg, 2010. Springer-Verlag.

[WSRE04] Stephen Weis, Sanjay Sarma, Ronald Rivest, and Daniel Engels. Security and privacy aspects of low-cost radio frequency identification systems. In Dieter Hutter, Gnter Mller, Werner Stephan, and Markus Ullmann, editors, *Security in Pervasive Computing*, volume 2802 of *Lecture Notes in Computer Science*, pages 50–59. Springer Berlin / Heidelberg, 2004.

[WVL06] Angela M. Wicks, John K. Visich, and Suhong Li. Radio frequency identification applications in hospital environments. *Hospital Topics*, 84(3):3–9, 2006.

[WY05] Xiaoyun Wang and Hongbo Yu. How to break MD5 and other hash functions. In Ronald Cramer, editor, *Advances in Cryptology – EUROCRYPT 2005*, volume 3494 of *Lecture Notes in Computer Science*, pages 561–561. Springer Berlin / Heidelberg, 2005.

[WYY05] Xiaoyun Wang, Yiqun Yin, and Hongbo Yu. Finding collisions in the full SHA-1. In Victor Shoup, editor, *Advances in Cryptology – CRYPTO 2005*, volume 3621 of *Lecture Notes in Computer Science*, pages 17–36. Springer Berlin / Heidelberg, 2005.

[XF09] Tao Xie and Dengguo Feng. How to find weak input differences for MD5 collision attacks. Cryptology ePrint Archive, Report 2009/223, 2009. http://eprint.iacr.org/.

[YL10] KH Yeh and NW Lo. Improvement of Two Lightweight RFID Authentication Protocols. *Information Assurance and Security Letters*, 1:6–11, 2010.

[YWKW10] Tzu-Chang Yeh, Yan-Jun Wang, Tsai-Chi Kuo, and Sheng-Shih Wang. Securing RFID systems conforming to EPC Class 1 Generation 2 standard. *Expert Systems with Applications*, 37(12):7678 – 7683, 2010.