RADIO FREQUENCY IDENTIFICATION SYSTEM SECURITY

Cryptology and Information Security Series

The Cryptology & Information Security Series (CISS) presents the latest research results in the theory and practice, analysis and design, implementation, application and experience of cryptology and information security techniques. It covers all aspects of cryptology and information security for an audience of information security researchers with specialized technical backgrounds.

Coordinating Series Editors: Raphael C.-W. Phan and Jianying Zhou

Volume 4

Recently published in this series

Vol. 3. C. Czosseck and K. Geers (Eds.), The Virtual Battlefield: Perspectives on Cyber Warfare
Vol. 2. M. Joye and G. Neven (Eds.), Identity-Based Cryptography
Vol. 1. J. Lopez and J. Zhou (Eds.), Wireless Sensor Network Security

ISSN 1871-6431

Radio Frequency Identification System Security

RFIDsec'10 Asia Workshop Proceedings

Edited by

Yingjiu Li

School of Information Systems, Singapore Management University, Singapore

and

Jianying Zhou

Network Security Group, Institute for Infocomm Research, Singapore

IOS Press

Amsterdam • Berlin • Tokyo • Washington, DC

ISBN 978-1-60750-484-9 (print)
ISBN 978-1-60750-485-6 (online)
Library of Congress Control Number: 2009944158

Publisher
IOS Press BV
Nieuwe Hemweg 6B
1013 BG Amsterdam
The Netherlands
fax: +31 20 687 0019
e-mail: order@iospress.nl

Distributor in the USA and Canada
IOS Press, Inc.
4502 Rachael Manor Drive
Fairfax, VA 22032
USA
fax: +1 703 323 3668
e-mail: iosbooks@iospress.com

PRINTED IN THE NETHERLANDS

Radio Frequency Identification System Security
Y. Li and J. Zhou (Eds.)
IOS Press, 2010

Preface

This volume contains the papers presented at the 2010 Workshop on RFID Security (RFIDsec'10 Asia) held in Singapore on February 22–23, 2010. The workshop was hosted by the School of Information Systems at Singapore Management University (SMU), and co-hosted by the Institute for Infocomm Research and Singapore Institute of Manufacturing Technology. The Honorary Chair was Steven Miller, and the General Chairs were Robert H. Deng and Lee Eng Wah.

RFIDSec'10 Asia is aligned with the earliest RFID security workshop (RFIDsec) that has been devoted to address the security and privacy issues in Radio Frequency Identification (RFID). Starting in 2005, RFIDsec has been organized as a series of workshops held in Graz (2005/06), Malaga (2007), Budapest (2008), and Leuven (2009). RFIDSec'10 Asia is the second edition of the series of workshops held in Asia followed by RFIDsec'09 Asia in Taipei (2009).

RFIDsec'10 Asia provided an international forum for sharing original research results and application experiences among researchers in the field of RFID system security. This year we had an excellent program that consists of 12 high-quality papers, including four invited papers, which were selected after a rigorous reviewing process by the Program Committee members and external reviewers. Covered are many interesting topics, including unconditionally secure RFID systems, dynamic RFID tag authentication, RFID ownership transfer, fingerprinting RFID tags, and secure RFID-supported supply chains. This is the first year for RFIDsec Asia to have the formal proceedings published by IOS Press in the Cryptology and Information Security (CIS) Series. Selected papers in the RFIDsec'10 Asia proceedings will be invited for submission to a special issue of the Journal of Computer Security.

RFIDsec'10 Asia was made possible only through the contributions from many individuals and organizations. We thank all the authors who submitted papers. We gratefully acknowledge the Program Committee members and external reviewers for the time and effort they put into reviewing the submissions. We further thank the workshop organization committee, especially, Ying Qiu for managing the web site for paper submission, review, and notification, Tieyan Li and Kevin Chiew for designing and managing the workshop web site, Chew Hong Ong for workshop registration, and Wei He for local arrangement. Last but not least, we are grateful to the SMU School of Information Systems for sponsoring the workshop.

Yingjiu Li and Jianying Zhou
February 2010

This page intentionally left blank

RFIDsec'10 Asia
The 2010 Workshop on RFID Security

February 22–23, 2010
Singapore

Hosted by
School of Information Systems, Singapore Management University

Co-hosted by
Institute for Infocomm Research
Singapore Institute of Manufacturing Technology

Sponsored by
School of Information Systems, Singapore Management University

Honorary Chair

Steven Miller (Singapore Management University, Singapore)

General Chairs

Robert H. Deng (Singapore Management University, Singapore)
Lee Eng Wah (National RFID Center, Singapore)

Program Chairs

Yingjiu Li (Singapore Management University, Singapore)
Jianying Zhou (Institute for Infocomm Research, Singapore)

Program Committee

Manfred Aigner (TU-Graz, Austria)
Mike Burmester (FSU, USA)
Colin Boyd (QUT, Australia)
Jingde Cheng (Saitama University, Japan)
Hung-Yu Chien (NCNU, Taiwan)
Shuo-Yan Chou (NTUST, Taiwan)
Chao-Hsien Chu (PSU, USA)
Tassos D. Dimitriou (AIT, Greece)
Juan Estevez-Tapiador (York University, UK)
Jinsong Han (HKUST, China)

Julio C. Hernandez-Castro (University of Portsmouth, UK)
Ari Juels (RSA Laboratories, USA)
Florian Kerschbaum (SAP Research, Germany)
Kwangjo Kim (KAIST, Korea)
Marek Klonowski (Wroclaw University of Technology, Poland)
Miroslaw Kutylowski (Wroclaw University of Technology, Poland)
Qun Li (College of William and Mary, USA)
Tieyan Li (I2R, Singapore)
Nai-Wei Lo (NTUST, Taiwan)
Li Lu (HKUST, China)
Masahiro Mambo (University of Tsukuba, Japan)
Yi Mu (UOW, Australia)
Pedro Peris-Lopez (TU-Delft, Netherlands)
Raphael Phan (Loughborough University, UK)
Reihaneh Safavi-Naini (University of Calgary, Canada)
Kazuo Sakiyama (University of Electro-Communications, Japan)
Kouichi Sakurai (Kyushu University, Japan)
Willy Susilo (UOW, Australia)
Guilin Wang (University of Birmingham, UK)
Yanjiang Yang (I2R, Singapore)
Chan Yeob Yeun (KUSTAR, UAE)

Organizing Committee

Shaoying Cai (SMU, Singapore)
Kevin Chiew (SMU, Singapore)
Ge Fu (SMU, Singapore)
Wei He (SIMTech, Singapore)
Ying Qiu (I2R, Singapore)
Qiang Yan (SMU, Singapore)

Contents

This page intentionally left blank

Radio Frequency Identification System Security
Y. Li and J. Zhou (Eds.)
IOS Press, 2010

doi:10.3233/978-1-60750-485-6-1

Securing Low-cost RFID Systems: an Unconditionally Secure Approach

Basel Alomair [a], Loukas Lazos [b], and Radha Poovendran [a]
[a] *Network Security Lab (NSL), University of Washington*
[b] *Electrical and Computer Engineering Dept., University of Arizona*
e-mail: {alomair,rp3}@u.washington.edu and llazos@ece.arizona.edu

Abstract. We explore a new direction towards solving the identity authentication problem in RFID systems. We break the RFID authentication process into two main problems: message authentication and random number generation. For parties equipped with a good source of randomness and a secure cryptographic primitive to authenticate messages, the literature of cryptography is rich with well-studied solutions for secure identity authentication. However, the two operations, random number generation and message authentication, can be expensive for low-cost RFID tags. In this paper, we lay down the foundations of a new direction towards solving these problems in RFID systems. We propose an unconditionally secure direction for authenticating RFID systems. We use the fact that RFID readers are computationally powerful devices to design a protocol that allows RFID readers to deliver random numbers to RFID tags in an unconditionally secure manner. Then, by taking advantage of the information-theoretic security of the transmitted messages, we develop a novel unconditionally secure message authentication code that is computed with a single multiplication operation. The goal of this work is to bring more research to the design of such unconditionally secure protocols, as opposed to the computationally secure protocols that have been proposed extensively, for the purpose of suiting the stringent computational capabilities of low-cost devices.

Keywords. RFID, unconditional security, authentication, secrecy

Introduction

While mutual authentication is a well-studied problem in the cryptographic literature, it becomes more challenging with the use of low-cost devices. Low-cost RFID tags, in particular, have limited computational capabilities that render them unable to perform sophisticated cryptographic operations. Hoping Moore's law will eventually render RFID tags computationally powerful, it might be tempting to consider the computational limitations of low-cost tags a temporary problem. The cost of tags, however, will remain a determining factor in the deployment of RFID systems in real life applications. When RFID technology is to replace barcodes to identify individual items, RFID tags will substantially contribute to the cost of these products. Even when the price of tags that can implement provably secure cryptography can be driven to 10 cents or less, it would still be impractical to attach them to low-cost items, e.g., 50-cent or cheaper products. When retailers are to choose between tags that can perform sophisticated cryptographic operations and cheaper tags that cannot, it seems inevitable that the cheaper tags will prevail.

The problem of mutual authentication in RFID systems has been studied under different constraints. Juels and Pappu proposed the use of a public key cryptosystem to solve the problem of consumer privacy in RFID banknotes [22]. Golle *et al.* [19] proposed the universal re-encryption, where re-encryption of the ciphertext does not require the knowledge of the corresponding public key. As public key proposals might be suitable for some applications (e.g., banknotes [22], ePassports [4], credit cards [21], etc.), they are impractical for low-cost RFID tags. Consequently, proposals based on the hardness of breaking symmetric key primitives have been the most popular solution for RFID systems. Such solutions include the use of symmetric key encryption [13,14,12], pseudo-random functions (PRF) [26,24], cryptographic hash functions [34,5,7,10,27], or other NP-hard problems such as the linear parity with noise problem [23,16,9,8,28]. Although computationally secure symmetric key solutions are usually less expensive than asymmetric ones, they still require expensive operations and/or carefully designed iterations of complicated operations.

In order to come up with cheaper solutions, lightweight protocols that are not based on computational assumptions and require tags to perform only simple bitwise operations have been proposed, e.g., in [33,29,30]. Such proposals, however, were not based on rigorous security analysis and have been shown to have severe security flaws, see e.g., [25,17,1] for analysis of specific protocols. In fact, it has been shown in [2] that simple bitwise operations cannot lead to secure authentication protocols.

As can be observed from the above examples, previous secure RFID protocols are all *computationally secure*. Hoping to meet the limited computational capabilities of low-cost tags, we start the search for *unconditionally secure* protocols for RFIDs. Unconditional security relies on the freshness of the keys rather than the hardness of solving mathematical problems. Thus, an appropriately designed unconditionally secure protocol will normally require less computational effort.

Contributions. We propose the first UnConditionally Secure mutual authentication protocol for RFID systems (UCS-RFID). To minimize the computational effort on tags, we develop an unconditionally secure method for delivering random number from RFID readers to tags. Thus, allowing tags to benefit from the functionalities of random numbers without the hardware to generate them. Then, we take advantage of the secrecy of exchanged messages to develop a novel unconditionally secure technique for message authentication using only a single multiplication operation. Since modular multiplication is the most expensive operation tags are to perform, we show, for completeness, a simple modular multiplication algorithm that requires minimum circuitry.

The rest of the paper is organized as follows. In Section 1, we state our model assumptions. In Section 2, we describe our UCS-RFID protocol. In Section 3, we analyze the security of our UCS-RFID, and conclude the paper in Section 4. The modular multiplication algorithm is presented in the Appendix.

1. System Model

1.1. Computational Capabilities

We assume low-cost RFID tags identified via unique identifiers. Tags are assumed to be capable of performing bitwise (XOR) operations, in addition to modular multiplication and addition. RFID tags are not assumed to have the capability of performing traditional

computationally-secure cryptographic primitives such as MACs, hash functions, random numbers generations, etc.

RFID readers are powerful devices capable of performing sophisticated cryptographic primitives. Readers are also assumed to be connected to the database via a secure link (whether by establishing a secure wireless channel or using wired connection) in order to retrieve information about tags. Addressing the security between RFID readers and the database is beyond the scope of this paper.

1.2. Adversarial Model

We assume an adversary with a complete control over the communication channel. The adversary can observe messages exchanged between readers and tags, initiate communication with a reader or a tag, modify message exchanged between the authorized reader and tags in the system, block messages and replayed them later, and generate messages of her own. We do not consider an adversary whose only goal is to jam the communication channel.

The adversary is modeled as a polynomial-time algorithm. Similar to the adversarial model proposed by Avoine in [3], given a tag T and a reader R, we assume that the adversary has access to the following oracles:

- *Send*(R, m_1, x_2, m_3): The adversary executes the protocol, acting as a tag. The adversary sends m_1 to identify itself to R; receives the reader's response, x_2; and authenticate itself with $m_3(x_2)$. This oracle models the adversary's ability to impersonate a tag in the system.
- *Query*(T, x_1, m_2, x_3): The adversary acts as the reader in an instance of the protocol. The adversary interrogates T; receives the tag's response, x_1; sends the message $m_2(x_1)$ to authenticate itself; and receives x_3. This oracle models the adversary's ability to impersonate valid readers.
- *Execute*(T, R): The tag T and the reader R execute an instance of the protocol. The adversary eavesdrops on the channel; this oracle models the adversary's ability to monitor the channel between tag and reader.

Observe that in practical RFID systems, unlike the *Send* and *Query* oracles, the adversary does not have complete control over the number of *Execute* oracles she can call on a particular tag. This is due to the fact that the *Execute* oracle simulates a complete protocol run between *authorized* reader-tag pairs. Thus, unless the adversary is physically capturing a tag, she does not have a complete control of when the tag is in the vicinity of an authorized reader.

1.3. Security Model

Inspired by the work of Bellare and Rogaway in [6], we define honest protocol runs as follows: A mutual authentication protocol run is said to be honest if the parties involved in the protocol run use their shared key to exchange messages, and the messages exchanged in the protocol run have been relayed faithfully (without modification).

Another term that will be used in the reminder of the paper is the definition of negligible functions. A function $\gamma : \mathbb{N} \to \mathbb{R}$ is said to be negligible if for any nonzero polynomial p, there exists N_0 such that for all $N > N_0$, $|\gamma(N)| < 1/|p(N)|$. That is, the

function is said to be negligible if it converges to zero faster than the reciprocal of any polynomial function [18].

We now provide a formal definition of secure mutual authentication for RFID systems.

Definition 1 *A mutual authentication protocol for RFID systems is said to be secure if it satisfies the following conditions:*

1. *No information about the secret keys of the RFID tag is revealed by observing messages exchanged in protocol runs.*
2. ***Honest protocol*** $\Rightarrow$ ***Authentication:*** *if the protocol run is honest, the tag-reader pair must authenticate each other with probability one.*
3. ***Authentication*** $\Rightarrow$ ***Honest protocol:*** *the probability of authentication when the protocol is not honest is negligible in the security parameter.*

To model the adversary's attempt to authenticate herself to a reader (tag), we propose the following game between the challenger $\mathcal{C}$ (the RFID system) and an adversary $\mathcal{A}$.

1. $\mathcal{A}$ signals $\mathcal{C}$ to begin the game.
2. $\mathcal{C}$ chooses a tag, T, at random, and a reader, R, and gives them to $\mathcal{A}$.
3. $\mathcal{A}$ calls the oracles *Query, Send*, and *Execute* using T and R (this is the data collecting phase).
4. $\mathcal{A}$ decides to stop and signals $\mathcal{C}$ to move on to the next phase.
5. $\mathcal{A}$ *Send* (*Query*) $\mathcal{C}$ as if it is a tag (reader) in the system (this is the actual attempt to break the security of the system).
6. If $\mathcal{A}$ is authenticated as a valid tag (reader), $\mathcal{A}$ wins the game.

Definition 1 implies that the protocol achieves secure mutual authentication only if the adversary's probability of winning the game is negligible in the security parameter.

1.4. Preliminaries

The following notations will be adopted throughout the paper. For a finite integer p, $\mathbb{Z}_p$ will denote the finite integer ring with the usual addition and multiplication modulo p. $\mathbb{Z}_p^*$ will denote the multiplicative group modulo p; that is, the subset of $\mathbb{Z}_p$ with elements relatively prime to p. For the special case at which p is a prime integer, $\mathbb{Z}_p^*$ will contain all non-zero elements of $\mathbb{Z}_p$; that is, $\mathbb{Z}_p^* = \mathbb{Z}_p \backslash \{0\}$. $\mathbb{Z}_p^*$ and $\mathbb{Z}_p \backslash \{0\}$ will be used interchangeably to emphasize the multiplicative property or the exclusion of the zero element, respectively. Throughout the rest of the paper, random variables will be represented by bold font symbols, whereas the corresponding non-bold font symbols represent specific values that can be taken by these random variables.

The following are two important properties of the integer ring $\mathbb{Z}_p$ that will be used in the security analysis of UCS-RFID.

Lemma 1 *For any two integers α and β in $\mathbb{Z}_p$, if p is a prime integer and p divides $\alpha\beta$, then one of the integers α and β must be the zero element in $\mathbb{Z}_p$. Formally, if p is a prime integer, $\{\alpha\beta \equiv 0 \mod p\}$ implies that $\{\alpha \equiv 0$ OR $\beta \equiv 0 \mod p\}$.*

Lemma 2 *Let p be a prime integer. Then, given an integer $k \in \mathbb{Z}_p^*$, for an r uniformly distributed over $\mathbb{Z}_p$, the value $\delta \equiv rk \mod p$ is uniformly distributed over $\mathbb{Z}_p$.*

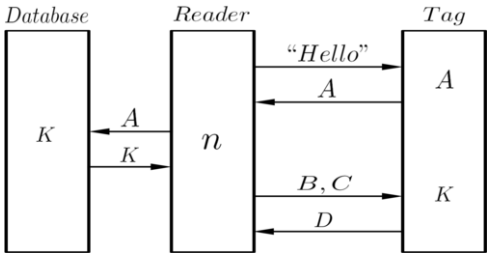


Figure 1. A schematic of one instance of the protocol.

Lemma 1 states that, for a prime integer p, the ring $\mathbb{Z}_p$ is an integral domain. Lemma 2 is a direct consequence of the fact that, for a prime integer p, the ring $\mathbb{Z}_p$ is a field. One more definition that is vital for this paper is the definition of Shannon's perfect secrecy [31].

Definition 2 (Perfect Secrecy [32]) *For a plaintext m and its corresponding ciphertext φ, the cipher is said to achieve perfect secrecy if $\Pr(\boldsymbol{m} = m|\boldsymbol{\varphi} = \varphi) = \Pr(\boldsymbol{m} = m)$ for all plaintext m and all ciphertext φ. That is, the a posteriori probability that the plaintext is m, given that the ciphertext φ is observed, is identical to the a priori probability that the plaintext is m.*

2. The proposed UCS-RFID Protocol

Based on pre-defined security requirements, a security parameter, N, is specified and a $2N$-bit prime integer, p, is chosen. Initially, each tag is loaded with an N-bit long identifier, $A^{(0)}$, and a secret key composed of five subkeys, i.e., $K^{(0)} = (k_a^{(0)}, k_b^{(0)}, k_c^{(0)}, k_d^{(0)}, k_u^{(0)})$. The length of k_a and k_d is N bits, while k_b, k_c, and k_u are $2N$-bit long. The subkeys, $k_a^{(0)}$ and $k_d^{(0)}$, and the identifier, $A^{(0)}$, are drawn independently and uniformly from $\mathbb{Z}_{2^N}$; $k_b^{(0)}$ is drawn uniformly from $\mathbb{Z}_p$; while $k_c^{(0)}$ and $k_u^{(0)}$ are drawn independently and uniformly from $\mathbb{Z}_p^*$. The subkeys k_a, k_b, k_c, and k_d will be used to generate messages exchanged in protocol runs, while the sole purpose of k_u is for updating the secret keys to maintain certain properties (details are discussed later).

The security of the protocol relies on the reader's ability to convey a random nonce to the tag in an *authenticated* and *secret* manner. When an RFID reader interrogates a tag within its communication range, the tag responds with its identifier, A. Once the tag has been identified, the reader generates a $2N$-bit long random nonce, n, and delivers it to the tag. If the reader is authenticated successfully, the received n will be used by the tag to authenticate itself to the valid reader.

For the rest of the paper, quantities involved in the generation of exchanged messages in different protocol runs will be differentiated by superscripts. When differentiation between protocol runs is unnecessary, superscripts will be dropped for ease of notation.The proposed UCS-RFID enables the mutual authentication between an RFID reader and a tag by executing four phases: a tag identification phase, a reader authentication phase, a tag authentication phase, and a key updating phase. Figure 1 depicts a single protocol run of the proposed UCS-RFID.

2.1. Tag Identification Phase

In order to carry out the authentication process, the reader must identify the tag it is communicating with to access its key information.
Step 1. The reader announces its presence by broadcasting a *"Hello"* message.
Step 2. The tag responds to the *"Hello"* message by sending its current identifier, A.
Step 3. The reader looks up the database for the key $K = (k_a, k_b, k_c, k_d, k_u)$ corresponding to the tag's current identifier, A.[1] If A is not recognized as a valid identifier, the tag is rejected.

2.2. Reader Authentication Phase

This is one of the most important phases in the proposed protocol. In this phase, the RFID reader authenticates itself to the tag by proving its knowledge of the tag's subkeys k_b and k_c. More importantly, the reader delivers a nonce, n, to the tag in an authenticated and perfectly secret manner.
Step 4. The reader generates a $2N$-bit random nonce, n, drawn uniformly from the *multiplicative group* $\mathbb{Z}_p^*$. We emphasize that n must be an unpredictable nonce; predictable nonces such as time stamps do not induce the required randomness.
Step 5. With k_b, k_c, and n, the reader broadcasts two messages, B and C, generated according to the following formulas:

$$B \equiv n + k_b \mod p, \tag{1}$$

$$C \equiv n \times k_c \mod p. \tag{2}$$

Step 6. Upon receiving B and C, the tag extracts n from message B and verifies its integrity using message C. The reader is authenticated if and only if the following integrity check is satisfied,

$$(B - k_b) \times k_c \equiv C \mod p. \tag{3}$$

2.3. Tag Authentication Phase

In the tag authentication phase, the tag is authenticated by its ability to extract the correct nonce, n, and its knowledge of the secret key k_d.
Step 7. If the reader failed the authentication process, the tag aborts the protocol. Otherwise, the tag broadcasts message D, given by

$$D = n_\ell \oplus k_d, \tag{4}$$

where n_ℓ denotes the N most significant bits of n.
Step 8. Upon receiving D, the reader authenticates the tag by verifying that the received D is equal to $n_\ell \oplus k_d$. Otherwise, the tag is rejected.

[1] Database management is beyond of the scope of this paper.

2.4. Key Update Phase

After a mutual authentication between the RFID reader and the tag is achieved, the parameters are updated at the database and the tag for the next mutual authentication run.

Step 9. The reader and the tag update the key, K, and the tag identifier, A. Let $A^{(m)}$, $k_i^{(m)}$, and $n^{(m)}$ denote the identifier A, k_i, and n used to execute the m^{th} protocol run; let n_r denotes the N least significant bits of n. Then, the parameters are updated as follows,

$$k_a^{(m+1)} = n_r^{(m)} \oplus k_a^{(m)}, \tag{5}$$

$$k_b^{(m+1)} \equiv k_u^{(m)} + (n^{(m)} \oplus k_b^{(m)}) \mod p, \tag{6}$$

$$k_c^{(m+1)} \equiv k_u^{(m)} \times (n^{(m)} \oplus k_c^{(m)}) \mod p, \tag{7}$$

$$k_d^{(m+1)} = n_r^{(m)} \oplus k_d^{(m)}, \tag{8}$$

$$k_u^{(m+1)} \equiv k_u^{(m)} \times n^{(m)} \mod p, \tag{9}$$

$$A^{(m+1)} \equiv n_\ell^{(m)} + k_a^{(m+1)} \mod 2^N. \tag{10}$$

It is vital for the security of the protocol that the updated $k_b^{(m+1)}$ and $k_c^{(m+1)}$ remain uniformly distributed over $\mathbb{Z}_p$ and $\mathbb{Z}_p \backslash \{0\}$, respectively. Here is where the updating key, k_u, comes to play. In addition to inducing a desired independence between message $B^{(m)}$ and the updated $k_b^{(m+1)}$, and between message $C^{(m)}$ and the updated $k_c^{(m+1)}$, observe that, by Property 2, $k_u^{(m)}$ will always be uniformly distributed over $\mathbb{Z}_p^*$ (since the initial $k_u^{(0)}$ is drawn uniformly from $\mathbb{Z}_p^*$ and every generated nonce is a random element of $\mathbb{Z}_p^*$). Therefore, $k_b^{(m+1)}$ is uniformly distributed over $\mathbb{Z}_p$. However, there is a possibility that $k_c^{(m+1)}$ will be equal to zero; which will occur, *with negligible probability*, when $n^{(m)} \oplus k_c^{(m)}$ is congruent to zero modulo p. In this case, $n^{(m)} \oplus k_c^{(m)}$ in equation (7) is replaced with $n^{(m)} \times k_c^{(m)}$. Now, $n^{(m)} \times k_c^{(m)}$ is guaranteed not to be congruent to zero (by Property 1), which guarantees that $k_c^{(m+1)}$ is not zero. The reason for not starting with $n^{(m)} \times k_c^{(m)}$ in the update equation of $k_c^{(m+1)}$ is that this is equal to $C^{(m)}$, which will lead to revealing information about the nonce with the observation of multiple consecutive protocol runs. With the update procedure described above, $n^{(m)} \times k_c^{(m)}$ will be used for updating $k_c^{(m+1)}$ with negligible probability, and even when it is used, the adversary can never know that it is being used. Therefore, without loss of generality, we will assume for the rest of the paper that equation (7) always results in a $k_c^{(m+1)}$ that is uniformly distributed over $\mathbb{Z}_p \backslash \{0\}$.

3. Security Analysis

Before we show the security of our UCS-RFID, we will first prove our claims that, under our adversarial model, the integrity of the delivered nonce, n, can be verified using a single modular multiplication, and show that the random nonce is delivered to tags in an unconditionally secure manner.

3.1. Integrity of the Delivered Nonce

In this section, we will show how the integrity of the nonce, n, is preserved without resorting to computationally secure cryptographic primitives. The integrity of the delivered nonce in our UCS-RFID is accomplished in a novel way, by taking advantage of the properties of the integer field $\mathbb{Z}_p$, with only a single multiplication operation.

There are two cases to consider: modifying message B alone and modifying both B and C in order to make the tag authenticate a false nonce. Modifying message C alone, since its main purpose is to authenticate the received nonce, does not lead to the extraction of a modified nonce.

Lemma 3 *Given that k_c is uniformly distributed over $\mathbb{Z}_p\backslash\{0\}$, the probability of accepting a modified nonce by a valid tag is at most $1/(p-1)$.*

Proof: Assume that message B has been modified to B'. This modification will lead to the extraction of a nonce, n', different than the authentic n generated by the reader; that is, $n' \equiv B' - k_b \mod p$. Message C, however, is used to verify the integrity of the extracted n'. Let $n' \equiv n + \epsilon \mod p$; for some $\epsilon \in \mathbb{Z}_p\backslash\{0\}$. To be accepted by the tag, n' must satisfy the integrity check of equation (3). That is,

$$n' \times k_c \equiv (n+\epsilon) \times k_c \equiv (n \times k_c) + (\epsilon \times k_c) \stackrel{?}{\equiv} C \equiv n \times k_c \mod p. \tag{11}$$

Clearly, the congruence in equation (11) will be satisfied only if $\epsilon \times k_c \equiv 0 \mod p$. However, since k_c is a nonzero element by design, and $\epsilon \not\equiv 0$ (since $\epsilon \equiv 0$ implies that $n' \equiv n \mod p$), Property 1 guarantees that $\epsilon \times k_c \not\equiv 0 \mod p$. Therefore, the congruence of equation (11) can never be satisfied, and any modification of message B *alone* will be detected with probability *one*.

The second case to consider here is when both messages B and C are corrupted simultaneously. Assume that message B has been modified so that the extracted nonce becomes $n' \equiv n + \epsilon \mod p$; for some $\epsilon \in \mathbb{Z}_p\backslash\{0\}$. Also, assume that message C has been modified to $C' \equiv C + \delta \mod p$, for some $\delta \in \mathbb{Z}_p\backslash\{0\}$. The integrity of the extracted n' is verified using the received C' as follows:

$$\begin{aligned} C + \delta \equiv C' \stackrel{?}{\equiv} n' \times k_c &\equiv (n+\epsilon) \times k_c \\ &\equiv (n \times k_c) + (\epsilon \times k_c) \equiv C + (\epsilon \times k_c) \mod p. \end{aligned} \tag{12}$$

Equivalently, the false n' is accepted only if $\delta \equiv \epsilon \times k_c$. Since k_c is unknown to the adversary, for any fixed δ, by Property 2, there exists a unique $\epsilon \in \mathbb{Z}_p\backslash\{0\}$ that satisfies (12). Therefore, the probability of modifying both B and C in a way undetected by the tag is at most $1/(p-1)$ (equivalently, guessing the value of k_c). ■

3.2. Secrecy of the Delivered Nonce

Before we show that the nonce is delivered to the tag in an unconditionally secure manner, we need the following lemma.

Lemma 4 *Given that the preloaded subkeys $k_a^{(0)}$, $k_b^{(0)}$, $k_c^{(0)}$, and $k_d^{(0)}$ are mutually independent, the subkeys $k_a^{(m)}$, $k_b^{(m)}$, $k_c^{(m)}$, and $k_d^{(m)}$ at the m^{th} protocol run are mutually independent, for any $m \in \mathbb{N}$.*

Proof: Let $\boldsymbol{k_a^{(m)}}, \boldsymbol{k_b^{(m)}}, \boldsymbol{k_c^{(m)}}$, and $\boldsymbol{k_d^{(m)}}$ be the random variables representing the subkeys involved in the generation of the messages exchanged between an authorized RFID pair during the m^{th} protocol run of our UCS-RFID. Then, for any $k_a^{(1)}, k_b^{(1)}, k_c^{(1)}$, and $k_d^{(1)}$,

$$\Pr\left(\boldsymbol{k_a^{(1)}} = k_a^{(1)}, \boldsymbol{k_b^{(1)}} = k_b^{(1)}, \boldsymbol{k_c^{(1)}} = k_c^{(1)}, \boldsymbol{k_d^{(1)}} = k_d^{(1)}\right)$$

$$= \sum_{n,k_u} \Pr\left(\boldsymbol{k_a^{(1)}} = k_a^{(1)}, \boldsymbol{k_b^{(1)}} = k_b^{(1)}, \boldsymbol{k_c^{(1)}} = k_c^{(1)}, \boldsymbol{k_d^{(1)}} = k_d^{(1)} | \boldsymbol{n} = n, \boldsymbol{k_u} = k_u\right)$$

$$\cdot \Pr\left(\boldsymbol{n} = n, \boldsymbol{k_u} = k_u\right) \quad (13)$$

$$= \sum_{n,k_u} \Pr\left(\boldsymbol{k_a^{(0)}} = k_a^{(1)} \oplus n_r, \boldsymbol{k_b^{(0)}} = (k_b^{(1)} - k_u^{(0)}) \oplus n, \boldsymbol{k_c^{(0)}} = (k_c^{(1)} \times {k_u^{(0)}}^{-1}) \oplus n\right.$$

$$\left., \boldsymbol{k_d^{(0)}} = k_d^{(1)} \oplus n_r\right) \cdot \Pr\left(\boldsymbol{n} = n, \boldsymbol{k_u} = k_u\right) \quad (14)$$

$$= \sum_{n,k_u} \Pr\left(\boldsymbol{k_a^{(0)}} = k_a^{(1)} \oplus n_r\right) \cdot \Pr\left(\boldsymbol{k_b^{(0)}} = (k_b^{(1)} - k_u^{(0)}) \oplus n\right)$$

$$\cdot \Pr\left(\boldsymbol{k_c^{(0)}} = (k_c^{(1)} \times {k_u^{(0)}}^{-1}) \oplus n\right) \cdot \Pr\left(\boldsymbol{k_d^{(0)}} = k_d^{(1)} \oplus n_r\right) \cdot \Pr\left(\boldsymbol{n} = n, \boldsymbol{k_u} = k_u\right) \quad (15)$$

$$= \sum_{n,k_u} \frac{1}{2^N} \cdot \frac{1}{p} \cdot \frac{1}{p-1} \cdot \frac{1}{2^N} \cdot \Pr\left(\boldsymbol{n} = n, \boldsymbol{k_u} = k_u\right) \quad (16)$$

$$= \Pr\left(\boldsymbol{k_a^{(1)}} = k_a^{(1)}\right) \cdot \Pr\left(\boldsymbol{k_b^{(1)}} = k_b^{(1)}\right) \cdot \Pr\left(\boldsymbol{k_c^{(1)}} = k_c^{(1)}\right) \cdot \Pr\left(\boldsymbol{k_d^{(1)}} = k_d^{(1)}\right). \quad (17)$$

Equations (15) and (16) hold due to the independence and the uniform distribution of the initial subkeys $(\boldsymbol{k_a^{(0)}}, \boldsymbol{k_b^{(0)}}, \boldsymbol{k_c^{(0)}}, \boldsymbol{k_d^{(0)}})$, respectively; while equation (17) holds due to the uniform distribution of the updated subkeys $(\boldsymbol{k_a^{(1)}}, \boldsymbol{k_b^{(1)}}, \boldsymbol{k_c^{(1)}}, \boldsymbol{k_d^{(1)}})$. The existence of ${k_u^{(0)}}^{-1}$, the multiplicative inverse of $k_u^{(0)}$ in $\mathbb{Z}_p$, is a direct consequence of the fact that $k_u^{(0)} \in \mathbb{Z}_p^*$. The proof of the lemma follows by induction. ■

Lemma 5 *At each instance of the protocol, the random nonce generated by the authorized reader in an instance of our UCS-RFID protocol is delivered to the tag in a perfectly secret manner.*

Proof: Fix k_b, k_c, and let $\boldsymbol{n}$ be uniformly distributed over $\mathbb{Z}_p \backslash \{0\}$. (Recall that, by Lemma 4, $\boldsymbol{k_b}$ and $\boldsymbol{k_c}$ are statistically independent in every protocol run; so, the superscript will be dropped for ease of notation.) Then the resulting $\boldsymbol{B}$ and $\boldsymbol{C}$ will be uniformly distributed over $\mathbb{Z}_p$ and $\mathbb{Z}_p \backslash \{0\}$ (by Property 2), respectively. Consequently, for any arbitrary $b \in \mathbb{Z}_p$ and $c \in \mathbb{Z}_p \backslash \{0\}$, the probability of $\boldsymbol{B}$ and $\boldsymbol{C}$ taking these specific values are $\Pr(\boldsymbol{B} = b) = 1/p$ and $\Pr(\boldsymbol{C} = c) = 1/(p-1)$.

Now, given a specific value of the random nonce $\boldsymbol{n} = n$, the probability that $\boldsymbol{B}$ takes a value b is

$$\Pr(\boldsymbol{B} = b | \boldsymbol{n} = n) = \Pr(\boldsymbol{k_b} = b - n) = 1/p. \quad (18)$$

Similarly, given a specific value of the random nonce $\boldsymbol{n} = n$, the probability that $\boldsymbol{C}$ takes a value c is

$$\Pr(\boldsymbol{C} = c|\boldsymbol{n} = n) = \Pr(\boldsymbol{k_c} = c \times n^{-1}) = 1/(p-1). \tag{19}$$

Equations (18) and (19) hold since, by design, $\boldsymbol{k_b}$ and $\boldsymbol{k_c}$ are uniformly distributed over $\mathbb{Z}_p$ and $\mathbb{Z}_p \backslash \{0\}$, respectively. The existence of n^{-1} is a direct consequence of the fact that $n \in \mathbb{Z}_p^*$.

Therefore, for any nonce n and any values of b and c, Bayes' theorem [20] can be used to show that $\Pr(\boldsymbol{n} = n|\boldsymbol{B} = b) = \Pr(\boldsymbol{n} = n) = \Pr(\boldsymbol{n} = n|\boldsymbol{C} = c)$. That is, the a priori probabilities that the random nonce is n are the same as the a posteriori probabilities that the random nonce is n given the corresponding B and C. Hence, both B and C "individually" provided perfect secrecy. However, since they are both functions of the same variable, there might be information leakage about n revealed by the *combination* of B and C. One way of measuring how much information is learned by the observation of two quantities is the notion of mutual information. Consider an arbitrary $b \in \mathbb{Z}_p$ and arbitrary $c, n \in \mathbb{Z}_p^*$. Then, for independent $\boldsymbol{k_b}$ and $\boldsymbol{k_c}$ uniformly distributed over $\mathbb{Z}_p$ and $\mathbb{Z}_p^*$, respectively, we get:

$$\begin{aligned}
\Pr(\boldsymbol{B} = b, \boldsymbol{C} = c) &= \sum_n \Pr(\boldsymbol{B} = b, \boldsymbol{C} = c|\boldsymbol{n} = n) \Pr(\boldsymbol{n} = n) && (20)\\
&= \sum_n \Pr(\boldsymbol{k_b} = b - n, \boldsymbol{k_c} = c \times n^{-1}) \Pr(\boldsymbol{n} = n) && (21)\\
&= \sum_n \Pr(\boldsymbol{k_b} = b - n) \Pr(\boldsymbol{k_c} = c \times n^{-1}) \Pr(\boldsymbol{n} = n) && (22)\\
&= \sum_n \frac{1}{p} \cdot \frac{1}{p-1} \cdot \Pr(\boldsymbol{n} = n) && (23)\\
&= \Pr(\boldsymbol{B} = b) \cdot \Pr(\boldsymbol{C} = c). && (24)
\end{aligned}$$

Equation (22) holds by the independence of $\boldsymbol{k_b}$ and $\boldsymbol{k_c}$, while equations (23) and (24) hold by the uniform distribution of $\boldsymbol{k_b}$, $\boldsymbol{k_c}$, $\boldsymbol{B}$, and $\boldsymbol{C}$. Consequently, $\boldsymbol{B}$ and $\boldsymbol{C}$ are independent and, thus, their mutual information is *zero* [11]. In other words, observing both messages B and C gives no extra information about n than what they give individually. ■

Remark 1 Lemma 5 does not hold for an adversary who has observed multiple *consecutive* protocol runs between authorized reader-tag pairs. Consider observing three consecutive B messages, say $B^{(0)}, B^{(1)}, B^{(2)}$. The fundamental problem is that only $k_b^{(0)} \in \mathbb{Z}_p$ and $k_u^{(0)} \in \mathbb{Z}_p^*$ are involved in the update equation of the subkey k_b. Therefore, out of the total $(p-1)^3$ possible sequences of $\{n^{(0)}, n^{(1)}, n^{(2)}\}$, to an adversary who has observed $\{B^{(0)}, B^{(1)}, B^{(2)}\}$, there are only $p(p-1)$ possible $\{n^{(0)}, n^{(1)}, n^{(2)}\}$ sequences that could have generated the observed B's. A violation to the definition of perfect secrecy. Obviously, one can include more variables in the update equation of k_b but that will only increase the number of consecutive protocol runs an adversary is allowed to observe to a certain number.

However, breaking perfect secrecy does not imply breaking the system. In what follows, we provide an example to further illustrate the remark; then, we give detailed probabilistic analysis to show that the system can still provide unconditional security given some practical assumptions about the RFID system.

Example 1 This example illustrates the effect of observing consecutive protocol runs between authorized reader-tag pairs. For simplicity, assume that the used prime number is $p = 7$. Assume further that the initial keys $k_b^{(0)} = 2$ and $k_u^{(0)} = 5$ are preloaded into the tag. Consider the first three protocol runs as follows.
First run: let the generated nonce be $n_0 = 1$. Then by equation (1) the adversary can observe $B_0 = 3$ broadcasted by the reader. The tag and the reader will then update the keys according to equations (6) and (9) to $k_b^{(1)} = 1$ and $k_u^{(1)} = 5$.
Second run: let the generated nonce be $n_1 = 6$. Then by equation (1) the adversary can observe $B_1 = 0$. The tag and the reader will then update the keys according to equations (6) and (9) to $k_b^{(2)} = 5$ and $k_u^{(2)} = 2$.
Third run: let the generated nonce be $n_2 = 2$. Then the adversary can observe $B_2 = 0$.
Now, with some algebra the adversary can construct the following system of equations:

$$B_0 = k_b^{(0)} + n_0, \tag{25}$$

$$B_1 = k_u^{(0)} + (n_0 \oplus k_b^{(0)}) + n_1, \tag{26}$$

$$B_2 = (k_u^{(0)} \times n_0) + \Big(n_1 \oplus \big(k_u^{(0)} + (n_0 \oplus k_b^{(0)})\big)\Big) + n_2. \tag{27}$$

Consider now the sequence $\{n_0 = 1, n_1 = 1, n_2 = 1\}$. Given the observed B's, by checking equations (25), (26), and (27), one can see that the sequence $\{n_0 = 1, n_1 = 1, n_2 = 1\}$ cannot satisfy the three equations simultaneously. Moreover, by checking all possible $6\times6\times6$ sequences, one can find that only 7×6 of them can satisfy all three equations simultaneously. The fundamental problem is that only $k_b^{(0)} \in \{0, 1, 2, 3, 4, 5, 6\}$ and $k_u^{(0)} \in \{1, 2, 3, 4, 5, 6\}$ are involved in the three equations.

Observe, however, that this does not imply anything more than that the sequence $\{n_0 = 1, n_1 = 1, n_2 = 1\}$ cannot generate the observed B's. That is, it does not imply that $n_0 \neq 1$, nor that $n_1 \neq 1$, nor that $n_2 \neq 1$. In other words, individually, any one of the n's can be equal to one (indeed, $n_0 = 1$ in the above example).

Therefore, for the adversary to obtain meaningful information, she must know the exact value of at least one of the nonces (so that possible values of other nonces can be eliminated). This can only occur if for at least one nonce n_i, only one value in $\mathbb{Z}_p^*$ is possible. That is, all the possible values n_i is allowed to take can be eliminated, except for exactly one value.

We will now provide probabilistic analysis of the number of consecutive protocol runs an adversary must observe in order to learn the value of at least one of the transmitted nonces.

By the randomness nature of the generated nonces, the total number of possible sequences is uniformly distributed over the nonces. That is, given there are $p(p-1)$ possible sequences, if the adversary has observed m consecutive protocol runs, each of the m nonces is expected to have $\sqrt[m]{p(p-1)}$ possible values. Therefore, for m consecu-

tive protocol runs, the total number of possible values distributed over the m nonces is $m\sqrt[m]{p(p-1)}$.

To give a lower bound on the number of consecutive protocol runs an adversary must observe in order to infer at least one nonce with a certain probability, we use the well-known "balls in bins without capacity" problem in probability theory. Given r balls thrown uniformly at random at m bins, the probability that at least one bin remains empty is given by [15]:

$$\Pr(\text{at least one bin remains empty}) = \frac{\binom{r-1}{m-1}}{\binom{m+r-1}{m-1}}. \tag{28}$$

Given that each nonce will take at least one value, the problem reduces to distributing $(m\sqrt[m]{p(p-1)} - m)$ values uniformly at random at m nonces and finding the probability that at least one nonce does not receive another possible value. Substituting $r = m\sqrt[m]{p(p-1)} - m$ in equation (28), we plot the results in Figure 2. Each plot shows the number of consecutive protocol runs an adversary must observe in order to infer at least one nonce with a certain probability. In the top left plot, the security parameter N is 128-bit long, with only k_b and k_u are involved in the update equation of k_b. The plot in the top right shows the result when all secret keys are involved in the update equation of k_b. The two bottom plots shows the result when the used security parameter is 256-bit long.

As can be seen in Figure 2, the number of consecutive protocol runs an adversary has to observe to learn the value of at least one nonce is much higher than the number of protocol runs needed to break perfect secrecy. Depending on how many secret keys are used in the update equations and the length of the security parameter, for an adversary to have a 50% chance of exposing a secret nonce value, the number of *consecutive* protocol runs needed to be observed can be as high as 240 complete runs.

Remark 2 Observe that, unlike general computer communications, that many consecutive protocol runs can be sufficiently high for RFID systems. Consider, for example, an RFID tag used for a pay-at-the-pump application. If the user goes to the same gas station every single time, this implies that for an adversary to extract secret tag information, she must be in a close proximity to the user for about 240 consecutive gas pumping. In the case in which the user goes to different gas stations, this implies that the adversary is following the user everywhere. Both scenarios are highly unlikely to occur in real life applications. In a different example, consider low-cost tags replacing barcodes for identifying grocery items. In such applications, that many authorized protocol runs are unlikely to occur during the entire life time of a low-cost tag. The following corollary is a direct consequence of this remark.

Corollary 1 *In order to expose secret tag information, the adversary must observe a sufficiently high number of honest protocol runs between authorized reader-tag pairs.*

This implies that adversaries, regardless of their computational power, must rely on authorized reader-tag interactions to have a chance of inferring secret information.

Assumption 1 *For the rest of the paper, we will adopt the assumption that observing enough protocol runs to expose the value of a nonce is impractical in low-cost RFID systems.*

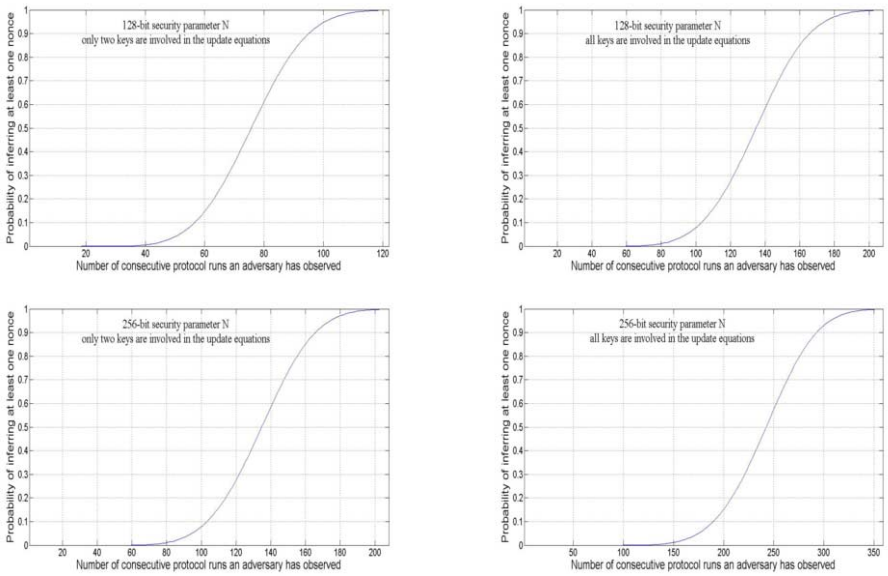


Figure 2. The probability of exposing at least one nonce as a function of the number of consecutive protocol runs observed by the adversary, for different size of security parameter and different number of parameters involved in the update equation.

3.3. Security of Mutual Authentication

Before we can state our main theorem regarding the security of mutual authentication in our protocol, we need two more lemmas.

Lemma 6 *Under Assumptions 1, given that the reader generates random nonces, no information about the secret key, K, is revealed by observing protocol runs of the proposed protocol.*

Proof: We start with the basic assumption that the key is loaded to the tag secretly; that is $k_a^{(0)}$, $k_b^{(0)}$, $k_c^{(0)}$, and $k_d^{(0)}$ are secret. By Lemma 5, messages $B^{(0)}$ and $C^{(0)}$ provide perfect secrecy. That is, no information about the nonce, $n^{(0)}$, nor the keys, $k_b^{(0)}$ and $k_c^{(0)}$, will be leaked by $B^{(0)}$ and $C^{(0)}$. Now, $n^{(0)}$ will be used to generate $D^{(0)} = n_\ell^{(0)} \oplus k_d^{(0)}$ and $A^{(1)} = n_\ell^{(0)} + (n_r^{(0)} \oplus k_a^{(0)})$. Since $n^{(0)}$ is delivered in a perfectly secret manner, and $k_d^{(0)}$ and $k_a^{(0)}$ are secret, no information will be revealed by the observation of $D^{(0)}$ and $A^{(1)}$. (The proof is very similar to the proof of Lemma 5; it is based on the fact that $k_a^{(0)}$ and $k_d^{(0)}$ are random and independent.)

So far, no secret information about the initial key $K^{(0)}$ nor the nonce $n^{(0)}$ has been revealed. Therefore, there is no information leakage about the updated subkeys $k_a^{(1)} = n_r^{(0)} \oplus k_a^{(0)}$, $k_b^{(1)} = k_u^{(0)} + (n^{(0)} \oplus k_b^{(0)})$, $k_c^{(1)} = k_u^{(0)} \times (n^{(0)} \oplus k_c^{(0)})$, and $k_d^{(1)} = n_r^{(0)} \oplus k_d^{(0)}$. Given that the keys are updated to remain independent and to have the same distribution as the outdated keys, and that $n^{(1)}$ is random and independent from the previous nonce and from the secret keys, the proof follows by induction (given Assumption 1). ■

Lemma 7 *Under Assumption 1, an adversary making q_q Query oracles, q_s Send oracles will succeed with probability at most*

$$\max\{\frac{q_q}{p-1}, \frac{q_s}{2^N}\}. \tag{29}$$

Proof: By Lemma 6 and Corollary 1, calling the *Execute* oracle a practical number of consecutive times is of no help to the adversary, since no information is leaked by observing messages exchanged between authorized RFID pairs.

On the other hand, an adversary calling the *Query* oracle will receive A as the tag's response. Depending on the adversary's response, the tag will respond with message D with probability $1/(p-1)$ (the probability of successful forgery by Lemma 3), or abort the protocol with probability $(p-2)/(p-1)$. If the tag does respond, the protocol is considered broken. However, upon unsuccessful forgery, the tag will abort, and responds to the next *Query* with the same identifier, A. Therefore, no information about the tag's secret key is revealed by multiple *Query* calls.

Finally, an adversary calling the *Send* oracle to impersonate a valid tag will be successful with probability at most $1/2^N$. This is due to the fact that A might or might not be a valid tag identifier. If it is not, the reader will abort the protocol. Assume, however, that A is a valid identifier (the adversary can obtain a valid one by interrogating a tag in the system). An authorized reader, responding with B and C, will accept D if and only if $D = n_\ell \oplus k_d$. To extract the correct n_ℓ, however, the adversary must know k_b or k_c, which are kept secret by Lemma 6. Moreover, k_d is unknown to the adversary (also by Lemma 6). Hence, the adversary's probability of success is $1/2^N$, and the lemma follows. ■

Given that p is a $2N$-bit prime integer, the adversary's probability of falsely authenticating herself to a valid tag is at most $1/2^{2N-1}$, and the adversary's probability of authenticating herself to a valid reader is $1/2^N$. That is, the probability of mutual authentication when the protocol is not honest is negligible in the security parameter N. We can now state our main theorem.

Theorem 1 *Under Assumption 1, the proposed UCS-RFID is a secure mutual authentication protocol for RFID systems.*

Proof: Lemma 6 implies that the first condition of Definition 1 is satisfied. The second condition of Definition 1 can be easily verified; it merely means that if the messages exchanged between legitimate RFID pairs are relayed faithfully to one another, mutual authentication is achieved. The third condition of Definition 1 is shown to be satisfied in Lemma 7. Thus, all three conditions of Definition 1 are satisfied. ■

3.4. Desynchronization Attacks and the Update Procedure

It is critical to point out the importance of the key update procedure to the security of our protocol. Both the tag and the reader must update their parameters for the security to hold.

Consider an adversary blocking message A and replaying it to the reader. Since the adversary does not know k_b, k_c nor k_d, she cannot extract the correct n and generate a valid D with a non-negligible probability.

Consider an adversary blocking messages B and C, and replaying them to the tag. Of course, the adversary will be authenticated. However, this is considered as faithfully relaying messages, which does not affect the honesty of the protocol. This makes sense, because the tag will respond with a message D which does not reveal extra information about the tag that has not been revealed by A.

Consider an adversary blocking message D sent to the reader. The reader will assume that the tag has not updated its parameters while, in fact, it has. Consequently, the secret keys at the tag's side will be different than the secret keys stored at the database, causing a possible desynchronization between the tag and the reader. A solution to this problem is that the reader updates the parameters even if it does not receive message D from the tag. The reader, however, must store both the updated and the outdated parameter values at the database to count for the possible scenario that the tag has not updated its parameters.

A more dangerous attack can be launched by blocking messages B and C sent to the tag, or message D sent to the reader, if the *same* keys, with *different* nonces are used. Let $B^{(1)} \equiv n^{(1)} + k_b^{(1)} \mod p$ and $C^{(1)} \equiv n^{(1)} \times k_c^{(1)} \mod p$ be blocked by an active adversary. If the same keys $k_b^{(1)}$ and $k_c^{(1)}$ are used to generate $B^{(2)} \equiv n^{(2)} + k_b^{(1)} \mod p$ and $C^{(2)} \equiv n^{(2)} \times k_c^{(1)} \mod p$, the difference between the two nonces, $n^{(1)}$ and $n^{(2)}$, is simply the difference between $B^{(1)}$ and $B^{(2)}$. It can be easily seen that:

$$\begin{aligned} C^{(2)} \equiv n^{(2)} \times k_c^{(1)} &\equiv (n^{(1)} + \delta) \times k_c^{(1)} \\ &\equiv (n^{(1)} \times k_c^{(1)}) + (\delta \times k_c^{(1)}) \equiv C^{(1)} + (\delta \times k_c^{(1)}) \mod p. \end{aligned} \quad (30)$$

Hence, with the knowledge that $n^{(2)} \equiv n^{(1)} + \delta \mod p$, where $\delta \equiv B^{(2)} - B^{(1)} \mod p$, the value of $k_c^{(1)}$ can be easily computed as $k_c^{(1)} \equiv (C^{(2)} - C^{(1)}) \times \delta^{-1} \mod p$. Thus, we emphasize that whenever the reader receives an outdated identifier, the reader retransmits the same messages $B^{(1)}$ and $C^{(1)}$, as opposed to generating a new nonce and transmitting $B^{(2)}$ and $C^{(2)}$ as above.

The requirement that the reader responds with the same B and C when receiving an outdated A, however, introduces a vulnerability to a man-in-the-middle (MITM) attack. Consider an adversary observing messages $A^{(1)}, B^{(1)}, C^{(1)}$, and then intercepting message $D^{(1)}$. The reader will assume that the tag has not updated its parameter. Hence, the adversary can impersonate the tag by sending its $A^{(1)}$ and, upon receiving the same $B^{(1)}$ and $C^{(1)}$, she can replay the intercepted $D^{(1)}$, which will be accepted by the reader.

Fortunately, there is an easy fix for this vulnerability. Whenever a valid reader receives an outdated identifier $A^{(1)}$, it responds with the same $B^{(1)}$ and $C^{(1)}$ to avoid key exposure (as discussed above). But the tag does *not* get authenticated upon the reception of $D^{(1)}$ (to avoid the man-in-the-middle attack described above). The reader continues by carrying out another protocol run with the tag (with updated keys this time), and *only* if the second authentication run is passed, with updated parameters to generate $A^{(2)}, B^{(2)}, C^{(2)}$, and $D^{(2)}$, the tag is authenticated.

4. Conclusion and Future Work

In this paper, a new direction into the problem of authenticating low-cost RFID systems is proposed. The aim of this paper was to investigate the possibilities of unconditional

security in the design of RFID protocols. An instance of such protocols was proposed. Under a restriction on the number of consecutive protocol runs an adversary is assumed to observe, the proposed protocol is shown to achieve unconditional secrecy and unconditional integrity.

References

[1] B. Alomair, L. Lazos, and R. Poovendran. Passive attacks on a class of authentication protocols for RFID. *International Conference on Information Security and Cryptology – ICISC'07*, 2007.

[2] B. Alomair and R. Poovendran. On the Authentication of RFID Systems with Bitwise Operations. *New Technologies, Mobility and Security, 2008. NTMS'08.*, 2008.

[3] G. Avoine. Adversary model for radio frequency identification. Technical report, Swiss Federal Institute of Technology (EPFL), Security and Cryptography Laboratory (LASEC), 2005.

[4] G. Avoine, K. Kalach, and J.-J. Quisquater. ePassport: Securing international contacts with contactless chips. In *Financial Cryptography and Data Security – FC'08*, 2008.

[5] G. Avoine and P. Oechslin. A Scalable and Provably Secure Hash Based RFID Protocol. In *International Workshop on Pervasive Computing and Communication Security – PerSec 2005*, 2005.

[6] M. Bellare and P. Rogaway. Entity authentication and key distribution. In *Advances in Cryptology-CRYPTO'93*, 1993.

[7] A. Bogdanov, G. Leander, C. Paar, A. Poschmann, M. J. Robshaw, and Y. Seurin. Hash Functions and RFID Tags : Mind The Gap. In *Proceedings of the 10th International Workshop Cryptographic Hardware and Embedded Systems, CHES'08*, 2008.

[8] J. Bringer and H. Chabanne. Trusted-HB: A Low-Cost Version of HB^+ Secure Against Man-in-the-Middle Attacks. *IEEE Transactions on Information Theory*, 2008.

[9] J. Bringer, H. Chabanne, and D. Emmanuelle. HB^{++}: a Lightweight Authentication Protocol Secure against Some Attacks. In *Security, Privacy and Trust in Pervasive and Ubiquitous Computing-SecPerU'06*, 2006.

[10] J. Bringer, H. Chabanne, and T. Icart. Improved Privacy of the Tree-Based Hash Protocols Using Physically Unclonable Function. In *Security and Cryptography for Networks-SCN'08*, 2008.

[11] T. Cover and J. Thomas. *Elements of Information Theory*. Wiley-Interscience New York, 2006.

[12] S. Dominikus, E. Oswald, and M. Feldhofer. Symmetric Authentication for RFID Systems in Practice. Handout of the Ecrypt Workshop on RFID and Lightweight Crypto, 2005.

[13] M. Feldhofer, S. Dominikus, and J. Wolkerstorfer. Strong authentication for RFID systems using the AES algorithm. In *Workshop on Cryptographic Hardware and Embedded Systems – CHES 2004*, 2004.

[14] M. Feldhofer, J. Wolkerstorfer, and V. Rijmen. AES Implementation on a Grain of Sand. *IEE Proceedings - Information Security*, 2005.

[15] W. Feller. *An Introduction to Probability Theory and its Applications*. Wiley India Pvt. Ltd., 2008.

[16] H. Gilbert, M. Robshaw, and Y. Seurin. HB#: Increasing the Security and Efficiency of HB. 2008.

[17] H. Gilbert, M. Robshaw, and H. Sibert. An active attack against HB^+ – a provably secure lightweight authentication protocol. Manuscript, 2005.

[18] O. Goldreich. *Foundations of Cryptography*. Cambridge University Press, 2001.

[19] P. Golle, M. Jakobsson, A. Juels, and P. Syverson. Universal re-encryption for mixnets. In *The Cryptographers' Track at the RSA Conference – CT-RSA*, 2004.

[20] J. Gubner. *Probability and Random Processes for Electrical and Computer Engineers*. Cambridge University Press, 2006.

[21] T. S. Heydt-Benjamin, D. V. Bailey, K. Fu, A. Juels, and T. O'Hare. Vulnerabilities in First-Generation RFID-Enabled Credit Cards. Manuscript, 2006.

[22] A. Juels and R. Pappu. Squealing euros: Privacy protection in RFID-enabled banknotes. In *Financial Cryptography – FC'03*, 2003.

[23] A. Juels and S. Weis. Authenticating pervasive devices with human protocols. In *Advances in Cryptology – CRYPTO'05*, 2005.

[24] J. Lee and Y. Yeom. Efficient RFID Authentication Protocols Based on Pseudorandom Sequence Generators. Cryptology ePrint Archive, Report 2008/343, 2008.

[25] T. Li and R. H. Deng. Vulnerability analysis of EMAP - an efficient RFID mutual authentication protocol. In *Second International Conference on Availability, Reliability and Security – AReS 2007*, 2007.

[26] D. Molnar and D. Wagner. Privacy and Security in Library RFID: Issues, Practices, and Architectures. In *Conference on Computer and Communications Security – ACM CCS*, 2004.
[27] M. O'Neill (McLoone). Low-Cost SHA-1 Hash Function Architecture for RFID Tags. In *Workshop on RFID Security – RFIDSec'08*, 2008.
[28] K. Ouafi, R. Overbeck, and S. Vaudenay. On the Security of HB# against a Man-in-the-Middle Attack. In *Advances in Cryptology - Asiacrypt 2008*, 2008.
[29] P. Peris-Lopez, J. C. Hernandez-Castro, J. Estevez-Tapiador, and A. Ribagorda. LMAP: A real lightweight mutual authentication protocol for low-cost RFID tags. Workshop on RFID Security – RFIDSec'06, 2006.
[30] P. Peris-Lopez, J. C. Hernandez-Castro, J. Estevez-Tapiador, and A. Ribagorda. M2AP: A minimalist mutual-authentication protocol for low-cost RFID tags. In *International Conference on Ubiquitous Intelligence and Computing – UIC06*, 2006.
[31] C. Shannon. *Communication Theory and Secrecy Systems*. Bell Telephone Laboratories, 1949.
[32] D. Stinson. *Cryptography: Theory and Practice*. CRC Press, 2002.
[33] I. Vajda and L. Buttyán. Lightweight authentication protocols for low-cost RFID tags. In *Second Workshop on Security in Ubiquitous Computing – Ubicomp 2003*, 2003.
[34] S. Weis, S. Sarma, R. Rivest, and D. Engels. Security and privacy aspects of low-cost radio frequency identification systems. In *International Conference on Security in Pervasive Computing-SPC'03*, 2003.

A. An Example of Multiplication with Cheap Circuitry

For completeness of presentation, we show here a simple algorithm that multiplies two numbers with minimum circuitry, in expense of time efficiency.

To multiply two N-bit integers, the two operands are stored in two registers R_1 and R_2, while a third register R_3, used to store a temporary partial sum, is initialized to zero. The algorithm starts by examining the least significant bit of R_2; if it is one, then R_1 is added to R_3. The register R_1 is then shifted one bit to the left, which corresponds to multiplying the operand stored in R_1 by two. If the second least significant bit of R_2 is one, then R_1 is added to R_3 and then R_1 is shifted. If it is zero, then the bits in R_1 are shifted one bit to the left without addition. After the $(i-1)^{th}$ shift of R_1, the i^{th} least significant bit of R_2 is examined; if it is one, then the value in the register R_1 is added to the temporary sum in register R_3, and so forth, until the bits in R_1 are shifted N times. The value stored in R_3 is the result of multiplying the two integers modulo p.

The addition algorithm `ADD` can be any modular adder. The multiplication algorithm `MULT` is shown in Algorithm 1.

Algorithm 1 `MULT`(a, b)

$R_1 \leftarrow a$;
$R_2 \leftarrow b$;
$R_3 \leftarrow 0$;
for $i = 0, ..., N-1$ **do**
  if $R_2[i] = 1$ **then**
    $R_3 \leftarrow$ `ADD`(R_3, R_1);
  end if
  $R_1 \leftarrow$ `ShiftLeft`(R_1);
end for
return R_3

Therefore, by implementing algorithm `MULT` described above, multiplication of two elements of the field $\mathbb{Z}_p$ can be performed using only three N-bit registers and one modular adder.

This page intentionally left blank

Radio Frequency Identification System Security
Y. Li and J. Zhou (Eds.)
IOS Press, 2010

doi:10.3233/978-1-60750-485-6-19

The Case for Dynamic RFID Tag Authentication

M.J.B. ROBSHAW [a] and A. POSCHMANN [b]

[a] *Orange Labs, 38-40 rue du Général Leclerc, Issy les Moulineaux, France*
[b] *Division of Mathematical Sciences, Nanyang Technological University, Singapore*

Abstract. RFID tag authentication is widely viewed as a valuable tool in the fight against product counterfeiting. In this paper we describe some of the different approaches that have been proposed and we focus on what is arguably the most secure; that of dynamic tag authentication. We highlight different ways of supporting dynamic tag authentication and provide the latest implementation results. The net result is that dynamic authentication using on-tag cryptography is a reality and should be considered for deployment before other less secure options. As a side-result we note that the on-tag overhead when supporting an asymmetric rather than a symmetric cryptographic solution can be surprisingly light.

Keywords. RFID, tag, security, cost, authentication.

Introduction

Radio frequency identification (RFID) has been used for many years in such diverse applications as tracking livestock and for ticketing on the metro [16]. Today many new applications are being made possible by major advances in the manufacture of cheap RFID devices. These devices can be manufactured by the million and interrogated at speed over several metres and several new applications are centered around their deployment in the supply chain. By necessity these devices are very limited and it is common to refer to them as "tags". Indeed tags are so basic that even the simplest contactless smart cards used in transport applications will be more powerful than the tags we typically encounter in supply chain management.

The tags of interest to us are, in effect, those supported by the industry initiative EPCglobal [8]. The latest version of the EPCglobal tag-interface standard describes the communications protocol for Class-1 Generation-2 UHF tags [9] where "Generation-2" distinguishes this work from earlier standards developed in the Auto-ID center and "Class-1" indicates a basic passive RFID tag. Such tags are limited though they can support a moderate amount of additional memory. The security currently supported in these tags is limited to very basic features such as fixed passwords to control access to read/write memory and to operate the KILL command. While there is some work in the industry on HF-based solutions, for read-at-a-distance applications UHF will dominate and, in fact, it is UHF-based tags that pose the most challenging deployment environment for additional functionality such as security.

The application of interest to us in this paper is the use of RFID tags as an anti-counterfeiting tool. It has been reported [31] that already 11% of global pharmaceutical commerce is counterfeit, which translates to a financial sum of $39 billion. In other high-value industries there are similar problems, some with fatal consequences as a range of accidents attributed to fake automobile, helicopter, and plane parts illustrate [11]. The motivation, therefore, is clear and can only get stronger.

1. Tags and capabilities

It is often stated that adding security to RFID tags is difficult because the cost of the tag limits the amount of silicon available. While this captures part of the problem, hardware specialists observe that " *[. . .] silicon is actually not the dominant expense if done well* " [25]. Instead the difficulties are both varied and complex, and among them we have the issue of power or, rather, the lack of it.

The amount of power that can be emitted by a reader is limited by legislation. A tag has a working range, and at the furthest extremes of that range we can calculate the power received at the tag antenna. Such power estimates would be *ideal case* and may be impacted by the physical environment. The power received is converted for use by the tag, with efficiency losses, and power is consumed by the analog components and the essential digital components. Whatever power is left, which is sometimes estimated to be less than 15% of the power received at the tag antenna, is available for extra functionality. If we add more functionality than our power budget allows, then we will compromise the read-range of the tag. Incidentally this is the reason we concentrate on longer-range UHF-based applications. According to [14] there can be up to four times more power available for security on an HF-tag placed 1 m from the reader than on an UHF-tag at 5 m. Loosely speaking more power means more room for security.

With a long-range system, *i.e.* where tags can be read at distances over five metres or more, it is very likely that many tags will be within range of a given reader. Indeed the environment can be very complicated with some tags going out of range of one particular reader, others entering, and even some strange interactions between multiple readers are entirely possible. We therefore need a mechanism by which the reader can identify which tags are within range and manage, in an orderly fashion, some way to communicate with a particular tag. This mechanism is referred to as *singulation*. Thus we find that a passive RFID tag must contain some basic components and to be functional it must have: an antenna, an analog component to manage the reader-tag interface and powering the tag, some digital components—such as anti-collision mechanisms—to support essential tag functionality, and some memory. It is entirely possible that these are the only components on a tag; indeed this is almost certainly the case for the cheapest tags on the market.

Once the basic features are supported the application will dictate whether any optional extras need to be added, *e.g.* read-write tag memory or features for tag security. Indeed one current trend is for more advanced tags to support writable memory so that an application can add information to the tag as it moves through the supply chain.

While it is well-known that RFID tags are particularly limited devices, it is not always clear what deployment constraints we face. With regards to space, the typical unit of measurement is the *gate equivalent* (GE) which is the amount of physical space occupied by the logical NAND gate. Even though the physical space of an implementation

will vary between fabrication technologies, the relative size of an implementation and the basic NAND gate is anticipated to stay reasonably constant. It turns out, after considering a variety of issues, that around 2 000 to 3 000 GE are often quoted as being economically available for security features on the tag. While there is inevitably some vagueness about such an estimate, it offers a useful guide in differentiating between proposals that might or might not be suitable for tags, at least with regards to this one measure. Estimates for the power available range between 4 μW to 15 μW, though there are complicated dependencies that ensure these bounds are also rather crude.

1.1. Yet more constraints

While space and time are probably the most striking limitations there are others. For instance, to reduce the energy consumption on the tag we would probably want to clock the digital components of the tag at a slower rate. A typical clocking rate for RFID tags is 100 KHz, but a low clocking rate means that a computation will take more time. This might have an adverse impact on higher-level protocols or on the number of items that can be reliably read over a given time. The type of tag-reader interaction can also be important and there is a difference between an application that merely requires a tag to identify itself and then to keep quiet, and an application that requires information to be written to a tag or for a tag to respond to some query. However, provided the bandwidth requirements are reasonable, this need not be a major burden.

The final constraint to address is the most important. The sole reason to add cryptography is to provide security, yet the levels of security we see in Internet applications are, for the most part, inappropriate for tag-based deployments. This is particularly the case when excess security leads to additional financial costs. Instead it is widely accepted, for instance in the eSTREAM project [47], that 80-bit security is likely to offer a reasonable level of security for such constrained devices.

2. Tag authentication

Our starting point, and indeed the whole purpose behind efforts such as EPCglobal, is a mechanism for tag/product identification. The essential purpose of the tag is to send an identifier to the reader. This allows the tag, and the product to which it is attached, to be followed through the supply chain. The *electronic product code* (EPC) number is therefore a mechanism for *identification*.

However the EPC number can be readily copied and programmed into a blank tag so if we are to use the RFID tag as a tool to authentication then we need some additional evidence that the tag is genuine. There are a variety of ways of doing this[1] and these are described in an impressive series of white papers [2,11,12,32] from the Bridge project. There is however one important omission that this paper will cover.

Within [11,12] and elsewhere, it is typical to make a distinction between "weak" and "strong" authentication where an example of the former is the use of passwords and an example of the latter is the use of cryptography. However in this paper, we prefer to use a classification that closely matches the evolution of security features in the credit

[1]Note that we don't consider physical measures for product authentication such as holograms *etc.*

card industry. Indeed, in the fight against anti-counterfeiting this is perhaps the closest precedent that we have and it makes sense to learn from that experience.

In this paper, therefore, we will separate the different approaches to tag authentication into the three groups we present below. They are presented in order of improved security, with unprotected static data being augmented with a static authentication mechanism and then, for better protection, replaced by a dynamic solution.

2.1. Network solution

For this first approach, we assume that a reader has access to an on-line database that indicates whether the same tag, *i.e.* a second reading of the same EPC number, has been simultaneously seen in different parts of the network. This is sometimes referred to in the literature as *location-based* authentication [12] or sometimes as *track-and-trace*.

The main advantage of this mechanism is that tags can be used *as is* and there is no need for additional functionality on the tag. However there are disadvantages. The network along with an appropriate database needs to be continually available and if an anomaly is detected, for instance there are two sightings of the same tag, then it might not be easy to decide which is the correct one. This gives rise to an area of study referred to as *rule-based* anti-counterfeiting where increasingly sophisticated rules can be used to track and interpret events in the supply chain [2].

In fact it appears that, almost intrinsically, protection can only ever be partial in this kind of solution since information can still be copied at a distance from a genuine tag and put in a fake tag. Depending on the rules and the sophistication of the techniques deployed, whenever a genuine tag is invisible to the network, for instance it is stationary or a package is in storage, then counterfeit products with a duplicate tag can be introduced into the system. Even though anomalies could later be discovered, the damage to some extent is done. In general, it seems that this solution falls short of what might be needed and the Security Group of the Bridge project comment that "*it is not enough for many applications that cloned tags in supply chains are detectable* " [1].

2.2. Static authentication

In this solution the tag carries information that has been cryptographically generated and can be readily verified. Tag manufacturers usually provide a tag or transponder identifier, something called the *tag identification number (TID)*, which is fixed in the tag at the time of manufacture. In its basic form this identifies the manufacturer and model of the tag. But by using additional user memory this can be extended to become a unique identifier for each tag. Note that this has nothing to do with the EPC number, but it is an important feature of the tag and typically, for static authentication, some combination of ownership information, EPC number, and/or TID would be digitally signed with the signature being stored on-tag.

The argument goes that even though the signed data can be copied, such a signature cannot be used in a new tag since a different tag would have a different TID. Static authentication should offer a better solution than the networked solution since the authentication information can be verified off-line by the reader and a fake tag immediately identified. A small price to pay is that we have introduced a cryptographic mechanism and this will need an accompanying infrastructure. Here we will need a *public key infrastructure* (PKI) though this need not be overly complex for some applications.

At first sight it seems likely that a signed TID will offer some limited success against counterfeiting. Indeed, some companies have already started prototyping solutions with the digital signature method typically being based on elliptic curves [6]. At the same time, several RFID chip manufacturers have announced extended support of the TID down to the level of not just identifying the manufacturer or model of the tag, but also using additional user memory to support individual tag numbers.

However the use of TIDs is not without problems, as has been observed elsewhere [2, 33]. The most important disadvantage is that of relying on the difficulty of copying or altering a TID. If TIDs can be altered, or if a fake device can be programmed to respond with a copied signature and a false TID then the system fails. Note that it is possible that if such a device is not visible then it need not even look like an RFID tag provided it emulates the correct response [33].

2.3. Dynamic authentication

The third approach, dynamic authentication, offers the most secure solution and this coincides with what other work has sometimes called "strong" authentication. In fact looking at the evolution of attacks in other fields, it is the only long-term solution. Indeed, while much industry focus appears to be on static authentication Alien Technology has announced an EPC-compliant UHF tag that supports dynamic authentication [41] though no technical details are available.

Dynamic authentication, like static authentication, uses cryptography. But as the name implies the tag actively computes something instead of merely passing on static data. There is an additional cost since the tag now needs to support additional circuitry and store secrets, though this is possible at a reasonable cost. In return the solution is very secure. Eavesdropping on the tag-reader interaction will, for most authentication protocols, give the adversary no advantage since such information is only relevant to that authentication session. It cannot be re-used. At the same time the reader interrogating the tag is assured, at the end of the session, that the tag contains the correct secret. In Section 4 we will show how such dynamic authentication can be achieved in both a symmetric and an asymmetric setting. We will also show that these solutions can satisfy the restrictive space and power constraints that were set out in Section 1.

3. Dynamic tag authentication

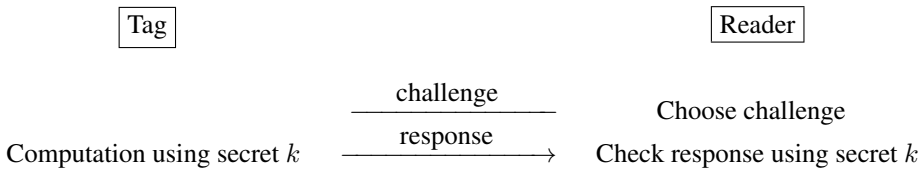
Mechanisms for dynamic tag authentication will involve a message exchange along with some form of computation on the tag and verification on the reader. In [12] it is suggested that all tag authentication protocols are *challenge-response* protocols but this is not strictly the case as we will show. And we will see that while the security of dynamic authentication can sometimes be reduced to that of an underlying cryptographic primitive, there are alternative approaches that reduce the security of the protocol directly to a hard problem.

Cryptographic primitives can be organised according to the way they use key material. Mechanisms that require the participants to share the same secret key are referred to as *secret key* or *symmetric* algorithms. Mechanisms that allow participants to use different key material are referred to as *public key* or *asymmetric* algorithms. Both types

of cryptography require a supporting infrastructure. However, given the essential nature of an RFID-based deployment with many (potentially unknown) players being involved, lightweight public-key cryptography is generally viewed as particularly attractive.

3.1. Symmetric solutions

For our purposes, the most useful primitive from the field of symmetric cryptography is the block cipher. Not only can a block cipher be used directly for encryption, but when used in an appropriate way it can be used to construct all the other symmetric primitives. More importantly for this paper it can also be used for tag authentication within a *challenge-response* protocol.

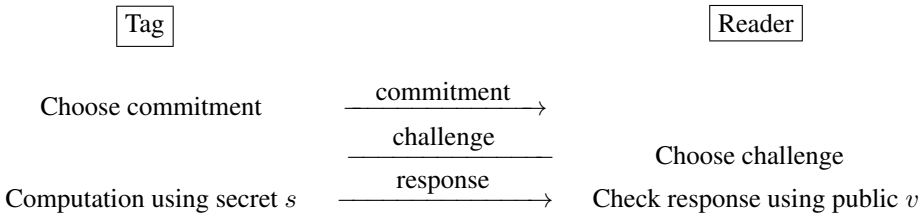


The prominence of block ciphers has lead to much research in the area [44]. One of the most promising recent developments is the block cipher PRESENT [4] which was specifically designed for constrained hardware. However research continues and new proposals such as the KATAN family [5] have been made.

3.2. Asymmetric solutions

While some future applications involving sensor networks are sometimes cited, there are currently no significant calls to add public-key *encryption* to RFID tags. This is in contrast to the case of digital signatures which could be used to demonstrate the authenticity of a tag if used in a challenge-response protocol (as in a similar way to block ciphers). The most promising candidate for dynamic signatures on constrained devices would probably be ECDSA [38], the elliptic-curve variant of the DSA signature scheme. However, with estimates for the EC engine alone running above the 10 000 GE threshold [3] the physical costs are likely to remain too large for deployment on passive tags for some time to come. Some figures for the encryption version of NTRU [40] appear to indicate a relatively small implementation footprint [17] but this is for a reduced-security variant and carries a significant time penalty.

Instead it is worth us considering *identification schemes*. While these schemes can be converted to give digital signatures [36], such a conversion is rarely done in practice and identification schemes are deployed on their own terms. In particular, these schemes allow a tag to "prove" that it contains a tag-specific secret during an interactive *commitment-challenge-response* protocol with the reader.



Oren and Feldhofer [42] propose an interesting public key identification scheme called WIPR that is based on the randomized variant of the Rabin cryptosystem. Their ASIC implementation requires 5 705 GE and 66 048 clock cycles which is too large (and too slow) to be considered truly viable. Some optimisations have been proposed [49], though the resultant reduced area requirement appears to be around 4 700 GE. Instead we will look elsewhere and it seems that one of the most promising options for on-tag deployment is CRYPTOGPS [22]. This is what we will use in Section 4.2.

4. Two practical methods for dynamic tag authentication

In this section we describe two techniques for providing dynamic tag authentication. One uses symmetric cryptography and the other uses asymmetric cryptography. For both we provide the latest implementation results that come from full silicon fabrication, and we confirm that both solutions are perfectly feasible within our space and power budget for passive tags.

4.1. Using PRESENT

The use of a block cipher for tag authentication has already been considered elsewhere [13] and some very impressive implementation optimisations for the AES [37] have been described with the latest figures showing that the AES with 128-bit keys can be realized with as few as 3 100 GE [26]. In addition, papers such as [15] describe the results of the full fabrication process. While very impressive, such implementation requirements are fractionally too great to really be considered for passive UHF tags. In particular it is not clear that a security level of 128 bits is really appropriate.

Recently there has been much advance in the design of lightweight primitives and instead of using the AES it is interesting to use a block cipher such as PRESENT. Clearly the two algorithms cannot be considered to be in the same league with regards to independent scrutiny, but interest in PRESENT has been growing and PRESENT has been proposed for inclusion in a new ISO standard on lightweight cryptography which is now moving on to work in progress [30]. Certainly, for the task at hand, PRESENT is a suitable solution offering a good security/performance breakpoint and the algorithm was deliberately designed with low-cost tags in mind.

4.1.1. Description.

A full description of PRESENT is available in [4] and the reader is referred there. We restrict ourselves to saying that PRESENT is a 64-bit block cipher that is particularly

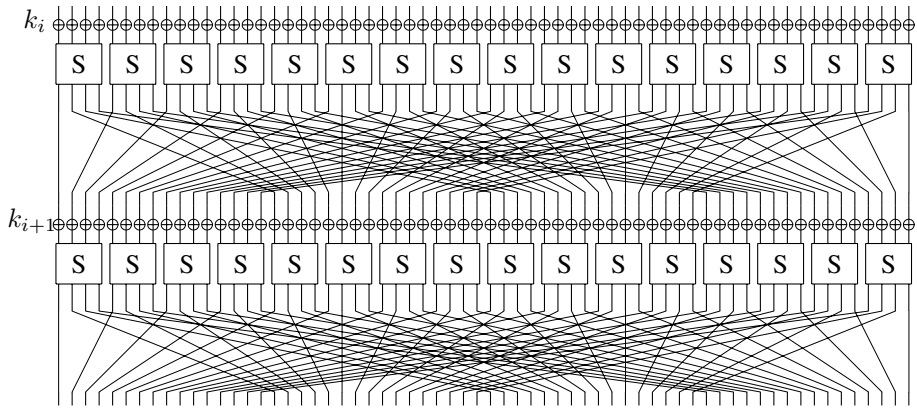


Figure 1. The substitution-permutation network for PRESENT.

designed around 80-bit keys. It is a classical substitution-permutation network [36] using a 4-bit S-box and two rounds of the algorithm are illustrated in Figure 1. By looking at the round structure it is immediately clear that there are two natural implementation approaches; to implement a *round-based* version with a 64-bit data-path or to implement a *serialised* version with a 4-bit data-path. These are the two implementation strategies that are used in the following implementation results.

4.1.2. Implementation results.

In the original paper describing PRESENT synthesis results of an implementation in VHDL for a 0.18 μm UMC technology were presented [4]. These results used a round-based 64-bit datapath for an area-optimized encryption-only version occupying 1 570 GE with a simulated current consumption of 2.8 μA at a frequency of 100 KHz, a supply voltage of 1.8 V and using the smallest wire-load model of that technology (for circuits of around 10 000 GE).

A similar round-based implementation as well as a serialized implementation with a 4-bit datapath using a 0.25 μm IHP technology was reported in [48]. These implementation required 31 (round-based) and 547 (serialized) clock cycles, occupied 1 594 and 1 075 GE and drew 1.9 μA and 1.4 μA, respectively. For both implementations the power consumption was simulated at a frequency of 100 KHz and a supply voltage of 2.5 V.

PRESENT was revisited in [45] as part of a larger system-on-chip ASIC, and there post-synthesis implementation figures using the same 0.25 μm IHP technology were presented. This time two different implementations of PRESENT were tailored to the needs of the ASIC and the parent application. They were fabricated in silicon and the current requirements for the synthesized and fabricated versions were simulated at a frequency of 100 KHz, a supply voltage of 2.5 V, and using the smallest wire-load model of that technology (for circuits of around 5 000 GE). This gave results that ranged between 1.1 μA to 2.1 μA. The area and timing results for all implementations of PRESENT using the 0.25 μm IHP technology are presented below. Full details appear in [45], but it is interesting to note that the latest round-based implementation required an 8-bit I/O interface for the key and the data—due to the demands of the intended application—and this gave a slight area and time overhead.

VARIANT	*round-based*		*serial*	
	area (GE)	*time* (cycles)	*area* (GE)	*time* (cycles)
synthesized [48]	1 594	31	1 075	547
synthesized [45]	1 646	41	1 105	547
fabricated [45]	1 751	41	1 200	547

It is noted in [45] that layout and fabrication produces an overhead that is equivalent to an area increment of between 12 and 18%. This is normal and has been mentioned elsewhere [15]. Indeed it helps to keep this in mind when viewing raw synthesis figures in the literature. Despite such overheads, however, it can be seen that PRESENT would be well-suited for a challenge-response protocol and its performance requirements are so modest that it can be supported on basic UHF tags.

4.2. Using cryptoGPS

The scheme CRYPTOGPS is due to Girault, Poupard, and Stern and it is well-established in the literature [18,22,46]. Several variants are already standardised in ISO/IEC 9798-5 [29] and listed in the final EU NESSIE portfolio [39]. The most efficient variant of CRYPTOGPS, namely that based around elliptic-curves, is currently undergoing standardisation in the new edition of ISO/IEC 9798-5. Over the years, the numerous optimisations [21,23,29] and the performance of CRYPTOGPS have been well-studied by implementors. In short it appears to be very attractive for low cost tags.

4.2.1. Description.

A description of CRYPTOGPS is given in Figure 2 and this is the typical approach for low-cost implementations. Most implementations further incorporate several important optimisations. These include using what is termed a *Low Hamming Weight (LHW) challenge* [21]. Here the reader chooses a challenge that is longer than usual but which has a very low Hamming weight. Since there are few ones in the challenge and they can be judiciously spaced, the multiplication $(s \times c)$ on the tag is turned into a modest number of additions. This allows the on-tag computation to be further optimised. Even though we must increase the length of the LHW challenge to maintain the same security level, the challenge is sparse and allows a variety of compact representations; see [35,45] for more details.

The second optimisation is the use of *coupons*. In [19] Girault described a storage/computation trade-off for CRYPTOGPS. Coupons are a form of pre-computation that is stored on the tag and t coupons, say, (r_i, x_i) for $1 \leq i \leq t$ are computed at the time of manufacture. The form of these numbers is illustrated in Figure 2. Under the assumption that we are required to use more than a handful of coupons ISO 9798-5 proposes to avoid storing the full coupon (r_i, x_i) but to store a partial coupon x_i. We can then use a compact *pseudo-random number generator* PRG to generate the r_i at the time of manufacture and to re-generate the r_i on the tag when needed. Clearly the PRG needs to be sufficiently efficient to be implemented and used on the tag.

Coupons are not always to everyone's taste and some commentators observe that coupons could be consumed in a denial-of-service attack. This is true, though the benefit to the attacker of such an expensive, time-consuming, and non-scalable attack is hard to

Tag	Reader
PARAMETERS	
Curve $\mathcal{C}$, point P	Curve $\mathcal{C}$, point P
KEYS	
Secret key $s \in_R \{0,1\}^\sigma$	Public key $V = -sP$
COUPON PRE-COMPUTATION WITH PRG	
For $0 \leq i \leq t-1$	
Let $r_i = \text{PRG}_k(i)$ where $\lvert r_i \rvert = \rho$	
Set $x_i = \texttt{HASH}(r_i P)$	
Store coupon x_i	
PROTOCOL USING ON-TAG PRG	
At time i fetch x_i $\xrightarrow{x_i}$	
$\xleftarrow{c}$	Pick $c \in_R \{0,1\}^\delta$
Generate $r_i = \text{PRG}_k(i)$	
$y = r_i + (s \times c)$ $\xrightarrow{y}$	$\texttt{HASH}(yP + cV) \stackrel{?}{=} x_i$

Figure 2. Overview of elliptic curve-based cryptoGPS using coupons that are partially re-generated. The parameters ρ, δ, and σ denote three particular bit lengths and these can be adjusted to offer a range of security/performance trade-offs.

articulate. As a result, there are others who observe that coupons are in fact well-suited to typical tag applications. CRYPTOGPS with coupons offers fast power-efficient computation on a very basic tag while using a small amount of user memory. At the same time, most tag-based applications might only require that the tag be verified a moderate number of times, perhaps at significant steps in the supply chain. The tag will then be thrown away or deactivated, as is currently considered in several privacy recommendations [7,10]. In many ways the use of coupons seems to be an ideal match.

4.2.2. Implementation results.

In a selection of previous papers a variety of implementations of CRYPTOGPS have been described. One of the first prototype implementations [20] used an FPGA to simulate the action of a tag. The break-down of the space requirements given in [20] shows that the entire implementation was estimated to require around 6 000 GE of which around 34% (43% including logic and memory) was devoted to supporting CRYPTOGPS. However the PRG used in [20] was not a standard solution, and so other authors [45] have replaced this by PRESENT in an appropriate mode of use.

Since the FPGA prototype, three sets of work have considered the ASIC implementation of CRYPTOGPS [34,35,45]. The two first cited papers considered the requirements of the core cryptographic computation within CRYPTOGPS (*i.e.* the computation of the tag response y) when implemented in 0.18 μm technology. They did not consider the role of the PRG or any supporting logic and the interested reader is referred to those papers for details.

Recently a full implementation of CRYPTOGPS has been completed using 0.25 μm fabrication technologies [45] and the area required for the most useful variants lies between 2 400 and 2 900 GE. A complete breakdown of the implementation figures are

given in [45], though when using a round-based implementation of PRESENT as the PRG CRYPTOGPS occupies 2 876 GE and gives a CRYPTOGPS computation in 724 cycles while a more space-efficient serialized variant occupying 2 433 GE requires 9 319 clock cycles. At the same time, estimates for the current consumption ranged between 1.6 μA to 2.6 μA. Interestingly, given the impact on the computation time, it is not clear that the most space-efficient implementation would be the most appropriate choice in practice.

VARIANT	Without PRG [34,35]		With PRG [45]			
	synthesized		synthesized		fabricated	
	area (GE)	*time* (cycles)	*area* (GE)	*time* (cycles)	*area* (GE)	*time* (cycles)
1-bit	317	1 088	-	-	-	-
4-bit	-	-	2 143	9 319	2 403	9 319
8-bit	431	136	2 433	724	2 876	724
16-bit	900	68	-	-	-	-

5. Discussion and open issues

This paper illustrates some interesting points and leaves some others for further work.

Perhaps the most striking result is that the incremental overhead of supporting CRYPTOGPS, an asymmetric solution, when compared to a symmetric solution such as PRESENT is very small. To see this, we consider the breakdown of implementation requirements for the different components that is described in [45].

COMPONENT	4-bit CRYPTOGPS [45]		8-bit CRYPTOGPS [45]	
	area		*area*	
	(GE)	%	(GE)	%
PRESENT	1 200	49.9	1 751	60.9
Addition	35	1.5	60	2.1
Controller	905	37.7	905	31.5
Storage	263	10.9	159	5.5
Sum	2 403	100.0	2 876	100.0

It is clear that the vast majority of the space is used for the PRG, in this case PRESENT, and the controller. Since these would be required in any symmetric-based solution, the *incremental* overhead in supporting the asymmetric solution CRYPTOGPS is minor when compared to a solution based on PRESENT. Indeed it seems that an asymmetric solution could even be smaller than one based on the AES. Thus it is important to understand that asymmetric (public-key) based solutions are not inherently unsuitable for the dynamic authentication of passive UHF tags.

Instead of using an established algorithm as a plug-in component, CRYPTOGPS illustrates that a protocol can be built around a hard problem. It would, of course, be interesting to replicate this approach in a symmetric key setting, and for this we can turn to the elegant HB-family of protocols. Based on a problem introduced by Hopper and Blum [27], there have been a series of related protocol proposals over recent years [24,28,43] that aim to provide tag authentication in a symmetric key setting, but for which the security is related to the *learning parity with noise* (LPN) problem. This

problem is very attractive in terms of implementation since it requires only simple on-tag computations. But while the latest versions are promising they still fall short of being suitable for deployment.

In this paper we have only considered the issue of tag authentication. However some believe it is important that a tag can check whether it is communicating with a genuine reader. Certainly this may be useful in preventing attacks against on-tag cryptography and it may be useful in preventing privacy violations (since sensitive information need not be released until the tag is sure the reader is legitimate). Thus a mechanism for reader authentication, or mutual authentication when combined with tag authentication, that can be accomplished in a truly satisfying way may also be very useful.

6. Conclusions

In this paper we have considered the dynamic authentication of passive UHF tags. This is a problem that offers considerable potential in the fight against product counterfeiting but which, at the same time, poses some very significant implementation challenges. By referring to the latest results we demonstrate that dynamic tag authentication using strong cryptography is a real possibility. Further we highlight the fact that the incremental overhead in supporting an asymmetric (public-key) solution instead of a symmetric solution is, in terms of on-tag functionality, slight. Indeed the implementation results for both types of solution suggest that dynamic tag authentication should be seriously considered by the RFID tag industry, particularly before opting for less secure partial solutions.

References

[1] M. Aigner, T. Burbridge, A. Ilic, D. Lyon, A. Soppera, and M. Lehtonen. RFID Tag Security, BRIDGE white paper. Available via www.bridge-project.eu.

[2] J. Al-Kassab, M. Lehtonen, and F. Michahelles. Anti-Counterfeiting Prototype Report. BRIDGE white paper, June 2008. Available via www.bridge-project.eu.

[3] L. Batina, J. Guajardo, B. Preneel, P. Tuyls, and I. Verbauwhede. Public-Key Cryptography for RFID Tags and Applications. In P. Kitsos and Y. Zhang, editors, *RFID Security: Techniques, Protocols and System-on-Chip Design*, pages 317–348. Springer, 2008.

[4] A. Bogdanov, L.R. Knudsen, G. Leander, C. Paar, A. Poschmann, M.J.B. Robshaw, Y. Seurin, and C. Vikkelsoe. PRESENT: An Ultra-Lightweight Block Cipher. In P. Paillier and I. Verbauwhede, editors, *Proceedings of CHES '07*, volume 4727 of LNCS, pages 450–466. Springer, 2007.

[5] C. de Cannière, O. Dunkelman, and M. Knezević. KATAN and KTANTAN–A Family of Small and Efficient Hardware-Oriented Block Ciphers. In C. Clavier and K. Gaj, editors, *Proceedings of CHES '09*, volume 5747 of LNCS, pages 272–288. Springer, 2009.

[6] Certicom. Company information available at www.certicom.com.

[7] Commission of the European Communities. Commission Recommendation on the Implementation of Privacy and Data Protection Principles in Applications Supported by Radio-Frequency Identification. May 12, 2009. Available via ec.europa.eu/information_society/policy/rfid/index_en.htm.

[8] EPCglobal. Organisation information available at www.epcglobal.com.

[9] EPCglobal. EPC Radio-Frequency Identity Protocols, Class-1 Generation-2 UHF FRID, Protocol for Communications at 860-960 MHz, version 1.2.0. October 23, 2008. Available via [8].

[10] EPCglobal. General Overview on the European Commission RFID Privacy and Data Protection Recommendation. May 13, 2009. Available via [8].

[11] ETH Zurich and SAP Research. Problem-Analysis Report on Counterfeiting and Illicit Trade. BRIDGE white paper, July 2007. Available via www.bridge-project.eu.

[12] ETH Zurich and SAP Research. Anti-Counterfeiting Requirements Report. BRIDGE white paper, July 2007. Available via www.bridge-project.eu.
[13] M. Feldhofer, S. Dominikus, and J. Wolkersdorfer. Strong Authentication for RFID Systems. *Proceedings of CHES '04*, volume 3156 of LNCS, pages 357–370. Springer, 2004.
[14] M. Feldhofer and J. Wolkerstorfer. Hardware Implementation of Symmetric Algorithms for RFID Security. In P. Kitsos and Y. Zhang, editors, *RFID Security: Techniques, Protocols and System-on-Chip Design*, pages 373–415. Springer, 2008.
[15] M. Feldhofer, J. Wolkerstorfer, and V. Rijmen. AES Implementation on a Grain of Sand. *Information Security, IEE Proceedings*, 152(1):13 – 20, 2005.
[16] K. Finkenzeller. RFID Handbook. Second edition, John Wiley, 2003.
[17] G. Gaubatz, J.-P. Kaps, and B. Sunar. Public key Cryptography in Sensor Networks—Revisited. In C. Castellucia, H. Hartenstein, C. Paar, and D. Westhoff, editors, *Proceeding of ESAS '04*, volume 3312 of LNCS, pages 2–18, Springer-Verlag, 2004.
[18] M. Girault. Self-certified public keys. In D.W. Davies, editor, *Proceedings of Eurocrypt '91*, volume 547 of LNCS, pages 490–497, Springer-Verlag, 1991.
[19] M. Girault. Low-Size Coupons for Low-Cost IC Cards. In J. Domingo-Ferrer, D. Chan, and A. Watson, editors, *Proceedings of Smart Card Research and Advanced Applications*, pages 39–50, Kluwer Academic Press, 2001.
[20] M. Girault, L. Juniot, and M. Robshaw. The Feasibility of On-the-Tag Public Key Cryptography. *RFIDsec 2007*, workshop record. Available via rfidsec07.etsit.uma.es/slides/papers/paper-32.pdf.
[21] M. Girault and D. Lefranc. Public Key Authentication with One (Online) Single Addition. In M. Joye and J.-J. Quisquater, editors, *Proceedings of CHES '04*, volume 3156 of LNCS, pages 967–984, Springer-Verlag, 2004.
[22] M. Girault, G. Poupard, and J. Stern. On the Fly Authentication and Signature Schemes Based on Groups of Unknown Order. Journal of Cryptology, vol. 19, pages 463–487, Springer, 2006.
[23] M. Girault and J. Stern. On the Length of Cryptographic Hash-Values Used in Identification Schemes. In Y. Desmedt, editor, *Proceedings of Crypto '94*, volume 893 of LNCS, pages 202–215. Springer-Verlag, 1994.
[24] H. Gilbert, M. Robshaw, and Y. Seurin. HB$^{\#}$, Increasing the Security and Efficiency of HB. In N. Smart, editor, *Proceedings of Eurocrypt '08*, volume 4965 of LNCS, pages 361–378, Springer, 2008.
[25] R. Glidden, C. Bockorick, S. Cooper, C. Diorio, D. Dressler, V. Gutnik, C. Hagen, D. Hara, T. Hass, T. Humes, J. Hyde, R. Oliver, O. Onen, A. Pesavento, K. Sundstrom, and M. Thomas. Design of Ultra-Low-Cost UHF RFID tags for Supply Chain Applications. *IEEE Communications Magazine*, vol. 42, issue 8, pages 140–151. IEEE Computer Society Press, 2004.
[26] P. Hämäläinen, T. Alho, M. Hännikäinen, and T. D. Hämäläinen. Design and Implementation of Low-Area and Low-Power AES Encryption Hardware Core. In *DSD*, pages 577–583, 2006.
[27] N. Hopper and M. Blum. Secure Human Identification Protocols. In C. Boyd, editor, *Proceedings of Asiacrypt '01*, volume 2248 of LNCS, pages 52–66, Springer, 2002.
[28] A. Juels and S. Weis. Authenticating Pervasive Devices With Human Protocols. In V. Shoup, editor, *Proceedings of Crypto '05*, volume 3126 of LNCS, pages 293–198, Springer-Verlag, 2005.
[29] ISO/IEC 9798: Information Technology – Security Techniques – Entity Authentication – Part 5: Mechanisms using Zero-Knowledge Techniques. Available via www.iso.org.
[30] ISO/IEC 29192-1: Information Technology – Security Techniques – Lightweight Cryptography – Part 1: General.
[31] J. Jenkins, P. Mills, R. Maidment, and M. Profit. Pharma Traceability Business Case Report. BRIDGE white paper, May 2007. Available via www.bridge-project.eu.
[32] M. Lehtonen, J. Al-Kassab, F. Michahelles, and O. Kasten. Anti-counterfeiting Business Case Report. BRIDGE white paper, December 2007. Available via www.bridge-project.eu.
[33] M. Lehtonen, A. Ruhanen, F. Michahelles, and E. Fleisch. Serialized TID Numbers – A Headache or a Blessing for RFID Crackers? In *Proceedings of IEEE RFID 2009*, April 2009.
[34] M. McLoone and M.J.B. Robshaw. Public Key Cryptography and RFID. In M. Abe, editor, *Proceedings of CT-RSA '07*, volume 4377 of LNCS, pages 372–384, Springer, 2007.
[35] M. McLoone and M.J.B. Robshaw. New Architectures for Low-Cost Public Key Cryptography on RFID Tags. In *Proceedings of SecureComm '05*, pages 1827–1830. IEEE Computer Society Press, 2007.
[36] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone. *Handbook of Applied Cryptography*. CRC Press,

Boca Raton, Florida, USA, first edition, 1996.
[37] National Institute of Standards and Technology. FIPS 197: Advanced Encryption Standard, November 2001. Available via csrc.nist.gov.
[38] National Institute of Standards and Technology. FIPS 186-3: Digital Signature Standard. June, 2009. Available via csrc.nist.gov.
[39] NESSIE consortium. Final Report of European Project IST-1999-12324: New European Schemes for Signatures, Integrity, and Encryption (NESSIE), April 2004. Available via https://www.cosic.esat.kuleuven.be/nessie/.
[40] NTRU Corporation. NTRUencrypt. Available via www.ntru.com.
[41] M. O'Conner. Alien Unveils Dynamic Security App for Higgs 3 Chip. *RFID Journal*, September 18, 2009. Available via www.rfidjournal.com/article/5230.
[42] Y. Oren and M. Feldhofer. A Low-Resource Public-Key Identification Scheme for RFID Tags and Sensor Nodes. In *Proceedings of WiSec '09*. ACM Press, 2009.
[43] K. Ouafi, R. Overbeck, and S. Vaudenay. On the Security of HB# against a Man-in-the-Middle Attack. In J. Pieprzyk, editor, *Proceedings of Asiacrypt '08*, volume 5350 of LNCS, pages 108–124. Springer, 2008.
[44] C. Paar, A. Poschmann, and M.J.B. Robshaw. New Designs in Lightweight Symmetric Encryption. In P. Kitsos and Y. Zhang, editors, *RFID Security: Techniques, Protocols and System-on-Chip Design*, pages 349–372. Springer, 2008.
[45] A. Poschmann, M.J.B. Robshaw, F. Vater, and C. Paar. Lightweight Cryptography and RFID: Tackling the Hidden Overheads. In *Proceedings of ICISC '09*, Springer, to appear.
[46] G. Poupard and J. Stern. Security Analysis of a Practical "on the fly" Authentication and Signature Generation. In K. Nyberg, editor, *Proceedings of Eurocrypt '98*, volume 1403 of LNCS, pages 422–436. Springer-Verlag, 1998.
[47] M.J.B. Robshaw. The eSTREAM Project. In M. Robshaw and O. Billet, editors, *New Stream Cipher Designs: The eSTREAM Finalists*, pages 1–6, Springer, 2008.
[48] C. Rolfes, A. Poschmann, G. Leander, and C. Paar. Ultra-Lightweight Implementations for Smart Devices – Security for 1000 Gate Equivalents. In *Proceedings of CARDIS '08*. Springer-Verlag, to appear.
[49] J. Wu and D. Stinson. How to Improve Security and Reduce Hardware Demands of the WIPR RFID Protocol. In *Proceedings of IEEE International Conference on RFID*, Orlando, Florida, USA, April 2009.

Radio Frequency Identification System Security
Y. Li and J. Zhou (Eds.)
IOS Press, 2010

doi:10.3233/978-1-60750-485-6-33

Practical RFID Ownership Transfer Scheme

Ching Yu NG [a], Willy SUSILO [a], Yi MU [a] and Rei SAFAVI-NAINI [b]
[a] *Centre for Computer and Information Security Research (CCISR)*
School of Computer Science and Software Engineering
University of Wollongong, Australia
e-mail: {cyn27, wsusilo, ymu}@uow.edu.au
[b] *Department of Computer Science, University of Calgary, Canada*
e-mail: rei@ucalgary.ca

Abstract. When an RFID tag changes hand, it is not as simply as handing over the tag secret to the new owner. Privacy is a concern if there is no secure ownership transfer scheme to aid the transfer. After sales service and temporary tag delegation are also features commonly seen in such applications. In this paper, we proposed a new RFID ownership transfer scheme that achieves the most security protections and properties in comparison to most of the previous schemes. We also introduced four new security properties that have not been considered before. This opens up new research directions for further development of RFID ownership transfer.

Keywords. RFID, ownership transfer, security model

Introduction

Low-cost RFID tags are very constrained devices that can only carry out basic and simple cryptographic operations. While asymmetric key systems are the norm of modern cryptography, they are considered to be too expensive for these tiny devices. On top of that, RFID tags are always made without tamper-proof protections, which opens the door for adversaries to compromise the tag memory and extract any stored secrets. Solving the privacy issues in RFID systems hence becomes much more challenging in the presence of these unique properties.

We have seen many works in the literature that propose secure RFID tag authentication protocols to preserve privacy of the tag information and privacy of the tag wearer. These protocols differ in efficiencies and their achievable privacy protections under various assumptions, but nearly all of them follow the same system model: there is always a centralized trusted back-end server available for tag responses resolution. Such centralized server model is good for tag secrets management and for RFID products that work within their own domains. In this system, every RFID reader is required to connect (the connection is always assumed to be secure) to the back-end server so that tag responses can be resolved properly using the matching tag secret stored in the server. According to security needs, it may followed with tag secret updates and secret synchronizations between tags and the server.

With more and more individual equipped with personal RFID readers (for smart home appliances as an example) in the future, centralized server model would become less convenient in everyday life. Especially when RFID tagged products have left the point of sales (POS) and reached hands of different individuals. Users should not be forced to make connections to each of the central servers from the respective domain of each RFID tagged product every time the tag is queried. This is not only about reducing the burden of the centralized servers, but also about the trust issues and the privacy of the owners when the product changes hands. In such an environment, users would rather choose to manage the tag secrets of the RFID tagged products they own by themselves. This is where the problem of *Ownership transfer* should be considered: when an RFID tagged product changes hands, the ownership (which effectively means the abilities to read the tag information and manage/update the tag secret) of the RFID tag should be transferred to the new owner (the buyer) from the previous owner (manufacturer, POS or the previous buyer), meaning that all the information about the tag is handed over and the previous owner should forfeit its control over the tag. The problem: "How to preserve privacy (among the owners and away from adversaries) when ownership of an RFID tagged product is transferred?" becomes a new research topic in RFID security and resulted in *Ownership transfer schemes* and its subclass *Delegation schemes*.

1. Previous Works

Compare to authentication/identification protocol researches in RFID, ownership transfer schemes have received less attention in the literature. During our work, we can easily find a lot of works about the former topic, while only around ten pieces of work, to the best of our knowledge, are related to or have mentioned about the latter topic. Here we give a brief overview of them.

Molnar et al. [7,8] are the first to discuss ownership transfer and delegation of RFID tags explicitly along with their pseudonym protocol. *Ownership transfer* and *Controlled delegation* are the new security properties they introduced. In their scheme, a trusted center (TC) manages all the tag secrets in a tree structure. Each tag has one unique key and multiple shared keys with other tags to aid faster tag lookup. Pseudonyms are generated per each query using these keys such that only the TC can disambiguate tag responses and identify each tag. Controlled delegation is done by giving authorized reader a derived key, obtained by running a pseudo-random generator on input the unique key of a tag. The tag will use also the derived key in generating the next q pseudonyms as controlled by an internal non-volatile counter. Delegation expires automatically after q queries. Ownership transfer in fact is done with two controlled delegations. When a tag changes hands, the new owner requests delegation with the TC and asks for the remaining number of delegated tag reads of the previous owner. The new owner then repeatedly queries the tag pass this number or send a new counter value to the tag that is greater than the current value plus the number. This prevents ownership overlap between the new and previous owner. Fouladgar and Afifi use a similar setting as Molnar et al. in [2,3,4] where the role of TC is replaced by a centralized database (CDB). Each tag has an internal counter that increase per each query. Once this counter reaches its fixed maximum value, the current tag key will expire and the CDB must be contacted to renew the tag key. Delegation is done by releasing the current tag key to an verified user by the CDB.

Ownership transfer is done by setting the tag counter to its maximum value first (to invalidate any delegation) and then renew the tag key, followed by a delegation to the new owner.

Since the TC or the CDB still holds all the tag secrets, tag queries made by future owners could still be monitored, which violates their privacy. Lim and Kwon [6] only consider these centralized management methods as temporary ownership transfer schemes and proposed "perfect" ownership transfer, which requires the previous owner to transfer all the tag secrets to the new owner and allow the new owner to secretly update them so that *New owner privacy* is preserved. Saito et al. have a similar idea in [10], however, the security of their scheme is only based on the short read range of the backward channel (tag to reader communication) by assuming that it is hard for adversaries to eavesdrop on this channel.

Instead of using a centralized server, Soppera and Burbridge [12] adopt the scheme of Molnar et al. by replacing the centralized TC with some distributed local devices called RFID acceptor tag such that delegations are done with them instead of the TC. Koralalage et al. [5] also suggest to use some key card reading devices to aid customers to directly overwrite the stored tag secret by swiping an universal customer card and inputting a PIN as the new tag secret. Both of these systems require the distribution of external devices, which adds extra cost and introduces new trust issues.

Previous owner privacy is another important security property in ownership transfer but it has not been addressed properly until Osaka et al. [9] proposed their scheme that preserves both the previous and the new owners' privacy by allowing the previous owner to change the tag key first, then send the new key to the new owner via a secure channel, and finally let the new owner to change the tag key again. This message flow pattern if designed correctly can protect both owners' privacy and we also adopted this pattern in our scheme. However, a flaw in their ownership transfer protocol allows an attacker to break previous owner privacy if the tag is compromised, hence their scheme failed *Forward security*.

Song [11] introduced a new property called *Authorization recovery*. In situations like after sales services or warranty purposes, a tag may be required to send back to its previous owner. This property ensures that ownership recovery is possible and is only temporary (i.e. does not involve another instance of ownership transfer between the current owner and the previous owner). The idea in [11] is fairly simple. The new owner just needs to record the tag key given by the previous owner when ownership transfer is carried out. At times when authorization recovery is needed, the current owner executes the key change protocol with the recorded tag key as input rather than using a random value. As the previous owner knew and recorded such key, his/her ownership is recovered. However, the author failed to achieve this property completely in a sense that the recovered authorization is not temporary. If the current owner wants to take back the ownership from the previous owner, a new instance of ownership transfer protocol has to be executed again. Dimitriou [1] also proposed a similar property called tag release where the current owner can issue a special command to let the tag restores back to its factory default key, which is always stored in the tag memory, allowing the manufacturer to gain back the access to the tag. But then again, to regain the authorization, the current owner requires the manufacturer to delegate the updated tag key to him/her followed by a new instance of ownership transfer.

Recently, Deursen et al. [13] presented a formal model for RFID ownership transfer. They defined secure ownership and exclusive ownership where the former states that the tag holder must be the tag owner and the latter states that there cannot be other tag owners beside the tag holder. However, they did not consider controlled delegation nor authorization recovery where a tag holder may not be a tag owner and hence their model cannot be applied in our scheme as we provide both of these properties.

2. Our Contributions

We propose a new RFID ownership transfer scheme that has all the security properties defined from the previous schemes including : *Controlled delegation*, *Previous owner privacy*, *New owner privacy* and *Authorization recovery*. Also, we firstly introduce four new properties, namely *Tag assurance*, *Current ownership proof*, *Undeniable ownership transfer* and *Owner initiation*. In the following sections, first we give details about the preliminaries to construct our scheme in section 3. Then we present our scheme in the next section. Backed by a security analysis section following the description of our scheme, we conclude this paper in section 6. We stress that readers should follow our models and assumptions in section 3 closely before directly jumping to our scheme in section 4.

3. Preliminaries

In this section, we outline the models, assumptions, security definitions and building blocks that are required to construct our scheme.

3.1. System Model

We use a simplified RFID system model where there are only readers and tags. At the beginning, *the manufacturer* executes $\texttt{SetupReader}(1^k)$ with a security parameter 1^k to properly setup a reader and initialize the system to use any pre-defined protocols. The manufacturer further creates and setups the tags by running $\texttt{SetupTag}(ID)$ with an unique ID for each tag as input, which assigns some unique tag secrets to each of the tags. Each ID and its corresponding tag secret K_{ID}, together with some axillary tag related information $Info_{ID}$ (e.g. product description, origin, manufacture date, etc.) are then stored and maintained in a back-end database server. Whenever a reader requires to interact with a tag, it will execute a tag authentication protocol $\texttt{Auth}()$ by first sending out a query and then relay the tag response to the back-end database server via a secure channel. After the server has processed the response, it will send back the result to the reader. Because of this, the reader and the back-end database server are always regarded as a single entity or simply referred as *the reader*. It is also common to assume that the reader cannot be compromised due to the fact that the secrets are actually stored in the back-end database server. Any user (including attackers) with a compatible reader can also setup their own reader by running $\texttt{SetupReader}(1^k)$ and start interacting with the tags but not accessing the back-end database server. Likewise, any user (mainly attackers) with a compatible tag can setup their own non-legitimate tag by running $\texttt{SetupTag}(ID)$ with some random or chosen ID and start interacting with the legitimate reader. It is assumed that such IDs do not exist in the system database.

3.2. Ownership Transfer Model

In case of ownership transfer, there are new roles we refer to *the previous owner*, *the current owner* and *the buyer/the new owner*. Originally, *the manufacturer* is the first owner of every RFID tag. Every buyer is equipped with his/her own system compatible reader, together with its own back-end database connected via a secure channel (personal readers may even have it installed internally). When an ownership transfer is required, the current owner and the buyer will run our *ownership transfer scheme*. If it is a success, the roles will change: all the tag related secret and other information will be passed along to the buyer, who becomes *the current owner*; the original owner now becomes *the previous owner*. In case of tag delegation, there is an additional role called *the delegate*. Differ from the current owner, who can authenticate the tag and change the tag key to any value, the delegate can only authenticate the tag temporary using the supplied tag key given by the current owner.

3.3. Basic Assumptions

As there can be different settings in the same system model, we have the following basic assumptions to characterize our model from the others. Our scheme and security proofs are also built upon these assumptions.

3.3.1. Capability assumption

We consider RFID tags as very constrained devices. They can at most perform some light-weight cryptographic hashing functions; on the contrary, readers are much more capable to perform more expensive cryptographic operations like asymmetric encryption and decryption, signing and signature verification.

3.3.2. Memory assumption

Tags are also vulnerable to key compromise attack. We always assume all the internal secrets stored in tag memory are also available to competent adversaries. The base requirement of RFID tags is some *incorruptible memory* or *delicate memory*, i.e. adversaries can read the memory by compromising the tag but they lack the ability/tool to corrupt the memory or write back some chosen value, even better is that once the tag is compromised, it will not be functioning anymore. The best they can do is to use the compromised memory content to create a clone by simulating the responses of the compromised tag. Whether this simulation or clone tag can be caught is beyond the scope of our work. Hence we generalize this to an assumption *"once a tag is compromised, its memory can only be read and the tag no longer responses to other commands"*[1]. On the contrary, tags are built with memory update mechanism but it only functions when the pre-defined protocols implemented in the tags are executed and followed faithfully.

[1]This resembles the *forward attacker* as defined in [14], which is the strongest adversary definition for non-PKC capable RFID tags.

3.3.3. Singulation assumption

Unlike tag authentication protocols where the reader needs to search for the correct tag ID from its database by matching the tag response generated by the corresponding tag secret, we assume that in our ownership transfer scheme, there is always a target tag, which has been authenticated already, such that the reader knows exactly the ID of the tag and which tag secret K_{ID} to use to communicate with it. This assumption makes sense as both the owner and the buyer are trading a particular item they are both interested and selected. For this assumption to be applicable, we require the trading item to be authenticated first by $\texttt{Auth}()$ and then singulated from other RFID items so that it will be the sole item involved in the ownership transfer scheme before the scheme can be carried out.

3.3.4. Communication assumption

For the communication between reader and tag, we always assume that all the reader to tag messages can be delivered although these messages can still be eavesdropped, recorded and replayed by adversaries but are never blocked (notice that this does not mean all the reader to tag messages are originated from an honest reader, they can come from the adversaries or replays too). This assumption is logical since the reader always broadcasts strong wireless signals, which is hard to block. Also, due to the previous assumption, the intended recipient tag is always participating in the scheme, which eliminates the situation that the reader is broadcasting valid commands to a fake tag ONLY and resulted in simple record and replay (or relay) attack later on[2].

3.4. Adversary Model

We adopt the adversary model proposed by Vaudenay in [14] and simplify it with the following adversary abilities:

- $\texttt{SetupReader}(1^k)$ allows the creation of a fake reader to interact with other tags.
- $\texttt{SetupTag}(ID)$ allows the creation of a fake tag to interact with the reader.
- $\texttt{SendReader}(m)$ sends a message m to the reader. A reply message m' from the reader may be returned depending on the protocol.
- $\texttt{SendTag}(m)$ sends a message m to a tag. A reply message m' from this tag may be returned depending on the protocol.
- $\texttt{Corrupt}()$ returns all the internal secrets stored inside the tag.

We do not assume users are honest in our system, hence it is possible that either the previous owner, the current owner, the buyer or the delegate is cheating in the scheme. However it is not realistic to consider when both sides are cheating (i.e. at most one adversary during any transaction), otherwise both can simply collogue and there can be no security property enforceable.

[2]In fact, this assumption can be easily removed if we require the tag to generate a random nonce for the reader first, and embed this nonce in the reader to tag message, then the tag can verify the freshness of the message using the embedded nonce. Since this assumption is not too strong, we just leave it here to keep our scheme simple and avoid the necessity of adding a random number generator in a tag.

3.5. Security Properties

We identify the following security properties from previous RFID ownership transfer schemes:

- Previous owner privacy - At the completion of the ownership transfer scheme, the privacy of the previous owner is preserved. Meaning that no future owners can relate or trace back any previous communication between the previous owner and the RFID tag even though a full history of transmitted messages is eavesdropped and recorded.
- New owner privacy - At the completion of the ownership transfer scheme, the privacy of the new owner is preserved. Meaning that no previous owners can relate or track any current communication between the new owner and the RFID tag even though all the transmission is being eavesdropped.
- Controlled delegation - The current owner of the RFID tag has the authority to execute a delegation protocol, which temporary delegates the access right of the tag to anyone without forfeiting the ownership to the tag. The delegate cannot overtake the ownership while the owner can cancel this delegation at anytime. Moreover, the delegation will automatically expires once a pre-determined number of queries value is reached.
- Authorization recovery - The current owner of the RFID tag can allow the previous owner to gain back the access to the RFID tag without going through another instance of the ownership transfer protocol. At the same time, the current owner can cancel the recovered authorization at anytime without the help from the previous owner.

We further introduce four new security properties firstly proposed in this work:

- Tag assurance - During the ownership transfer scheme, the buyer can be assured that the RFID tag undergoing the ownership transfer is the tag claimed by the current owner and requested by the buyer. This property guarantees that the current owner cannot randomly pick any tagged product he/she owns and sells it to the buyer. Together with the assumption 3.3.2, we provide in our scheme a way for the buyer to verify the ID of the tag.
- Current ownership proof - The current owner can prove to any third party that he/she is the current owner of the RFID tagged item.
- Undeniable ownership transfer - The current owner can prove to any third party that the RFID tagged item was owned by a previous owner and the previous owner cannot deny ever owning the tag.
- Owner initiation - The current owner and only the current owner can initiate an ownership transfer, key change and delegation. Unlike most of the other ownership transfer schemes where anyone who holds the current tag key can initiate an ownership transfer, we explicitly limit this to the current owner only (i.e. the delegate is excluded).

3.6. Building Blocks

To build our proposed scheme, we assume there exists a cryptographic hash function $\mathcal{H}() : \{0,1\}^* \rightarrow \{0,1\}^k$ that has the following properties:

- One-wayness - The computation of the hash value is efficient while it is hard to find the pre-image.
- Collision resistance - Given any hash value, it is hard to find another message not equal to the pre-image but gives the same hash value.

We also assume that there exists a public key cryptosystem (PKC) for the users to create publicly verifiable digital signatures such that for any given message m, a public key PK and a corresponding private key SK, we have

$$\sigma = \texttt{Sig}(m, SK) \text{ and } OK \leftarrow \texttt{Ver}(m, \sigma, PK)$$

where $\texttt{Sig}()$ is the signing operation that hash the input message m into proper length and outputs the signature σ signed with the private key SK on the hash of message. $\texttt{Ver}()$ is the signature verification operation that outputs OK if the signature is truly signed with the corresponding private key of the public key PK on m and outputs $\perp$ otherwise. We require that the signatures generated are unforgeable. As one may expected, the signature (together with the assumption 3.3.2) is used to provide current and undeniable ownership proofs.

The PKC is also capable to generate encrypted message from any given message m by an encryption function $\texttt{Enc}()$ using the public key PK and decrypt encrypted message by an decryption function $\texttt{Dec}()$ using the corresponding private key SK. i.e. we have

$$c = \texttt{Enc}(m, PK) \text{ and } m = \texttt{Dec}(c, SK)$$

These functions are only used to establish a secure channel to safely transfer the current tag key from the owner to the buyer. If there exists other form of secure channel (i.e. direct linkage between the readers of the owner and the buyer), these encryption and decryption functions are unnecessary.

Finally, as we mentioned in the assumption 3.3.3, there is a secure RFID authentication protocol $\texttt{Auth}()$ such that after its execution, it outputs $True$ if and only if the tag response r matches with the result generated using K_{ID}, otherwise it outputs $\perp$. Afterward, the real ID of the tag can be looked up by the reader using K_{ID} as the reference key from the back-end database server.

4. Our Ownership Transfer Scheme

We use the building blocks described in section 3.6 to construct our ownership transfer scheme. Our scheme composes of a setup and three protocols: key change protocol, controlled delegation protocol and ownership transfer protocol. Each protocol has its own security goal to achieve. Notice that during the protocols, some messages are intended for the tag only (e.g. the commands) but we still use message flow arrows between the owner and the buyer/the delegate to indicate that such messages can always be overheard by the participating parties. We give details of our scheme below.

4.1. Setup

Before anyone can apply our scheme to aid RFID ownership transfer, users (including the manufacturer) are required to obtain their own public key PK and private key SK of the PKC. The manufacturer chooses a security parameter 1^k and runs $\texttt{SetupReader}(1^k)$ to setup the reader and prepares the authentication protocol $\texttt{Auth}()$ and the hash function $\mathcal{H}()$. The output bits of $\mathcal{H}()$ is set to $k - bits$. The manufacturer then chooses an unique ID for each tag and runs $\texttt{SetupTag}(ID)$, which outputs a $k - bits$ random number K_{ID} as the initial tag key. For each of the tag entries, the reader records and maintains the following values:

- ID : The ID of the tag.
- $Info_{ID}$: The information about the tag.
- K_{ID} : The current tag key.
- $K_{H_0} = K_{ID}$: The tag session key used in generation of $\mathcal{O}$.
- $\sigma_0 \leftarrow \texttt{Sig}(V_{S_0}, SK_M)$: The signature of the manufacturer (first owner) for a tag signed using its private key SK_M. $V_{S_0} \leftarrow \mathcal{H}(ID||Info_{ID})$.

Each tag is then assigned the following values:

- K_{ID} : The symmetric key of the tag shared with its current owner.
- $V_{S_0} \leftarrow \mathcal{H}(ID||Info_{ID})$: The hash (chain) value of the tag ID and its information used in signature generation.
- $\mathcal{O} \leftarrow \mathcal{H}(\sigma_0||K_{H_0})$: The hash value of the current owner's signature.

4.2. Key Change Protocol

First of all, we present our key change protocol. There are two main instances where this protocol should be executed, one before and one after the ownership transfer protocol. Changing the current tag key before the ownership transfer protocol can eliminate all the linkage of the previous communications between the current owner and the tag when the current tag key was used. This effectively provides *previous owner privacy*. Later when the ownership transfer protocol is completed, the new owner must change the tag key again such that the current tag key obtained from the previous owner can be overwritten with a fresh new key unknown to him/her. Since there is no secret shared between the tag and the new owner yet, it is unavoidable to preform such key change in a private environment free from the interception of the previous owner. This private key change effectively provides *new owner privacy*. The protocol is presented in figure 1. (notice that assumption 3.3.4 applies here). We will violate the notation a bit from now on and use bold letters to indicate the type of command being sent in the protocol. Here we have **KC** to indicate the command "Key Change".

4.3. Controlled Delegation Protocol

Next, we present our controlled delegation protocol. Using a similar idea in [2], a counter c is kept in the tag memory if the tag received a delegation command. Each time the tag is queried the value will increase by 1. Once c reaches c_{max}, the delegation is automatically expired and the delegated key will be replaced with the original tag key that was backed up at the start of the delegation. There is also a delegation cancel protocol, which

OWNER		TAG
$\{K_{ID}, K_{H_i}, \sigma_i\}$		$\{K_{ID}, \mathcal{O}\}$
$r \xleftarrow{R} \{0,1\}^k$, $T \leftarrow \mathcal{H}(r\|\|K_{ID}\|\|\mathbf{KC})$, $\mathcal{O} \leftarrow \mathcal{H}(\sigma_i\|\|K_{H_i}), u = \mathcal{O} \oplus T$, $K_{ID} = \mathcal{H}(r \oplus K_{ID})$	$\xrightarrow{\mathbf{KC},r,u}$	If $u \oplus \mathcal{H}(r\|\|K_{ID}\|\|\mathbf{KC}) \neq \mathcal{O}$, then $Fail$; Otherwise $K_{ID} = \mathcal{H}(r \oplus K_{ID})$

Figure 1. Key change protocol

Controlled delegation:		
OWNER	TAG	DELEGATE
$\{K_{ID}, K_{H_i}, \sigma_i, PK_D\}$	$\{K_{ID}, \mathcal{O}\}$	$\{SK_D\}$
$r \xleftarrow{R} \{0,1\}^k$, pick max, $T \leftarrow \mathcal{H}(r\|\|K_{ID}\|\|max\|\|\mathbf{CD})$, $\mathcal{O} \leftarrow \mathcal{H}(\sigma_i\|\|K_{H_i}), u = \mathcal{O} \oplus T$, $K_D \leftarrow \mathcal{H}(r \oplus K_{ID})$, $e \leftarrow \mathtt{Enc}(K_D, PK_D)$	$\xrightarrow{\mathbf{CD},max,r,u,e}$ $m = r\|\|K_{ID}\|\|max\|\|\mathbf{CD}$, if $u \oplus \mathcal{H}(m) \neq \mathcal{O}$, then $Fail$; Otherwise $K_B = K_{ID}$, $c_{max} = max, c = 0$, $K_{ID} = \mathcal{H}(r \oplus K_{ID})$	$K_D \leftarrow \mathtt{Dec}(e, SK_D)$, $c_{max} = max, c = 0$
Subsequent tag queries:		
DELEGATE		TAG
$\{K_D, c, c_{max}\}$		$\{K_{ID}, c, c_{max}, K_B\}$
If $c = c_{max}$, then $Fail$; Otherwise $c = c + 1$, executes $\mathtt{Auth}(K_D)$	$\xrightarrow{Query}$ $\xleftarrow{Response}$	Executes $\mathtt{Auth}(K_{ID})$, if $c < c_{max}$, then $c = c + 1$, if $c = c_{max}$, then $K_{ID} = K_B, K_B = 0^k$
Delegation cancel:		
OWNER		TAG
$\{K_{ID}, K_{H_i}, \sigma_i\}$		$\{K_{ID}, \mathcal{O}, c, c_{max}, K_B\}$
$r \xleftarrow{R} \{0,1\}^k$, $T \leftarrow \mathcal{H}(r\|\|K_{ID}\|\|\mathbf{DC})$, $\mathcal{O} \leftarrow \mathcal{H}(\sigma_i\|\|K_{H_i}), u = \mathcal{O} \oplus T$	$\xrightarrow{\mathbf{DC},r,u}$	If $u \oplus \mathcal{H}(r\|\|K_B\|\|\mathbf{DC}) \neq \mathcal{O}$, then $Fail$; Otherwise $K_{ID} = K_B, K_B = 0^k$

Figure 2. Controlled delegation protocol

invalidates the delegated key despite the current value of c and restores the backed up key as the current tag key. This effectively provides *controlled delegation*. To complete the protocol, the current owner has to send the delegated key to the delegate via a secure channel. In our setting, the public key of the delegate can be used to encrypt the key in a secure manner thanks to the PKC. This protocol is also used to provide *authorization recovery* as the previous owner can be viewed as a delegate. Comparing to [11] and [1] where the ownership will be taken by the previous owner once authorization recovery is executed, our scheme allows the current owner to regain the ownership by executing the delegation cancel protocol without going through another ownership transfer instance with the previous owner. The protocol is presented in figure 2. Notice that when the delegated key is replaced by the backed up key at the end of delegation, the backed up key is zeroed out with k 0-bits to clear any possible trace of old tag key (in case the tag is compromised).

4.4. Ownership Transfer Protocol

Following the assumption in 3.3.3, an intended RFID item has already been authenticated using $\texttt{Auth}()$ and singulated from other RFID items. Its ID and $Info_{ID}$ are obtained and its corresponding tag key K_{ID} is selected. Before the protocol begins, the owner will forward the ID and $Info_{ID}$ to the buyer (notice that the buyer can only verify the validity of ID and $Info_{ID}$ until phase 5.). They also exchange their public keys PK_O and PK_B, allowing the other party to verify the validity of the public key with the PKC before actually starting the ownership transfer protocol. Our ownership transfer protocol contains several phases. One nice feature of this is that users can cancel the ownership transfer at any phase without sabotaging the security of the whole system. The first three phases are in fact the key change protocol, controlled delegation protocol and an execution of $\texttt{Auth}()$. At the end of the protocol, the new owner should execute the key change protocol in a private environment. We aware that a compact design of our ownership transfer protocol is possible (e.g. using the same random number) but we consider the current form a clearer version to present our idea and we leave the compact version in the full paper. The protocol is presented in figure 3.

5. Security Analysis

5.1. Previous owner privacy and new owner privacy

We have already mentioned about the security properties *previous owner privacy* and *new owner privacy*, which are achieved by the key change protocol described in section 4.2. By running the key change protocol before and (secretly) at the end of the ownership transfer protocol, any trace of the previous tag key is eliminated thanks to the one-wayness property of the hash function $\mathcal{H}()$. We prove this by contradiction: suppose there is an attacker who can output the previous tag key $K_{ID_{i-1}}$ given the current tag key K_{ID_i} as input (i.e. it is a forward security attacker who compromises the memory of the tag to extract the current tag key), one can use this attacker to find the pre-image of K_{ID_i} in $\mathcal{H}()$ by computing $r \oplus K_{ID_{i-1}}$, where r was the random number used in the last instance of the key change protocol sent in plaintext. This contradicts the assumption that finding the pre-image of a hash value is hard under the one-wayness property. Hence either the output of the previous tag key $K_{ID_{i-1}}$ is only a blind guess (which only has negligible probability 2^{-k} to be a correct guess) or the attacker knows the previous tag key from other source. There are two cases for the attacker to obtain the previous tag key: i.) by compromising the tag before the key exchange protocol was carried out. However, this violates the assumption 3.3.2 that once a tag is compromised, it is not functioning anymore and would not have completed the key exchange protocol. ii.) the attacker is the previous owner who always know the previous tag key. Since the previous owner will not attack his own privacy, he will only target on attacking the new owner privacy. Hence the only fix to this is to require the new owner to carry out the key change protocol in a private environment away from the interception of the previous owner, such that the random number r becomes a secret added into the computation of the new tag key. Guessing r would take the same effort as guessing K_{ID_i} as they are both k-bits.

OWNER $\{PK_O, SK_O, PK_B,$ $K_{ID}, K_{H_i}, \sigma_i\}$	TAG $\{K_{ID}, V_{S_i}, \mathcal{O}\}$	BUYER $\{PK_B, SK_B, PK_O,$ $ID, Info_{ID}\}$
Phase 1. Key change		
$r \xleftarrow{R} \{0,1\}^k$, $T \leftarrow \mathcal{H}(r\|\|K_{ID}\|\|\mathbf{KC})$, $\mathcal{O} \leftarrow \mathcal{H}(\sigma_i\|\|K_{H_i}), u = \mathcal{O} \oplus T$, $K_{ID} = \mathcal{H}(r \oplus K_{ID})$	$\xrightarrow{\mathbf{KC}, r, u}$ $m = r\|\|K_{ID}\|\|\mathbf{KC}$, If $u \oplus \mathcal{H}(m) \neq \mathcal{O}$, then $Quit$; Otherwise $K_{ID} = \mathcal{H}(r \oplus K_{ID})$	
Phase 2. Delegation		
$r \xleftarrow{R} \{0,1\}^k, max = 1$, $T \leftarrow \mathcal{H}(r\|\|K_{ID}\|\|1\|\|\mathbf{CD})$, $\mathcal{O} \leftarrow \mathcal{H}(\sigma_i\|\|K_{H_i}), u = \mathcal{O} \oplus T$, $K_{H_{i+1}} \leftarrow \mathcal{H}(r \oplus K_{ID})$, $e \leftarrow \mathtt{Enc}(K_{H_{i+1}}, PK_B)$	$\xrightarrow{\mathbf{CD}, 1, r, u, e}$ $m = r\|\|K_{ID}\|\|1\|\|\mathbf{CD}$, if $u \oplus \mathcal{H}(m) \neq \mathcal{O}$, then $Quit$; Otherwise $K_B = K_{ID}$, $c_{max} = 1, c = 0$, $K_{ID} = \mathcal{H}(r \oplus K_{ID})$	$K_{H_{i+1}} \leftarrow \mathtt{Dec}(e, SK_B)$
Phase 3. Authentication		
	$\xleftarrow{Query}$ $\xrightarrow{Response}$ Executes $\mathtt{Auth}(K_{ID})$, $K_{ID} = K_B$	Executes $\mathtt{Auth}(K_{H_{i+1}})$, if returns $\perp$, then $Quit$; Otherwise proceed
Phase 4. Ownership transfer starts		
$r \xleftarrow{R} \{0,1\}^k$, $T \leftarrow \mathcal{H}(r\|\|K_{ID}\|\|\mathbf{TS})$, $\mathcal{O} \leftarrow \mathcal{H}(\sigma_i\|\|K_{H_i}), u = \mathcal{O} \oplus T$	$\xrightarrow{\mathbf{TS}, r, u}$ $m = r\|\|K_{ID}\|\|\mathbf{TS}$, if $u \oplus \mathcal{H}(m) \neq \mathcal{O}$, then $Quit$; Otherwise proceed	
Phase 5. Tag assurance		
$V_{S_{i+1}} \leftarrow \mathcal{H}(V_{S_i})$	$\xleftrightarrow{\mathbf{TA}, V_{S_i}}$	Let $V_{S_0} = \mathcal{H}(ID\|\|Info_{ID})$, for $j = 0 \ldots n$, $V_{S_{j+1}} \leftarrow \mathcal{H}(V_{S_j})$, until $V_{S_j} = V_{S_i}$ If not found, then $Quit$; Otherwise proceed
Phase 6. Buyer signature verification		
If $\mathtt{Ver}(V_{S_{i+1}}, \sigma_{i+1}, PK_B)$ returns $\perp$, then $Quit$; Otherwise proceed	$\xleftarrow{\mathbf{VR}, \sigma_{i+1}}$ Stores σ_{i+1}	$\sigma_{i+1} \leftarrow \mathtt{Sig}(V_{S_{i+1}}, SK_B)$
Phase 7. Ownership transfer ends		
$K = K_{ID} \oplus K_{H_{i+1}}$, $r \xleftarrow{R} \{0,1\}^k$, $T \leftarrow \mathcal{H}(r\|\|K_{ID}\|\|\mathbf{TE})$, $\mathcal{O} \leftarrow \mathcal{H}(\sigma_i\|\|K_{H_i}), u = \mathcal{O} \oplus T$	$\xrightarrow{\mathbf{TE}, r, u, \sigma_i, K}$ $m = r\|\|K_{ID}\|\|\mathbf{TE}$, if $u \oplus \mathcal{H}(m) \neq \mathcal{O}$, then $Fail$; Otherwise $V_{S_i} = \mathcal{H}(V_{S_i})$, $\mathcal{O} = \mathcal{H}(\sigma_{i+1}\|\|K_{H_{i+1}})$	If $\mathtt{Ver}(V_{S_i}, \sigma_i, PK_O)$ returns $\perp$, then $Fail$; Otherwise $K_{ID} = K \oplus K_{H_{i+1}}$, records $\sigma_{i+1}, K_{H_{i+1}}$

Figure 3. Ownership transfer protocol

5.2. Controlled delegation and authorization recovery

Both of these security properties are provided by our controlled delegation protocol. Since these two properties are more like security features rather than security protections, it is trivial enough to verify their correctness from the protocol description. The only

thing to keep in mind is that the delegated key computation is the same as the new tag key computation (i.e. $\mathcal{H}(r \oplus K_{ID})$), one should not reuse the same random number r for the key change protocol after the controlled delegation protocol. Otherwise the delegate can instantly obtain the new tag key, which was in fact the delegated key he received before. Also, notice that as long as the current tag key does not change, the delegation message $\mathbf{CD}, max, r, u, e$ and the delegation cancel message $\mathbf{DC}, r, u$ can be replayed. e.g. the delegate may want to gain additional access to the tag after the first controlled delegation has expired. Hence one may want to execute the key change protocol to renew the tag key after a delegation has expired.

5.3. Tag assurance

Tag assurance is guaranteed in phase 3. and 5. of the ownership transfer protocol. In most of the previous ownership transfer schemes, the buyer can only choose to believe the RFID tagged item presented by the current owner is the item he/she wants and not something else (consider a cheating owner who swapped the trading item with something else that looks similar to the original item but at a lower quality). In phase 3. of our protocol, it allows the buyer to make sure the owner actually knows the tag key of the trading item. This avoids someone trying to sell stolen goods. Next in phase 5. by verifying the hash chain value V_{S_i} generated from $ID, Info_{ID}$ gives the buyer confidence on the true identity of the tag (under the assumption 3.3.2). Together they guarantee to the buyer that the owner owns the item and the information $ID, Info_{ID}$ supplied by the owner is the correct description of the item. Thanks to the collision resistance property of $\mathcal{H}()$, it is hard for the owner to find another message/pre-image $ID', Info'_{ID'}$ (to replace the description of the swapped lower quality item with some exaggerated information) such that it gives the same hash chain value V_{S_i} after hashing it several times with $\mathcal{H}()$ provided that n (the maximum acceptable number of previous owners/number of hash chains) is reasonably small. Again, we prove this by contradiction: suppose there is an attacker who can output a fake description $ID', Info'_{ID'}$ of the trading item by inputting a hash chain value V_{S_i}, where $i <= n$. One can use this attacker to find a collision in $\mathcal{H}()$. Let $ID, Info_{ID}$ be the original message and $V_{S_j} = V_{S_i}$ is the hash chain value of it under $\mathcal{H}()$ where $j <= n$, then the collision is $V_{S_{j-1}}$ and $V_{S_{i-1}}$. This contradicts the assumption that finding a collision in a hash function is hard under the collision resistance property. Hence either $ID', Info'_{ID'}$ is in fact the correct description of the item (i.e. $ID, Info_{ID}$) or V_{S_i} must be fake as well. There can be two cases: i.) the attacker has overwritten the hash chain value stored in the tag with V_{S_i}. However, this violates the assumption 3.3.2 that the tag has incorruptible memory. ii.) the whole tag is a fake tag created by the attacker by running `SetupTag`(ID'). Whether a fake tag can be spotted or not is beyond the scope of this paper.

5.4. Current ownership proof and Undeniable ownership transfer

Tag ownership cannot be defined simply as the one who holds the tag or someone who knows the tag key, especially when delegation is implemented. Our scheme requires a tag to store the value $\mathcal{O}$, which is the hash value of the owner's signature. Hence ownership in our scheme is defined as someone who knows both the pre-image of this hash value and the current tag key. Since the pre-image is a signature, it can be tightly bound to the

owner as he/she is the only one who can generate such signature. To bind the owner to the tag, the message signed is the hash chain value V_{S_i}. To prove previous ownership and current ownership, it is suffices (together with the assumption 3.3.2) to present V_{S_i}, σ_i and $V_{S_{i+1}}, \sigma_{i+1}$ to any third party. One cannot deny ever created the signatures and hence they become the evidence of ownership transfer and the proof of current ownership.

5.5. Owner initiation

Since both the current owner and the delegate may hold the current tag key, the owner must possess some additional secret to distinguish the owner's role from the delegate's role, so that only the owner can issue commands to the tag but not the delegate. We use the hash value of the owner's signature $\mathcal{O}$ as the additional key to initiate tag commands. Notice that in each of the commands of the three protocols in our scheme, the owner is required to compute $\mathcal{O} \leftarrow \mathcal{H}(\sigma_i||K_{H_i})$ and $T \leftarrow \mathcal{H}(r||K_{ID}||\mathbf{COMMAND})$. $\mathcal{O}$ remains the same throughout the ownership of the same owner, while T changes every time when the current tag key changes and its freshness is guaranteed by the random number r. As σ_i is sent in plaintext in the ownership transfer protocol, the secrecy of $\mathcal{O}$ is protected by K_{H_i}, the delegated key sent via a secure channel to the current owner when ownership is transferred. Hence to break owner initiation, one must obtain K_{H_i}, which is impossible because it is sent in a secure channel, or $\mathcal{O}$. To obtain $\mathcal{O}$, one may guess on the value of K_{H_i}, which has negligible success probability. One may also compromise the tag as the tag stores $\mathcal{O}$. But once the tag is compromised, it will be virtually dead and rendered the acquisition of $\mathcal{O}$ useless. Otherwise one may try to compute $\mathcal{O}$ from $\mathcal{O} = u \oplus T$. But to compute T, the knowledge of the current tag key K_{ID} is required. At the end, we have guaranteed owner initiation.

6. Conclusions

We proposed a new RFID ownership transfer scheme in this paper. Our scheme consists of three protocols: key change protocol, delegation protocol and ownership transfer protocol. Our scheme combines these three protocols to provide a secure method for users to transfer their RFID tags to new hands. We also considered some practical needs users may request in ownership transfer. For example, we have tag assurance to deal with cheating sellers. Current ownership proof creates a tight binding between the current owner and the tag. Undeniable ownership transfer is aimed to handle dispute that may occur when the previous owner denies selling a faulty item to the current owner. Owner initiation guarantees only the current owner can give various commands to the tag. We believe this will open up new research directions in this area and allow more new ideas to come and strengthen the development of RFID applications.

References

[1] T. Dimitriou. rfidDOT: RFID Delegation and Ownership Transfer made simple. In *4th International Conference on Security and Privacy in Communication Netowrks - SecureComm*, pages 1–8, Istanbul, Turkey, September 2008. ACM Press.

[2] S. Fouladgar and H. Afifi. A Simple Delegation Scheme for RFID Systems (SiDeS). In *IEEE RFID*, Grapevine, TX, USA, March 2007. IEEE.

[3] S. Fouladgar and H. Afifi. A Simple Privacy Protecting Scheme Enabling Delegation and Ownership Transfer for RFID Tags. *Journal of Communications*, 2(6):6–13, November 2007.

[4] S. Fouladgar and H. Afifi. An Efficient Delegation and Transfer of Ownership Protocol for RFID tags. In *1st International EURASIP Workshop on RFID Technology*, Vienna, Austria, September 2007. EURASIP.

[5] K. H. S. S. Koralalage, M. R. Selim, J. Miura, Y. Goto, and J. Cheng. POP Method: An Approach to Enhance the Security and Privacy of RFID Systems Used in Product Lifecycle with an Anonymous Ownership Transferring Mechanism. In *ACM Symposium on Applied Computing - SAC*, pages 270–275, Seoul, Korea, March 2007. ACM Press.

[6] C. H. Lim and T. Kwon. Strong and Robust RFID Authentication Enabling Perfect Ownership Transfer. In P. Ning, S. Qing, and N. Li, editors, *8th International Conference on Information and Communications Security - ICICS*, volume 4307 of *Lecture Notes in Computer Science*, pages 1–20, Raleigh, NC, USA, December 2006. Springer-Verlag.

[7] D. Molnar, A. Soppera, and D. Wagner. A Scalable, Delegatable Pseudonym Protocol Enabling Ownership Transfer of RFID Tags. In B. Preneel and S. Tavares, editors, *12th International Workshop on Selected Areas in Cryptography - SAC*, volume 3897 of *Lecture Notes in Computer Science*, pages 276–290, Kingston, ON, Canada, August 2005. Springer-Verlag.

[8] D. Molnar, A. Soppera, and D. Wagner. A Scalable, Delegatable, Pseudonym Protocol Enabling Ownership Transfer of RFID Tags. In *Workshop on RFID and Light-Weight Crypto*, Graz, Austria, July 2005. IAIK TU Graz.

[9] K. Osaka, T. Takagi, K. Yamazaki, and O. Takahashi. An Efficient and Secure RFID Security Method with Ownership Transfer. In Y. Wang, Y. ming Cheung, and H. Liu, editors, *Computational Intelligence and Security - CIS*, volume 4456 of *Lecture Notes in Computer Science*, pages 778–787, Guangzhou, China, November 2006. Springer-Verlag.

[10] J. Saito, K. Imamoto, and K. Sakurai. Reassignment Scheme of an RFID Tag's Key for Owner Transfer. In T. Enokido, L. Yan, B. Xiao, D. Kim, Y.-S. Dai, and L. T. Yang, editors, *Embedded and Ubiquitous Computing Workshops - EUC*, volume 3823 of *Lecture Notes in Computer Science*, pages 1303–1312, Nagasaki, Japan, December 2005. Springer-Verlag.

[11] B. Song. RFID Tag Ownership Transfer. In *4th Workshop on RFID Security - RFIDSec*, Budapest, Hungary, July 2008. IAIK TU Graz.

[12] A. Soppera and T. Burbridge. Secure by Default: The RFID Acceptor Tag (RAT). In *2nd Workshop on RFID Security - RFIDSec*, Graz, Austria, July 2006. IAIK TU Graz.

[13] T. van Deursen, S. Mauw, S. Radomirović, and P. Vullers. Secure Ownership and Ownership Transfer in RFID Systems. In *14th European Symposium on Research in Computer Security - ESORICS*, volume 5789 of *Lecture Notes in Computer Science*, pages 637–654, Saint-Malo, France, September 2009. Springer-Verlag.

[14] S. Vaudenay. On Privacy Models for RFID. In K. Kurosawa, editor, *13th International Conference on the Theory and Application of Cryptology and Information Security - ASIACRYPT*, volume 4833 of *Lecture Notes in Computer Science*, pages 68–87, Kuching, Malaysia, December 2007. Springer-Verlag.

This page intentionally left blank

Radio Frequency Identification System Security
Y. Li and J. Zhou (Eds.)
IOS Press, 2010

doi:10.3233/978-1-60750-485-6-49

An Efficient Ultralightweight Authentication Protocol for RFID Systems

Kuo-Hui Yeh, N.W. Lo[1] and Enrico Winata

Department of Information Management,
National Taiwan University of Science and Technology, Taiwan R.O.C.

Abstract. Since Peris-Lopez et al. proposed the design of ultralightweight authentication schemes [35-37] for low-cost RFID tags in 2006, research community has demonstrated a significant advancement on this interesting research area in recent years. However, previously published studies are subject to either various security vulnerabilities or inefficient management on tag memory. Motivated by the nature of resource limitation in a tag, we develop a process-oriented ultralightweight RFID authentication protocol which delivers strong security intensity, robust privacy protection as well as less tag memory space required. In addition, a randomness evaluation on the output values of our scheme is performed to ensure the proposed authentication protocol produces qualified output randomness. Our security analysis and performance comparison show that our process-oriented authentication scheme outperforms relevant works by supporting essential system security criteria with less computation effort and better tag memory utilization.

Keywords. Authentication, Privacy, RFID, Security, Ultralightweight

Introduction

As a tiny-sized, cheap electronic information container with wireless accessibility, Radio Frequency Identification (RFID) technology has gotten increasing attention during the past few years and is envisioned as the next generation technology for object identification and management. Well-known RFID applications such as livestock tracking, children/seniors location monitoring, and supply chain management have been developed and deployed to reduce management cost and obtain important business conditions in time. However, the widespread deployment of RFID based applications may expose potential security vulnerabilities and privacy threats to both corporations and individuals. For example, corporate spies may perform espionage activities to gather valuable information of tag carriers (objects) from unprotected RFID tags by querying these tagged objects with illegal RFID readers. A man can easily be traced where he went as long as an identified RFID-tagged object is carried by him. Similarly, the monetary values of RFID-tagged items a person worn or carried with him can be determined by an adversary effortlessly. It is possible for adversaries to develop association rules on a set of RFID-tagged items to gain transaction information on identified items, and to track people without knowing their identities. Even more sophisticated security threats such as breadcrumb threat or tag cloning [16] can potentially emerge from cur-

[1] Corresponding author.

rent RFID development. Based on these observations, RFID authentication scheme is definitely necessary for RFID applications to defend against these potential security and privacy threats.

The distinguishable characteristics between RFID authentication protocol and general-purpose authentication scheme are elaborated as follows. First of all, the nature of restricted computation ability and limited memory space of low-cost RFID tags make existing RFID-based systems vulnerable to many security attacks and potential privacy threats. Secondly, contact-less identification technique on RFID and new type of RFID-based application also pave the way for new privacy threats under the insecure wireless RF environment. The third, Le et al. [24] showed that availability, forward security, and concurrent security should all be specially emphasized in RFID-based systems. From system availability aspect, an RFID device may turn itself into non-available state such that it cannot be successfully authenticated by readers when encountering malicious attacks or abnormal responses. The most famous attack of this type is called de-synchronization attack. Regarding to forward security, once a RFID device was discarded or compromised, forward security becomes an essential requirement to guarantee the privacy of past transactions with this RFID device. Furthermore, it is important to take concurrent operation environment (involving RFID tags and other system entities) into design consideration when designing a secure RFID-based system. According to the reasons stated previously, we argue that traditional authentication schemes cannot plug-in and play without further evolution in current RFID-based systems. In brief, providing a secure authentication scheme with enhanced privacy protection and robust data security will be crucial to future RFID systems. In order to fulfill security and privacy requirements on low-cost RFID tags and their corresponding systems, how to use simple bitwise or arithmetic operations such as XOR, AND, OR and addition to design a secure authentication scheme, has become one of the hottest research topics throughout the past few years of RFID security community. This kind of mechanism is classified as the ultralightweight based authentication scheme [11]. However, previously proposed schemes [11, 25, 35-38] are either subject to various security flaws [5, 10, 13, 20, 27-28, 39-40, 44, 46] or inefficient management on tag memory. In this paper, we propose a novel ultralightweight RFID authentication protocol to achieve requirements on security criteria, privacy protection and system performance. Our scheme adopts a process-oriented design to gain better utilization on memory space and less computation workload at both tag end and the backend server (database).

Related Work

In this section, we briefly categorize previous research works into several clusters based on the main security methodology and specific requirement from application domain.

The vast literature devoted to hash-based RFID authentication field has been reviewed on several occasions [1-2, 7, 12, 19, 23, 26, 29, 32-34, 42, 45]. In these authentication protocols, tag should support the random number generator operation and one-way hash function to provide mutual authentication and strong data integrity. However, the cost of one-way hash function is too high to be afforded in a low-cost passive tag (usually in the range of five to ten cents); and most of these hash-based authentication schemes have weaknesses in terms of the aspects of data security or individual privacy. In 2003, Weis et al. [45] proposed two RFID authentication schemes, called hash-based

access control and randomized access control. However, both schemes are not able to defend against the tag tracking attack and replay attack. Later, Ohkubo et al.'s scheme [32] successfully provides the indistinguishability and forward security. Unfortunately, their scheme cannot resist to the replay attack. Next, Yang et al. [42] pointed out the mechanism in [19] cannot satisfy the claimed security, i.e. the resistance to tag location tracking, and accordingly proposed a security enhanced version. Nevertheless, Avoine et al. [2] had reported that Yang et al.'s scheme cannot guarantee tag carrier privacy protection. Similarly, the protocols in [1, 12, 23] cannot provide the anonymity property and resistance to replay attack. Recently, Lo and Yeh [29] had demonstrated that the anonymity and forward security cannot be guaranteed in the scheme developed by Park and Lee [34]; and the same weaknesses emerge in the studies [7] and [33] also. Later, Lee et al. [26] proposed an authentication mechanism which provides stronger security than previous studies. However, the replay attack is unsolved in the backend server.

The study of HB-series authentication protocol [3, 17, 21, 31, 41] has become an important and interesting topic due to low cost demand (lightweight computation) on tag-side. However, Chien [11] pointed that such kind of protocol does not take the authentication of reader side into consideration. This will cause the privacy issue during the tag identification. On the other hand, a series of authentication protocols [4, 6, 9, 14-15, 18, 22, 30, 46] conforming to EPC Class-1 Generation-2 (EPC Gen-2) standards [15] had been developed. Among them, the authentication scheme proposed by Karthikeyan and Nesterenko [22] cannot provide anonymity property to RFID tag and resist security threats such as de-synchronization attacks and replay attack. Furthermore, Chien and Chen [9] pointed out Duc et al.'s scheme [14] cannot prevent against de-synchronization attacks, counterfeit tag attack and forward secrecy revealing, and developed a novel authentication process to provide stronger privacy and security properties. However, their scheme cannot resist the replay attack, forward secrecy revealing and tag-tracking attack [18, 30]. Recently, Burmester and Mederiros introduce a more robust EPC Gen-2 conformed protocol, called TRAP-3, to pursue stronger anonymity property and security feature. Nevertheless, TRAP-3 still suffers from the de-synchronization attacks [46].

In 2006, Peris-Lopez et al. [35-37] published a series of ultralightweight RFID authentication protocol in which only very lightweight operations such as XOR, AND, OR and addition operation are required at tag side. However, the studies in [10, 27-28] had shown the de-synchronization attacks and disclosure attacks cannot be solved in Peris-Lopez et al.'s authentication protocols. Later, Chien [11] proposed an ultralightweight authentication protocol, called SASI, to provide strong mutual authentication and robust data integrity under the insecure communication environment. Unfortunately, several security flaws, such as de-synchronization attacks and disclosure attack, have been introduced by [5, 13, 20, 40, 44] Later, Peris-Lopez et al. [38] developed a Gossamer protocol to pursue better security intensity. However, Gossamer protocol is vulnerable to de-synchronization attacks [46].

The Proposed Authentication Scheme

In this section, we present a process-oriented (*flag*=0 or *flag*=1) ultralightweight RFID authentication protocol. This design enhances the protocol efficiency in terms of memory space economizing and computation cost reducing. Our protocol is designed to accommodate with very low-cost RFID tag which only involves simple bitwise op-

erations such as *XOR*, *AND*, *OR*, addition mod 2^m and circular shift rotation $Rot(x, y)$. Note that m is the bit-length of each tag's identity. Similar to the SASI and Gossamer schemes, we assume that tag is vulnerable to all possible passive attacks and the communication channel between the reader and the backend database is secure.

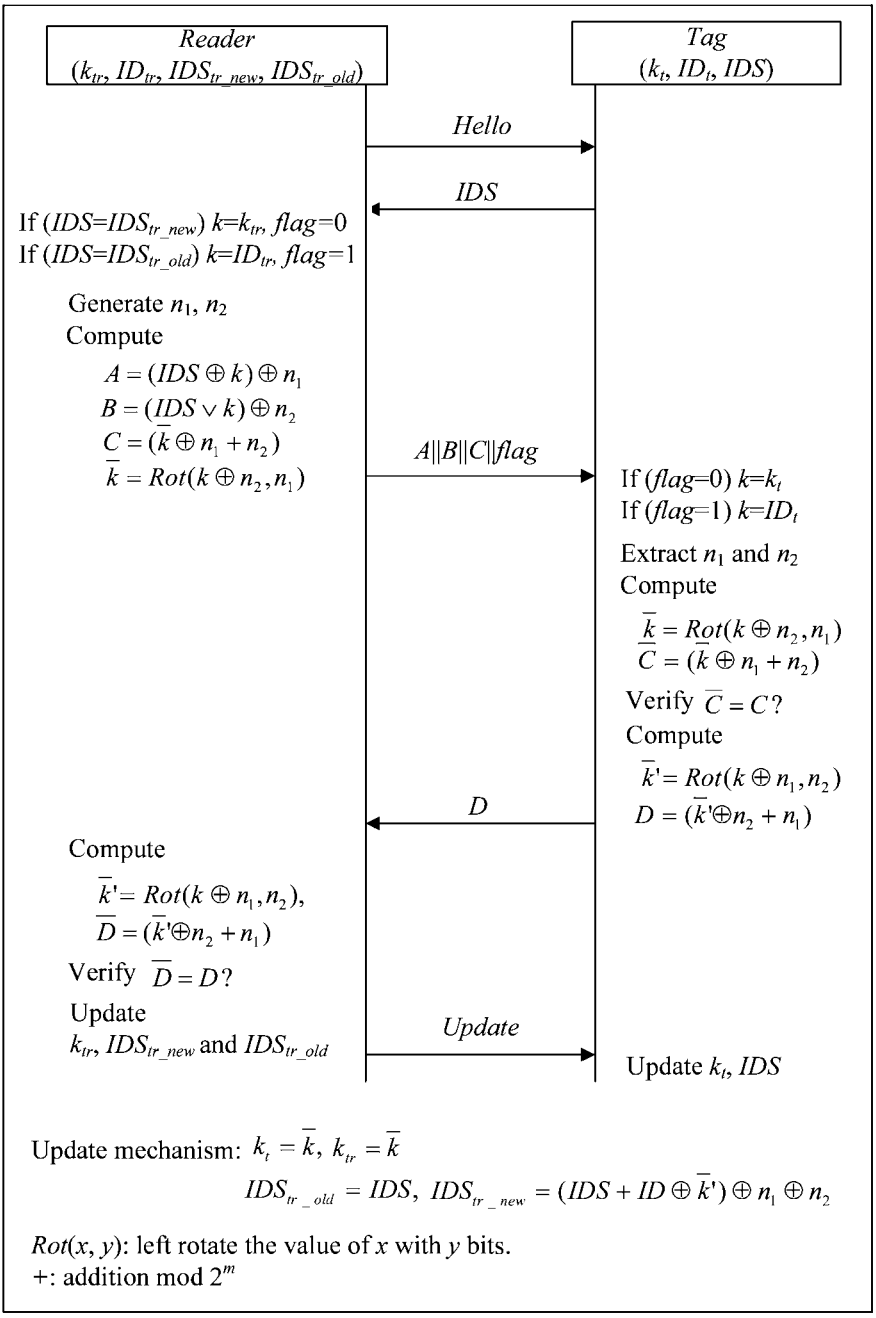


Figure 1. The proposed authentication scheme.

For each tag, an initial setup is performed to store three values in tag's memory. First, an authentication key k_t, and an index-pseudonym *IDS* are initially generated at the backend database before inserting into each tag. Next, each tag stores a pre-defined identity *ID* which is assigned to both the backend database and the corresponding tag during system initialization. From the viewpoint of forward security, the authentication key will be updated at both the backend database and the corresponding tag after each successful authentication session.

For each tag, the backend database maintains a record of four fields including ID_{tr}, an authentication key k_{tr} and two index-pseudonyms IDS_{tr_new} and IDS_{tr_old} associated with this ID_{tr}. Note that the dual-record design of the index-pseudonym is utilized to resist to the de-synchronization attacks. For convenience, the IDS_{tr_new} and IDS_{tr_old} are set to same value initially. Once the reader queries the tag, the normal authentication process is activated as shown in Fig. 1. We describe the detailed operation procedures in the following. Note that our protocol considers two different situations based on previous authentication session is safely terminated (*flag*=0) or not (*flag*=1). This design is utilized to solve the situation of tag may not be authenticated with backend database as the shared secret information between them is out-of-synchronized.

Reader → Tag*: *Hello

Tag → Reader*: *IDS

The reader sends a "*Hello*" message as a request to inquire the tag. Upon receiving this "*Hello*" message, the tag responds its index-pseudonym *IDS* back to the reader.

Reader → Tag*: *A||B||C||flag

After receiving the *IDS*, the reader utilizes the responded *IDS* value as an index to search the corresponding tuple from the backend database. If the match entry is not found, the reader drops the incoming message and stop. Once finding the match entry from the backend database, the reader sets the corresponding shared secret value *k* depending on the received value *IDS*.

If ($IDS=IDS_{tr\_new}$) then $k=k_{tr}$ and *flag*=0
If ($IDS=IDS_{tr\_old}$) then $k=ID_{tr}$ and *flag*=1

The reader generates two random numbers n_1 and n_2, and performs the following equations. Then, the reader transmits messages *A*, *B*, *C* with value *flag* back to the tag.

$$A = (IDS \oplus k) \oplus n_1,\ B = (IDS \vee k) \oplus n_2$$
$$\overline{k} = Rot(k \oplus n_1, n_2),\ C = (\overline{k} \oplus n_1 + n_2)$$

Tag → Reader*: *D

Upon receiving the response message (*A||B||C||flag*), the tag sets the value *k* depending on the value *flag* and utilizes this value *k* to extract the n_1 and n_2 from *A* and *B*. Next, the tag calculates the values $\overline{k}$ and $\overline{C}$ to verify whether the calculated $\overline{C}$ and received *C* are the same or not. This verification process can ensure the data integrity of n_1 and n_2.

If (*flag*=0) $k=k_t$ or If (*flag*=1) $k=ID_t$
Compute $\overline{k} = Rot(k \oplus n_2, n_1),\ \overline{C} = (\overline{k} \oplus n_1 + n_2)$, and verify $\overline{C} = C$?

If both of *C* and $\overline{C}$ are identical, the tag computes the values $\overline{k}'$ and *D* and sends *D* as a response to the reader.

$$\overline{k}' = Rot(k \oplus n_1, n_2),\ D = (\overline{k}' \oplus n_2 + n_1)$$

Once receiving D, the reader computes $\overline{k}'$ and $\overline{D}$ to verify the correctness of incoming message. If the computed value $\overline{D}$ is equal to the received value D, the reader updates the values k_{tr}, IDS_{tr_new} and IDS_{tr_old}. Otherwise, the reader drops the incoming message and stop. Finally, the reader sends an *Update command* back to the tag.

$$k_{tr} = \overline{k}, IDS_{tr\_old}=IDS,\ IDS_{tr\_new} = (IDS + ID \oplus \overline{k}') \oplus n_1 \oplus n_2$$

Reader → Tag*: *Update

Once the tag receives the *Update command*, it updates the values k_t and IDS.

$$k_t = \overline{k},\ IDS = (IDS + ID \oplus \overline{k}') \oplus n_1 \oplus n_2$$

Security Analysis

In this section, we analyze the security and performance requirements of our protocol and compare it with the Gossamer and SASI schemes.

- *Mutual Authentication and Data* Confidentiality

  In our protocol, only genuine tag and legitimate reader can authenticate each other via the shared secret key values k_t and k_{tr}. It is obvious that the mutual authentication can be provided with the verification of values A, B, C and D. In addition, each submessage, such as A, B, C and D at each session, is protected by at least two secret values, i.e. k, n_1 and n_2. The malicious attacker cannot derive any useful information from A, B, C and D. Hence, our protocol can guarantee the data confidentiality.

- *Tag Anonymity*

  Anonymity to tags can be provided in our protocol due to only enciphered and randomized messages A, B, C and D are broadcasted during the reader-tag mutual communication periods of time. Without knowing the shared secret values, i.e. k_t, k_{tr}, ID_t and ID_{tr}, the attacker cannot derive the identity of specific tag and trace it. Furthermore, the index-pseudonym IDS will be updated after each successfully authentication session. This design makes the IDS randomized at each new session. Note that even if the attacker probes the same tag more than two times between two normal authentication processes, the tag will respond the same IDS. However, once the tag updates its IDS, the tag becomes anonymous again and it is difficult for adversaries to keep tracking the specific tag in a group of observed tags. In addition, the IDS does not provide any useful information about the tag or tag-carrier. Hence, we argue that this light concern will not destroy any individual privacy property in our scheme.

- *Replay Attack*

  Because the generated random number n_1 and n_2 are different and independent at each session, even if the attacker replays the eavesdropped D value (obtained in previous session) to reader; the reader can examine the invalidity of this replayed message.

- *Desynchronization attacks*

  Our protocol utilizes the process-oriented design (*flag*=0 and *flag*=1) and dual-record design (IDS_{tr_new} and IDS_{tr_old}) to re-authenticate tag successfully once the shared secret key k_{tr} is out of synchronization at the previous session. Both of these two designs allow a tag with non-synchronized keys or index-pseudonym values can

still be authenticated by the legitimate reader and resynchronize its data with the reader. Note that instead of adopting two-record design on index-pseudonym at tag side, we store the old and new index-pseudonym values, i.e. IDS_{tr_new} and IDS_{tr_old}, at backend database. This mechanism can prevent de-synchronization attacks [46] as well as reduce more memory space consumption at tag side.

- *Forward Security*

In the worst case, if the tag was compromised and all data stored in it was known by the adversary, the attacker still cannot trace back the trajectory of the compromised tag as the keys are updated at each session. Since the shared key value in the tag will be automatically updated with two one-time-valid random numbers (n_1 and n_2) after each successful authentication session, the forward security feature is naturally embedded in our proposed scheme.

- *Memory Utilization*

For a low-cost (5-cents) passive tag, the number of gates available for security design cannot exceed 2.5 to 5K gates [32]. Take the resource limitation into consideration; the cost of tag implementation is very sensitive regarding the computation operation utilization and memory storage requirement. From the viewpoint of computation operation utilization, the tag in our protocol only requires a few number of simple bitwise operations such as *Rot*, +, *XOR*, *AND*, and *OR*. This makes our proposed scheme very efficient and practical. On the other hand, the memory space requirement is one of the most major considerations while designing a secure ultralighweight RFID authentication scheme. In our protocol, the memory storage requirement of the tag only needs one identity ID_t, one shared secret key k_t, and one index-pseudonym value *IDS*. Apparently, the memory space requirement in the tag side of our scheme (3*L* bits) is less than that (7*L* bits) in SASI and Gossamer protocols, where *L* denotes the bit length of index-pseudonym *IDS*, identity ID_t and shared key value k_t. In brief, the memory space consumption at tag side can be reduced about 57.14% and accordingly the cost of tag can be significantly decreased. According to the above excellences, we believe that our protocol is one of the most promising candidates to secure RFID systems with very low cost tags.

Randomness Evaluation

As the ultralightweight authentication scheme become the most promising technology for securing RFID systems, the security intensity of primitive operations implemented on low-cost tags are worried by individuals or organizations. Under insecure wireless communication environment, attackers may either utilize the brute force cryptanalysis to decipher the output messages, such as *IDS*, *A*, *B*, *C* and *D* in our protocol, or use the traffic analysis method to analyze the output messages and carry out some potential malicious attacks. Note that the brute force cryptanalysis attacks cannot be avoided as the cost of tag is very low; we do not accordingly consider this irresistible attack in our study. From the viewpoint of traffic analysis mechanism, the randomness of the output values in our scheme must be significantly tenable. Hence, we adopt the following experiment to evaluate the randomness of our protocol. The simulation program is written with C# under Visual Studio .NET environment. First, the *IDS*, *ID* and *k* are randomly initialized. Then, a 10^6 bits-length sequence, consists of the output messages (*IDS*, *A*, *B*, *C* and *D*) at each session of our protocol, are generated as the input value of well-

known randomness testing proposed by Rukhin et al. [43]. The purpose of Rukhin et al.'s randomness testing is to ensure the output bit sequence must be unpredictable or not easily identified as a set of distinguishable patterns for the cryptanalyst. For achieving this goal, Rukhin et al. recommended sixteen statistical testing to validate the arbitrarily long randomness of binary bit sequences. These tests concentrate on a variety of different types of non-randomness that could exist in a sequence; and it may be useful as a cryptanalysis step. Here, we briefly describe the purpose of each statistical testing.

- *Frequency (Monobit) Test*

The purpose of this test is to determine whether the binary sequence consists of independent identically distributed Bernoulli random variables or not.

- *Frequency Test within a Block*

This test first divides the binary sequence into several non-overlapping subsequences and then utilizes chi-square test to examine whether the probability of bit '1' is approximately 1/2 in each subsequence to detect localized deviations from the ideal 50% frequency.

- *Runs Test*

The purpose of this test is to determine whether the oscillation of various-length runs of bit '1' and '0' is too fast or too low, where a run denotes an uninterrupted sequence of identical bits.

- *Test for the Longest-Run-of-*Ones *in a Block*

This test first divides the binary sequence into several non-overlapping tested subsequences and then examines whether the length of the longest run of bit '1' within the tested subsequence is consistent.

- *Binary Matrix Rank Test*

The purpose of this test is to check for linear dependence among fixed length substrings of the original sequences.

- *Discrete Fourier Transform (Spectral) Test*

This test is based on Discrete Fourier Transform. The purpose of this test is to detect whether periodic features (repetitive patterns) exists in the tested sequence by adopting the following equation: the number of peaks exceeding the 95% threshold is significantly different than 5%.

- *Non-overlapping Template Matching Test*

The purposed of this test is to detect whether too many occurrences of a given non-periodic pattern are happened.

- *Overlapping Template* Matching *Test*

The purposed of this test is to detect whether too many or too few occurrences of m-bit runs of bit '1' are happened, where m is an arbitrary value.

- *Maurer's Universal Statistical Test*

This test is related to the per-bit entropy of the stream which is the correct quality measure for a secret-key source in a cryptographic application. The purpose of the test is to detect whether the sequence can be significantly compressed without loss of information or not. A significantly compressible sequence is considered to be non-random.

- *Lempel-Ziv Compression Test*

The purpose of the test is to examine how far the tested sequence can be compressed. The sequence is considered to have a characteristic number of distinct patterns (non-random) if it can be significantly compressed.

- *Linear Complexity Test*

The purpose of this test is to validate whether the sequence can be characterized by longer LFSRs (Linear Fedback ShiftRgister) or not. The shorter of LFSR is, the less randomness of the tested sequence is.

- *Serial Test*

The focus of this test is the frequency of all possible overlapping m-bit patterns across the entire sequence, where m is an arbitrary value. The purpose of this test is to determine whether the random sequences have uniformity; that is, every m-bit pattern has the same chance of appearing as every other m-bit pattern.

- *Approximate Entropy Test*

The focus of this test is the frequency of all possible overlapping m-bit patterns across the entire sequence, where m is an arbitrary value. The purpose of the test is to compare the frequency of overlapping blocks of two consecutive lengths (m and m+1) against the expected theoretic result for a random sequence.

- *Cumulative Sums (Cumsums) Test*

The focus of this test is the maximal excursion (from zero) of the random walk defined by the cumulative sum of adjusted (-1, +1) digits in the sequence. The purpose of the test is to determine whether the cumulative sum of the partial sequences occurring in the tested sequence is too large or too small.

- *Random Excursions Test*

In this test, the cumulative sum random walk is derived from partial sums after the (0, 1) sequence is transferred to the appropriate (-1, +1) sequence. The purpose of this test is to decide if the number of visits to a particular state within a cycle deviates from the expected theoretic value for a random sequence, where a cycle of a random walk consists of a sequence of steps of unit length taken at random that begin at and return to the origin.

- *Random Excursions Variant Test*

The focus of this test is the total number of times that a particular state is visited (i.e., occurs) in a cumulative sum random walk. The purpose of this test is to detect deviations from the expected number of visits to various states in the random walk.

Table 1 briefly summarizes the results of the randomness evaluation of our scheme. Obviously, the proposed protocol passes all statistical testing. The results indicate that the output messages of our scheme are not easily identified as a set of distinguishable patterns for the eavesdropper/cryptanalyst. In addition, table 2 shows the comparison among our proposed scheme, SASI scheme and Gossamer protocol in accordance with the security and system efficiency requirements. Apparently, our protocol is superior to SASI and Gossamer protocols by supporting all security requirements with lower computation cost and less tag memory exploiting.

Table 1. The results of randomness testing

Statistical testing	P-value	Result
Frequency (Monobit) Test	0.678874*	Pass
Frequency Test within a Block	0.535768*	Pass
Runs Test	0.980716*	Pass
Test for the Longest-Run-of-Ones in a Block	0.651698*	Pass
Binary Matrix Rank Test	0.931527*	Pass
Discrete Fourier Transform (Spectral) Test	0.192145*	Pass
Non-overlapping Template Matching Test	0.464268*	Pass
Overlapping Template Matching Test	0.794579*	Pass
Maurer's Universal Statistical Test	0.258872*	Pass
Lempel-Ziv Compression Test	0.628152*	Pass
Linear Complexity Test	0.159199*	Pass
Serial Test	0.970976*	Pass
Approximate Entropy Test	0.832106*	Pass
Cumulative Sums (Cumsums) Test	0.727083*	Pass
Random Excursions Test	0.332527*	Pass
Random Excursions Variant Test	0.623045*	Pass

* denotes significant (> 0.05)

Table 2. Comparison among our scheme, SASI protocol and Gossamer protocol

		SASI [11]	Gossamer [38]	Our protocol
Resistance to desynchronization attacks		X	X	O
Resistance to disclosure attack		X	O	O
Anonymity		O	O	O
Mutual authentication and forward Security		O	O	O
Randomness testing		X	X	O
Transmission rounds / total transmission message sizes		4 / 5L	4 / 5L	5 / 5L+1
Memory consumed on tag		7L	7L	3L
Memory consumed on backend database		4L	4L	4L
Number of operations on tag/database	Random number	2	2	2
	Rot function	2	18	2
	Addition (+) mod 2^m	4	44	3
	XOR operation	10	6	10
	AND operation	2	0	0
	OR operation	0	0	1
	MixBits function	0	3	0
O: provide X: not provide L: the bit-length of pseudonym *IDS*, shared secret key *k* and identity *ID*				

Conclusion

In this study, we develop a secure and privacy-aware ultralightweight RFID authentication scheme. From the aspect of tag memory utilization, our protocol reduces around 57.14% memory consumption at tag side in comparison with SASI and Gossamer mechanisms. With the results of randomness evaluation, we have proved that the outputs of our proposed authentication scheme preserve excellent randomness property; accordingly, our protocol can withstand all passive attacks. In addition, based on our security analysis, the proposed scheme provides mutual authentication, data confidentiality, tag anonymity and forward security.

References

[1] Y. An and S. Oh, RFID System for User's Privacy Protection, *Proc. of the Asia-Pacific Conference on Communications* (2005), 516-519.

[2] G. Avoine, E. Dysli and P. Oechslin, Reducing Time Complexity in RFID Systems, *Proc. of the Workshop on SAC'05* (2005).

[3] J. Bringer, H. Chabanne and E. Dottax, HB++: A Lightweight Authentication Protocol Secure against Some Attacks, *Proc. of the 2nd International Workshop on Security, Privacy and Trust in Pervasive and Ubiquitous Computing* (2006), 28-33.

[4] M. Burmester and B. de Medeiros, 'The Security of EPC Gen2 Compliant RFID Protocols, *LNCS 5037*, *Proc. of the 6th International Conference of Applied Cryptography and Network Security* (2008), 490-506.

[5] T. Cao, E. Bertino and H. Lei, Security Analysis of the SASI Protocol, *IEEE Transactions on Dependable and Secure Computing* 6 (2008), 73-77.

[6] C.-L. Chen and Y.-Y. Den, Conformation of EPC Class 1 Generation 2 Standards RFID System with Mutual Authentication and Privacy Protection, *Engineering Applications of Artificial Intelligence* 22 (2009), 1284-1291.

[7] Y.-C. Chen, W.-L. Wang and M.-S. Hwang, RFID Authentication Protocol for Anti-Counterfeiting and Privacy Protection, *Proc. of the ICACT'07* (2007), 255–259.

[8] Y. Chen, J.-S. Chou and H.-M. Sun, A Novel Mutual Authentication Scheme based on Quadratic Residues for RFID Systems, *Computer Networks* 52, no.12, 2008, 2373-2380.

[9] H.-Y. Chien and C.-H. Chen, Mutual Authentication Protocol for RFID Conforming to EPC Class 1 Generation 2 Standards, *Computer Standards & Interfaces* 29 (2007), 254-259.

[10] H.-Y. Chien and C.-W. Huang, Security of Ultra-Lightweight RFID Authentication Protocols and its Improvements, *ACM SIGOPS Operating System Review* 41 (2007), 83–86.

[11] H.-Y. Chien, SASI: A New Ultralightweight RFID Authentication Protocol Providing Strong Authentication and Strong Integrity, *IEEE Trans. On Dependable and Secure Computing* 4 (2007), 337-340.

[12] M. Conti, R. D. Pietro and L. V. Mancini, RIPP-FS: an RFID Identification, Privacy Preserving protocol with Forward Secrecy, *Proc. of the IEEE PerComW'07* (2007), 229–234.

[13] P. D'Arco and A. De Santis, From Weaknesses to Secret Disclosure in a Recent Ultra-Lightweight RFID Authentication Protocol, *Cryptology ePrint Archive* 470 (2008).

[14] D. N. Duc, J. Park, H. Lee and K. Kim, Enhancing Security of EPCglobal GEN-2 RFID Tag against Traceability and Cloning, *Proc. of the Symposium on Cryptography and Information Security* (2006).

[15] *EPC™ Radio-Frequency Identification Protocols Class 1 Generation-2 UHF RFID Protocol for Communication at 860-960 MHz Version 1.0.9*, EPCGlobal Inc., Dec. 2005.

[16] S. L. Garfinkel, A. Juels and R. Pappu, RFID Privacy: An overview of Problems and Proposed Solutions, *IEEE Security & Privacy Magazine* 3 (2005), 34-43.

[17] H. Gilbert, M. Robshaw and H. Sibert, An Active Attack against HB+-A Provably Secure Lightweight Authentication Protocol, *Cryptology ePrint Archive* (2005).

[18] D. Han and D. Kwon, Vulnerability of An RFID Authentication Protocol Conforming to EPC Class 1 Generation 2 Standards, *Computer Standards & Interfaces* 31 (2009), 648-652.

[19] D. Henrici and P. Műller, Hash-based Enhancement of Location Privacy for Radio-Frequency Identification Devices using Varying Identifiers, *Proc. of the Workshop on IEEE PerSec'04* (2004).

[20] J.C. Hernandex-Castro, J.M. Estevex-Tapiador, P. Peris-Lopez and J.-J. Quisquater, Cryptanalysis of the SASI ultralightweight RFID Authentication Protocol, *arXiv:0811.4257v1* (2008).

[21] A. Juels and S.A. Weis, Authenticating Pervasive Devices with Human Protocols, *Proc. of the CRYPTO'05* (2005), 293-308.

[22] S. Karthikeyan and M. Nesterenko, RFID Security without Extensive Cryptography, *Proc. Of ACM Workshop on Security of Ad Hoc and Sensor Networks* (2005), 63-67.

[23] H.-W. Kim, S.-Y. Lim and H.-J. Lee, Symmetric Encryption in RFID Authentication Protocol for Strong Location Privacy and Forward-Security, *Proc. of the ICHIT'06* (2006), 718–723.

[24] T. V. Le, M. Burmester and B. de Medeiros, Universally Composable and Forward-secure RFID authentication and Authenticated Key Exchange, *Proc. of the ASIACCS'07* (2007).

[25] Y.-C. Lee, Y.-C. Hsieh, P.-S. You and T.-C. Chen, A New Ultralightweight RFID Protocol with Mutual Authentication, *Proc. of WASE'09* (2009), 58-61.

[26] S. Lee, T. Asano and K. Kim, RFID Mutual Authentication Scheme based on Synchronized Secret Information, *Proc. of the Symposium on Cryptography and Information Security* (2006).

[27] T. Li and R.H. Deng, Vulnerability Analysis of EMAP-An Efficient RFID Mutual Authentication Protocol, *Proc. of the AReS'07* (2007).

[28] T. Li and G. Wang, Security Analysis of Two Ultra-Lightweight RFID Authentication Protocols, *Proc. of the IFIP Information Security* (2007).

[29] N.W. Lo and K.-H. Yeh, Novel RFID authentication Schemes for Security Enhancement and System Efficiency, *Proc. of the SDM'07* (2007), 203-212.
[30] N.W. Lo and K.-H. Yeh, An Efficient Mutual Authentication Scheme for EPCglobal Class-1 Generation-2 RFID System, *Proc. of the TRUST'07* (2007), 43-56.
[31] J. Munilla and A. Peinado, HB-MP: A Further Step in the HB-Family of Lightweight Authentication Protocols, *Computer Networks* (2007).
[32] M. Ohkubo, K. Suzki and S. Kinoshita, Cryptographic Approach to Privacyfriendly Tags, *Proc. of the RFID Privacy Workshop* (2003).
[33] K. Osaka, T. Takagi, K. Yamazaki and O. Takahashi, An Efficient and Secure RFID Security Method with Ownership Transfer, *Proc. of the IEEE ICCIAS* (2006), 1090–1095.
[34] J.-S. Park and I.-Y. Lee, RFID Authentication Protocol Using ID Synchronization in Insure Communication, *Proc. of the ICHIT'06* (2006), 664 – 667.
[35] P. Peris-Lopez, J.C. Hernandex-Castro, J.M. Estevex-Tapiador and A. Ribagorda, LMAP: A Real Lightweight Mutual Authentication Protocol for Low-Cost RFID Tags, *Proc. of the 2nd Workshop RFID Security* (2006).
[36] P. Peris-Lopez, J.C. Hernandex-Castro, J.M. Estevex-Tapiador and A. Ribagorda, EMAP: An Efficient Mutual Authentication Protocol for Low-Cost Tags, *Proc. of the OTM Federated Conf. and workshop: IS Workshop* (2006).
[37] P. Peris-Lopez, J.C. Hernandex-Castro, J.M. Estevex-Tapiador and A. Ribagorda, M2AP: A Minimalist Mutual-Authentication Protocol for Low-Cost RFID Tags, *Proc. of the UIC'06* (2006), 912-923.
[38] P. Peris-Lopez, J.C. Hernandex-Castro, J.M. Estevex-Tapiador and A. Ribagorda, Advances in Ultralightweight Cryptography for Low-cost RFID Tags: Gossamer Protocol, *Proc. of the Workshop Information Security Applications* (2008).
[39] P. Peris-Lopez, J.C. Hernandez-Castro, J.M. Estevez-Tapiador, T. Li and J.C.A. van der Lubbe. Weaknesses in Two Recent Lightweight RFID Authentication Protocols. *Proc. of Workshop on RFID Security* (2009).
[40] C.-W. Phan, Cryptanalysis of a New Ultralightweight RFID Authentication Protocol – SASI, *IEEE Transactions on Dependable and Secure Computing* 6 (2009), 316-320.
[41] S. Piramuthu, HB and Related Lightweight Authentication Protocols for Secure RFID Tag/Reader Authentication, *Proc. of the CollECTeR Europe Conference* (2006).
[42] J. Yang, J. Park, H. Lee, K. Ren and K. Kim, Mutual Authentication Protocol for Low-cost RFID, *Proc. of the Workshop on RFID and Lightweight Crypto* (2005).
[43] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, E. Barker, S. Leigh, M. Levenson, M. Vangel, D. Banks, A. Heckert, J. Dray and S. Vo, A Statistical Test Suite for Random and Pseudorandom Number Generators, *NIST Special Publication 800-22* (2000).
[44] H.-M. Sun, W.-C. Ting and K.-H. Wang, On the Security of Chien's ultralightweight RFID Authentication Protocol, *Cryptology ePrint Archive* (2008).
[45] S. A. Weis, S. E. Sarma, R. L. Rivest and D. W. Engels, Security and Privacy Aspects of Low-Cost Radio Frequency Identification Systems, *Proc. of the Security in Pervasive Computing* (2003), 201-212.
[46] K.-H. Yeh and N.W. Lo, Improvement of Two Lightweight RFID Authentication Protocols, *Information Assurance and Security Letters* (2010).

Radio Frequency Identification System Security
Y. Li and J. Zhou (Eds.)
IOS Press, 2010

doi:10.3233/978-1-60750-485-6-61

Faster CRT-RSA Decryption towards RFID applications

Subhamoy MAITRA [a]
[a] *Applied Statistics Unit, Indian Statistical Institute,*
203 B T Road, Kolkata 700 108, India.
E-mail: subho@isical.ac.in

Santanu SARKAR [b]
[b] *Applied Statistics Unit, Indian Statistical Institute,*
203 B T Road, Kolkata 700 108, India.
E-mail: santanu_r@isical.ac.in

Morshed U. CHOWDHURY [c,1]
[c] *School of Information Technology, Deakin University-Melbourne Campus*
221 Burwood Highway, Burwood, Victoria 3125, Australia.
E-mail: morshed.chowdhury@deakin.edu.au

Abstract. In this paper we present a strategy to design the RSA parameters in such a manner so that the CRT-RSA decryption becomes more efficient than the existing methods. We achieve around 21% improvement in speed over the currently best known implementation strategy for CRT-RSA decryption with our properly chosen parameters that also helps in terms of less memory requirement. Moreover, we argue in detail the cryptographic security regarding our choice of the secret parameters.

Keywords. Cryptography, CRT-RSA, Factorization, Fast Decryption, RFID, RSA.

1. Introduction

RFID (Radio Frequency Identification) is a modern and fast growing mobile technology that uniquely and accurately identifies a RFID tag attached or embedded to an object or a container. The great appeal of RFID technology is that it allows information to be remotely stored, read and updated without requiring either physical or visual contact. Each tag contains an antenna and a tiny microchip smaller than a grain of sand. An RFID reader uses radio waves to retrieve object data from or write data to an object tag via wireless communication. The use of RFID technology can be employed for not only reducing management costs for organizations but also tracking each container, pallet, case, product being manufactured, shipped and sold, in order to increase visibility and

[1]This work has been performed when the author was visiting Indian Statistical Institute, Kolkata during August–December, 2009.

accountability in the supply chain. Application of RFID technology is not limited to the supply chain management system only. It can be used in many economic, health and environments sectors such as homeland security (RFID enabled passports), e-business (RFID enable credit cards), e-cash (RFID enabled bank notes), e-health system (RFID enable emergency health care system) etc.

Despite these potential benefits, there are challenges and obstacles with the deployment of RFID enabled system in the all above mentioned sectors. One of the major challenges with the RFID system is that it poses new serious security threats to RFID-enable system applications or infrastructures [22]. Without the appropriate security controls, it is quite easy to manipulate RFID-based systems (e.g., retail checkout system) by either cloning the RFID tags, modifying existing tag data, or by preventing RFID tags from being read in the first place. These security threats make businesses reluctant to use RFID technology and as a result have hampered deployment of RFID in a broad range of applications where product data confidentiality and consumer privacy are prime requirements.

The communication between the reader and the tag is through the wireless channel and may be intercepted by adversaries. Thus, we need secured and authenticated communication between the reader and the tag. It is expected, in near future, the use of active RFID tags will be frequent and reasonable amount of computation will be possible in such tags. Secured and authenticated communication between two parties is possible with symmetric key cryptosystem, but it is necessary to pre-distribute the secret keys in the tags. If key pre-distribution is not possible, then some kind of public key infrastructure is required and application of RSA is quite well known in this area. We concentrate on how fast RSA decryption is possible on a low end system. To be specific, our proposal is to implement RSA decryption algorithm on active RFID tags so that the tags can perform decryption using private key or sign on a message.

Our strategy provides the following advantages so that the RSA decryption can be deployed on a low end device such as an active RFID tag.

- The decryption exponents are much smaller than the largest possible size. As an example, for primes p, q of the size of 512 bits, we can consider d_p, d_q of size 192 bits only (when the maximum size of d_p, d_q can be as large as 512 bits). Further, d_p, d_q will have almost all the bits same except a few. These ideas will help in storing the private keys in a limited storage space.
- The overall proposal is towards making the RSA decryption much faster than the existing strategies, as well as much less power consuming. This will help in deploying the RSA decryption on low end CPUs (less clock speed) and also when the power availability is constrained, which is a frequent scenario on RFID tags.

One may also note that all these advantages are achieved without any compromise on security. We present detailed security analysis of our scheme to convince this.

Before proceeding further, let us briefly introduce RSA. RSA [23] is the most popular public key cryptosystem which has frequent application in many areas of secure communication. The RSA cryptosystem can be briefly described as follows. One may note that the plaintext M requires some pre-processing to make RSA secure (see for example [7]). That is, instead of using M in the following description, one may use some properly designed mapping $\mu(M)$ of M.

- primes p, q, (generally the primes are considered to be of same bit size, i.e., $q < p < 2q$);

- $N = pq$, $\phi(N) = (p-1)(q-1)$;
- e, d are such that $ed = 1 + k\phi(N)$, $k \geq 1$;
- N, e are publicly available and plaintext M is encrypted as $C \equiv M^e \bmod N$;
- the secret key d is required to decrypt the ciphertext as $M \equiv C^d \bmod N$.

The study of RSA is one of the most attractive areas in cryptology research as evident from many excellent works (one may refer [3,12,18] and the references therein for detailed information).

Speeding up RSA [23] encryption and decryption is of serious interest and for large N, both e, d cannot be small at the same time. For fast decryption, the value of d need to be small. However, Wiener [27] showed that when $d < \frac{1}{3}N^{\frac{1}{4}}$ then N can be factored easily. Later, Boneh-Durfee [2] increased this bound up to $d < N^{0.292}$. Thus use of smaller d is in general not recommendable. In this direction, an alternative approach has been proposed by Wiener [27] exploiting the Chinese Remainder Theorem (CRT) for decryption. This variant of RSA is popularly known as CRT-RSA and it can be described as follows:

- the public exponent e and the private CRT exponents d_p and d_q are used satisfying $ed_p \equiv 1 \bmod (p-1)$ and $ed_q \equiv 1 \bmod (q-1)$;
- the encryption is same as standard RSA;
- to decrypt a ciphertext C one needs to compute $M_1 \equiv C^{d_p} \bmod p$ and $M_2 \equiv C^{d_q} \bmod q$;
- using CRT, one can get the plaintext $M \in \mathbb{Z}^n$ such that $M \equiv M_1 \bmod p$ and $M \equiv M_2 \bmod q$.

We mainly concentrate on the CRT-RSA decryption and try to select the secret primes p, q and the secret exponents d_p, d_q in such a manner so that the operation becomes faster. The basic idea is to put large number of zeros in the bit pattern of p, q, d_p, d_q without compromising the security.

Towards a secure design, we need to revisit the cryptanalytic results on CRT-RSA. There exists a meet-in-the-middle attack [14] enabling the adversary to factor N in time and space of order of $\min(\sqrt{d_p}, \sqrt{d_q})$. In [8], an attack on CRT-RSA has been presented for small e when the primes are of same bit size. Recently, Jochemsz and May [13] presented an attack on CRT-RSA with primes of same bit size in poly$(\log N)$ time. In [13], it is shown that CRT-RSA can be attacked when the encryption exponents are of the order of N, and d_p and d_q are smaller than $N^{0.073}$. Qiao and Lam [21] proposed to use d_p and d_q such that $d_p - d_q = 2$ for fast signature generation on a low-cost smart card. For 1024 bits N, the size of d_p and d_q have been suggested to be around 128 bits in [21] to counter act the meet-in-the-middle attack as they found that 96 bits should be enough to provide sufficient security of attack complexity around 2^{48}. In [12], authors proved that when $d_p - d_q$ is known to the attacker, then one can factor N in polynomial time when $d_p, d_q < N^{0.099}$. Now $1024 \times 0.099 \approx 101$ and thus the scheme proposed in [21] with d_p, d_q less 96 bits is not secure. Further, trying exhaustive search on a few Most Significant Bits (MSBs) on d_p, d_q, the cryptanalysis of [12] can be extended further.

As we will require the security analysis of our proposal, we need the following background on lattice reduction techniques.

1.1. Preliminaries on Lattice Based Techniques

Consider the linearly independent vectors $u_1, \ldots, u_w \in \mathbb{Z}^n$, when $w \leq n$. A lattice, spanned by $\{u_1, \ldots, u_w\}$, is the set of all linear combinations of $u_1, \ldots, u_w$, i.e., w is the dimension of the lattice. A lattice is called full rank when $w = n$. Let L be a lattice spanned by linearly independent vectors $u_1, \ldots, u_w$, where $u_1, \ldots, u_w \in \mathbb{Z}^n$. By $u_1^*, \ldots, u_w^*$, we denote the vectors obtained by applying the Gram-Schmidt process to the vectors $u_1, \ldots, u_w$. It is known that given a basis $u_1, \ldots, u_w$ of a lattice L, LLL algorithm [15] can find a new basis $b_1, \ldots, b_w$ of L with the following properties.

- $\| b_i^* \|^2 \leq 2 \| b_{i+1}^* \|^2$, for $1 \leq i < w$.
- For all i, if $b_i = b_i^* + \sum_{j=1}^{i-1} \mu_{i,j} b_j^*$ then $|\mu_{i,j}| \leq \frac{1}{2}$ for all j.
- $\| b_1 \| \leq 2^{\frac{w}{2}} \det(L)^{\frac{1}{w}}$, $\| b_2 \| \leq 2^{\frac{w}{2}} \det(L)^{\frac{1}{w-1}}$.

By $b_1^*, \ldots, b_w^*$, we mean the vectors obtained by applying the Gram-Schmidt process to the vectors $b_1, \ldots, b_w$.

The determinant of L is defined as $\det(L) = \prod_{i=1}^{w} ||u_i^*||$, where $||.||$ denotes the Euclidean norm on vectors. Given a polynomial $g(x, y) = \sum a_{i,j} x^i y^j$, we define the Euclidean norm as $\| g(x, y) \| = \sqrt{\sum_{i,j} a_{i,j}^2}$ and infinity norm as $\| g(x, y) \|_\infty = \max_{i,j} |a_{i,j}|$.

In [6], techniques have been discussed to find small integer roots of polynomials in a single variable $\bmod n$, and of polynomials in two variables over the integers. The idea of [6] extends to more than two variables also, but the method becomes probabilistic. The following theorem is also relevant to the idea of [6].

Theorem 1 *[10] Let $g(x, y, z)$ be a polynomial which is a sum of ω monomials. Suppose $g(x_0, y_0, z_0) \equiv 0 \bmod n$, where $|x_0| < X$, $|y_0| < Y$ and $|z_0| < Z$. If $\| g(xX, yY, zZ) \| < \frac{n}{\sqrt{\omega}}$, then $g(x_0, y_0, z_0) = 0$ holds over integers.*

Thus, the condition $2^{\frac{w}{2}} \det(L)^{\frac{1}{w-1}} < \frac{n}{\sqrt{\omega}}$ implies that if polynomials b_1, b_2 (corresponding to the two shortest reduced basis vectors) have roots over $0 \bmod n$, then those roots hold over integers also. Our approach relies on heuristic assumption for computations with multivariate polynomials. We formally state the following assumption that we will consider for the theoretical results.

Assumption 1. Consider a set of polynomials $\{f_1, f_2, f_3\}$ on 3 variables having the roots of the form (x_0, y_0, z_0). Let J be the ideal generated by $\{f_1, f_2, f_3\}$. Then we will be able to collect the roots efficiently from the Gröbner Basis of J.

2. The strategy for fast decryption

In this section we present ideas how CRT-RSA decryption can be made faster than the usual case. Since we consider that the CRT-RSA decryption keys will be stored in the active RFID tag, we will try to minimize the space requirement. Further, the decryption process needs to be efficient. The decryption process requires two modular exponentiations and one application of CRT.

For any integer i, let us denote its bit size by l_i and the number of 1's in its bit pattern as w_i. Given two positive integers a, b, calculation of $x^a \bmod b$ requires l_a squarings and

w_a multiplications (both modular operations). Thus, the calculation of $x^a \bmod b$ needs $l_a + w_a$ many multiplications considering squaring and multiplication of large integers are of same time complexity [19, Section 14.18]. In average case, one may consider $w_a = \frac{l_a}{2}$ and hence the total cost is around $\frac{3}{2}l_a$ many multiplications.

As evident from literature (see [11] and the references therein), we consider (i) p, q are of same bit size and also (ii) the secret exponents d_p, d_q are of same bit size. To compute $C^{d_p} \bmod p$ and $C^{d_q} \bmod q$ one requires $(l_{d_p} + w_{d_p}) + (l_{d_q} + w_{d_q})$ many multiplications. Below we present how we can reduce this cost.

Proposal 1 *If the number of 1's is significantly small in each of d_p, d_q, then w_{d_p}, w_{d_q} will be small and hence the total number of multiplications will significantly reduced. We also consider a security parameter η, such that $w_{d_p}, w_{d_q} \geq \eta$.*

It should be noted that w_{d_p} or w_{d_q} should not be so small so that one can search a small space to guess the exponents d_p or d_q. Thus the lower bound η on the number of 1's in the decryption exponents has to be decided with proper care for enough security margin. We will discuss this later in Example 1.

Storage is also an important constraint in low end devices. Towards this motivation, in [21], a proposal has been presented to consider $d_p - d_q = 2$. This is because, it is enough to store only d_p and then d_q can be easily derived from d_p. However, the scheme of [21] did not consider how the exponentiations can be implemented efficiently and further it has been noted in [12] that for certain parameters, the proposal becomes insecure. However, with our design strategy, we find that for proper choice of parameters, one can design a secure system with $d_p - d_q = 2$. In a general framework, our design strategy is as follows.

Proposal 2 *The choice of d_p, d_q should be such that $d_p - d_q = \tau$, where τ is a small integer. Thus storing d_p, τ is enough instead of storing d_p, d_q.*

Since all the operations are modular, one needs to study how the $\bmod b$ part in the calculation of $x^2 \bmod b$ or $xy \bmod b$ can be done efficiently, where $x, y \in \mathbb{Z}_b$. It has been pointed out by A. Lenstra [16] that the operation becomes efficient when b is of the form $b = 2^{l_b - 1} + t$ for some positive integer t which is significantly smaller than $2^{l_b - 1}$. Then the reduction of modulo b of a number of approximately $2l_b$ many bits (as x, y are l_b bit integers, x^2 or xy will be of $2l_b$ bits) will be $\frac{l_b}{l_t}$ times faster by the method of [16] than the ordinary or Montgomery reduction [20]. Thus our strategy of choosing the secret primes is as follows.

Proposal 3 *The choice of the secret primes p, q of same bit size should be such that certain portion of the MSBs (leaving the MSB) should be zero. That is, $p = 2^{l_p - 1} + t_p$ and $q = 2^{l_q - 1} + t_q$, where $t_p < p$ and $t_q < q$. However, $p - q$ cannot be made very small, as in that case there are certain weaknesses in the system. Thus we need $\gamma l_p < l_{t_p - t_q} < l_p$ where $\frac{1}{2} < \gamma < 1$.*

In our proposal, we take $p = 2^{l_p - 1} + t_p$ with t_p quite smaller than $2^{l_p - 1}$. In standard arithmetic this is around $\frac{l_p}{l_{t_p}}$ times faster than ordinary or Montgomery reduction modulo numbers of the same size. Similar case will be considered for q. On the other hand, for security reason, $l_{t_p - t_q}$ should not be less than $\frac{l_p}{2}$ as in that case N will be factored by Fermat's method [25].

3. Security Analysis

In this section we analyse the cryptographic security of the RSA parameters that we choose for faster decryption. Since $ed_p \equiv 1 \bmod (p-1)$ and $ed_q \equiv 1 \bmod (q-1)$ one can write $ed_p = 1 + k_p(p-1)$ and $ed_q = 1 + k_q(q-1)$. We start with the following technical result from [24, Lemma 1].

Lemma 1 *Let $e < N$ and $d_p < N^{\lambda}$. Consider that d_{p_0}, d_{q_0}, p_0 are exposed such that $|d_p - d_{p_0}| < N^{\beta}$, $|d_q - d_{q_0}| < N^{\beta}$ and $|p - p_0| < N^{\delta}$. Then one can find the integers k_{p_0}, k_{q_0} such that $|k_p - k_{p_0}| < (1+\sqrt{2})N^{\gamma}$ and $|k_q - k_{q_0}| < (1+\sqrt{2})N^{\gamma}$, where $\gamma = \max\{\lambda + \delta, \beta + \frac{1}{2}\}$.*

Now we present the following theorem which is required for the security analysis.

Theorem 2 *Let $e < N$ and $d_p, d_q < N^{\lambda}$. Consider that d_{p_0}, d_{q_0}, p_0 are exposed such that $|d_p - d_{p_0}| < N^{\beta}$, $|d_q - d_{q_0}| < N^{\beta}$ and $|p - p_0| < N^{\delta}$. Let $\gamma = \max\{\lambda + \delta, \beta + \frac{1}{2}\}$. Then, under Assumption 1, one can factor N in poly$(\log N)$ time when $d_p - d_q$ is known and*

$$4\beta^2 - 32\beta\gamma - 48\gamma^2 + 24\beta\lambda + 48\gamma\lambda - 12\lambda^2 + 36\beta + 72\gamma - 36\lambda - 27 < 0,$$

with $\frac{9}{2} + 3\lambda - 9\beta - 6\gamma \geq 0$.

Proof: We have $ed_p = 1 + k_p(p-1), ed_q = 1 + k_q(q-1)$. Hence $ed_p - 1 + k_p = k_p p, ed_p - ce - 1 + k_q = k_q q$, where $c = d_p - d_q$. Multiplying the above two equations we get

$$(1+ce) - (2e+ce^2)d_p + e^2 d_p^2 - (ce+1)k_p - k_q + ed_p k_p + ed_p k_q + (1-N)k_p k_q = 0. \quad (1)$$

Since we know the approximation d_{p_0} of d_p and p_0 of p, from Lemma 1 we can find an approximation k_{p_0} of k_p such that $|k_p - k_{p_0}|$ is $O(N^{\gamma})$ where $k_{p_0} = \lfloor \frac{ed_{p_0}-1}{p_0-1} \rfloor$ and $\gamma = \max\{\lambda + \delta, \beta + \frac{1}{2}\}$. Similarly we can find an approximation k_{q_0} of k_q, such that $|k_q - k_{q_0}|$ is $O(N^{\gamma})$ where $k_{q_0} = \lfloor \frac{ed_{q_0}-1}{q_0-1} \rfloor$.

Let $d_{p_1} = d_p - d_{p_0}, k_{p_1} = k_p - k_{p_0}$ and $k_{q_1} = k_q - k_{q_0}$. So we can rewrite Equation (1) as $(1+ce) - (2e+ce^2)(d_{p_0} + d_{p_1}) + e^2(d_{p_0} + d_{p_1})^2 - (ce+1)(k_{p_0} + k_{p_1}) - (k_{q_0} + k_{q_1}) + e(d_{p_0} + d_{p_1})(k_{p_0} + k_{p_1}) + e(d_{p_0} + d_{p_1})(k_{q_0} + k_{q_1}) + (1-N)(k_{p_0} + k_{p_1})(k_{q_0} + k_{q_1}) = 0$.

Hence we want to find the root $(d_{p_1}, k_{p_1}, k_{q_1})$ of the polynomial $f(x,y,z) = (1+ce) - (2e+ce^2)(d_{p_0} + x) + e^2(d_{p_0} + x)^2 - (ce+1)(k_{p_0} + y) - (k_{q_0} + z) + e(d_{p_0} + x)(k_{p_0} + y) + e(d_{p_0} + x)(k_{q_0} + z) + (1-N)(k_{p_0} + y)(k_{q_0} + z) = (-ce^2 + 2e^2 d_{p_0} + ek_{p_0} + ek_{q_0} - 2e)x + e^2x^2 + (-ce + ed_{p_0} - k_{q_0}N + k_{q_0} - 1)y + (ed_{p_0} - k_{p_0}N + k_{p_0} - 1)z + exy + exz + (1-N)yz + R$, where $R = -ce^2 d_{p_0} + e^2 d_{p_0}^2 - cek_{p_0} + ed_{p_0}k_{p_0} + ed_{p_0}k_{q_0} - k_{p_0}k_{q_0}N + ce - 2ed_{p_0} + k_{p_0}k_{q_0} - k_{p_0} - k_{q_0} + 1$ is a known constant.

Let $X = N^{\beta}, Y = Z = N^{\gamma}$. Clearly X, Y, Z are upper bounds of desired root, neglecting the small constants. Let $W = ||f(xX, yY, zZ)||_{\infty}$. Clearly $W \geq |-ce + ed_{p_0} - k_{q_0}N + k_{q_0} - 1|Y \approx k_{q_0}NY = N^{\frac{1}{2}+\lambda+1+\gamma}$ (considering e is of full bit-size) $= N^{\frac{3}{2}+\lambda+\gamma}$. Note that this polynomial has the same monomials as of $f(x,y,z)$ presented in [12, Section 4.2], though the coefficients are different. Also, the upper bounds on the

variables are different. We apply extra shifts on x as advised in the "Extended Strategy" of [12, Section 3]. In this direction we define the following as in [12]:

$$S = \bigcup_{0 \le j \le t} \{x^{i_1+j} y^{i_2} z^{i_3} : x^{i_1} y^{i_2} z^{i_3} \text{ is a monomial of } f^{m-1}\},$$

$$M = \{\text{monomials of } x^{i_1} y^{i_2} z^{i_3} f : x^{i_1} y^{i_2} z^{i_3} \in S\}.$$

That is, $x^{i_1} y^{i_2} z^{i_3} \in S$ iff $i_2 = 0, \ldots, m-1, i_3 = 0, \ldots, m-1, i_1 = 0, \ldots, 2(m-1) - (i_2 + i_3) + t$, and

$x^{i_1} y^{i_2} z^{i_3} \in M$ iff $i_2 = 0, \ldots, m, i_3 = 0, \ldots, m, i_1 = 0, \ldots, 2m - (i_2 + i_3) + t$, for some non-negative integer t. Let C be the lcm of the monomials in S. Now we define $n = WC(X, Y, Z)$ and $f'(x, y, z) = R^{-1} f(x, y, z) \bmod n$. If $R^{-1} \bmod n$ does not exist then we increase X, Y, Z, W accordingly such that $gcd(X, R) = gcd(Y, R) = gcd(Z, R) = gcd(W, R) = 1$. Note that, $f'(d_{p_1}, k_{p_1}, k_{q_1}) = 0 \bmod n$. Now we define the following shift polynomials:

$$g_{i,j,k}(x, y, z) = x^i y^j z^k f'(x, y, z) \frac{C(X, Y, Z)}{X^i Y^j Z^k}, \text{ for } x^i y^j z^k \in S,$$

$$g'_{i,j,k}(x, y, z) = x^i y^j z^k n, \text{ for } x^i y^j z^k \in M \setminus S.$$

We create a lattice L using the coefficient vectors of $g_{i,j,k}(xX, yY, zZ)$ and $g'_{i,j,k}(xX, yY, zZ)$.

We need to find at least two more polynomials f_0, f_1 that share the same root $(d_{p_1}, k_{p_1}, k_{q_1})$ over the integers. It is known [12, Section 2.2] that these polynomials can be found by lattice reduction over L if $X^{s_1} Y^{s_2} Z^{s_3} < W^s$ for $s_r = \sum_{x^{i_1} y^{i_2} z^{i_3} \in M \setminus S} i_r$, $r = 1, 2, 3$ and $s = |S|$.

For a given integer m and $t = \tau m$, from the definition of S and M and neglecting the lower order terms we have the required condition

$$X^{7+9\tau+3\tau^2} (YZ)^{5+\frac{9}{2}\tau} < W^{3+3\tau}. \tag{2}$$

This inequality is same as the one presented in [12, Section 4.2] due to the same polynomial f used in both in [12] and in our case. However, the bounds on X, Y, Z, W are different than what presented in [12, Section 4.2].

Substituting the values of X, Y, Z and lower bound of W, it is enough to satisfy the following inequality:

$$3\tau^2\beta + 9\tau\beta + 6\tau\gamma - 3\tau\lambda - \frac{9}{2}\tau + 7\beta + 7\gamma - 3\lambda - \frac{9}{2} < 0. \tag{3}$$

The optimal value of τ is $\frac{\frac{9}{2}+3\lambda-9\beta-6\gamma}{6\beta}$. As $\tau \ge 0$ we need $\frac{9}{2} + 3\lambda - 9\beta - 6\gamma \ge 0$. Putting this optimal value of τ, we get the required condition as $4\beta^2 - 32\beta\gamma - 48\gamma^2 + 24\beta\lambda + 48\gamma\lambda - 12\lambda^2 + 36\beta + 72\gamma - 36\lambda - 27 < 0$.

Thus, we can find the root $(d_{p_1}, k_{p_1}, k_{q_1})$ from f_0, f_1 (corresponds the two shortest vectors of the lattice L) and f under Assumption 1. The time complexity poly$(\log N)$ is arrived from the following:

- the time complexity of the lattice reduction is poly$(\log N)$; and
- given the fixed lattice dimension with small and constant size, we get constant degree polynomials; the Gröbner Basis calculation is in general double-exponential in the degree of the polynomial.

This completes the proof. ■

The implication of Theorem 2 is as follows. As we put many constraints on d_p, d_q, p, q we like to study how secure our proposal is when adversary can guess some of the bits of these parameters or get some bits by side channel attacks [1]. As example, if following Proposal 3, we use primes p, q with quite a few MSBs as zero, then an attacker can guess the number of zeros (apart from the MSB) easily by trying it from 1 to l_p. In such a case, the attacker will also try to guess the MSBs of d_p, d_q with the understanding that the bit pattern of d_p, d_q are sparse. Given that RSA with 1024 bit N is quite secure till the year 2014 [5], we consider that as a benchmark in this paper. Number Field Sieve(NFS) [17] is the fastest known factorization algorithm that requires around 2^{86} time complexity. Known meet-in-the-middle attack [14] can factor N in time and space complexity of order of $\min(\sqrt{d_p}, \sqrt{d_q})$. Thus it is quite safe for us to take 192 bits d_p, d_q as the attack of [14] will require around 2^{96} time and space complexity. The reason we take 192-bit decryption exponents as it is divisible by 32 and we can store it in only 6 words of 32 bits each. Implementation of 32-bit words is quite popular in hardware.

We explain the security analysis when each of p, q are 512 bits and our expected lower bound on attack complexity is around 2^{80}. In Table 1, we consider the case when we make some MSBs (apart from the MSB) of p, q zero according to Proposal 3. In such a case the attacker will be able to guess those many bits easily. We denote this by l_p^0, l_q^0. The case $l_p^0 = 0, l_q^0 = 0$ means that we have no constraint on p, q and won't get any advantage explained in Proposal 3. The case $l_p^0 = 100, l_q^0 = 100$ means that apart from the MSB, the next 100 most significant bits in each of p, q are zero and we will get good advantage in terms of implementation.

l_p, l_q	l_p^0, l_q^0	l_{d_p}, l_{d_q}	$l_{d_p}^g, l_{d_q}^g$	$w_{d_p}^g, w_{d_q}^g$
512, 512	0, 0	192, 192	129, 129	25, 24
512, 512	25, 25	192, 192	113, 111	28, 27
512, 512	50, 50	192, 192	93, 91	36, 35

Table 1. Security analysis of our scheme based on Theorem 2 for security lower bound of time complexity 2^{80}.

As the MSBs of p, q are known, following Theorem 2, if some MSBs of d_p, d_q are leaked, then the scheme becomes insecure. As d_p, d_q are sparse, one may search these MSBs. Considering a specific data in Table 1, let us take that 50 many MSBs of p, q are zero (leaving the MSB). Then it is enough for the attacker to concentrate on the 91 many MSBs of d_p, d_q. These guessed/leaked part of d_p, d_q are denoted by $l_{d_p}^g, l_{d_q}^g$ in Table 1. One can calculate that $\binom{88}{36} > 2^{80}$ and thus it is enough to distribute 36 many 1's in the

93 many MSBs in each of d_p, d_q keeping all the other bits 0. The number of 1's in the guessed part of d_p, d_q are denoted by $w^g_{d_p}, w^g_{d_q}$ in Table 1. For this analysis, we consider that $d_p - d_q = 2$, and without loss of generality, we consider $w_{d_p} = w_{d_q} + 1$. Further as d_p, d_q are odd, we have the LSB of each of them as 1.

Similar security analysis need to be done when the attacker tries to guess the LSBs (Least Significant Bits) of d_p, d_q. This we present in the following theorem.

Theorem 3 *Let $e < N$ and $d_p, d_q < N^\lambda$. Consider that $(\lambda - \beta)\log_2 N$ many LSBs of d_p are exposed. Then, under Assumption 1, it is possible to factor N in poly$(\log N)$ time when $d_p - d_q$ is known and*

$$4\beta^2 - 8\beta\lambda - 12\lambda^2 + 20\beta + 12\lambda - 3 < 0$$

with $\frac{3}{2} - 3\lambda - 9\beta \geq 0$.

Proof: Given some LSBs of d_p, the attacker knows d_{p_0}, A such that $d_p = d_{p_1}A + d_{p_0}$ for some known A based on the number of LSBs known. However in such a situation the attacker can not approximate k_p, k_q by knowing the MSBs of p, q. In this case it is enough to know d_{p_1} for the attack.

Similar to the proof of Theorem 2, we get the following function $f(x, y, z)$ putting $d_p = d_{p_1}A + d_{p_0}$. As d_{p_1} is not known, we write it as the variable x. Further, k_p, k_q are identified with the variables y, z. Thus, (d_{p_1}, k_p, k_q) is the root of the equation $f(x, y, z) = (1 + ce) - (2e + ce^2)(d_{p_0} + Ax) + e^2(d_{p_0} + Ax)^2 - (ce + 1)y - z + e(d_{p_0} + Ax)y + e(d_{p_0} + Ax)z + (1 - N)yz$. Suppose $A = N^{\lambda-\beta}$. Let $X = N^\beta, Y = Z = N^{\lambda+\frac{1}{2}}$. Then clearly X, Y, Z are the upper bounds of the desired root. Also $W = ||f(xX, yY, zZ)||_\infty \geq e^2A^2X^2 = N^{2+2\lambda}$. So using same analysis as Theorem 2 we get the required condition as $(7+9\tau+3\tau^2)\beta+(5+\frac{9}{2}\tau)(2\lambda+1) < (3+3\tau)(2+2\lambda)$ i.e., $3\beta\tau^2+9\beta\tau+3\lambda\tau+7\beta+4\lambda-\frac{3}{2}\tau-1 < 0$. The optimal value of τ which is $\frac{\frac{3}{2}-3\lambda-9\beta}{6\beta}$ gives the condition as $4\beta^2 - 8\beta\lambda - 12\lambda^2 + 20\beta + 12\lambda - 3 < 0$ with $\frac{3}{2} - 3\lambda - 9\beta \geq 0$. ■

When d_p, d_q are of 192 bits each and p, q are 512-bit primes, then $\lambda = 0.187$. In such a situation one needs $(\lambda - \beta) \times 1024 = 130$ many LSBs of d_p to be guessed to factor N. One can note that $\binom{130}{24} > 2^{80}$. This provides some additional constraint over the parameters presented in Table 1 and the weight of d_p, d_q need to be increased further.

Let us concentrate on the situation when 50 many MSBs (except the MSB) of p, q are zero (corresponding to third row of data in Table 1). Following Theorem 2, it is enough to have $w_{d_p} = 36$, but we actually need to look into the attack presented in Theorem 3. So if we take $w_{d_p} = 36 + 24 = 60$ and $w_{d_q} = 35 + 24 = 59$, then the scheme will be secure.

Now let us look at the computational advantage. When 50 many MSBs of p, q are known, we have $w_{d_p} = 60, w_{d_q} = 59$. Thus, calculation of $x^{d_p} \bmod p$ requires 192 many square operations and 60 multiplications, i.e., in total 252 many multiplications (considering square and multiplication require equal cost). Similarly, $x^{d_q} \bmod q$ requires 251 multiplications. Thus, the total requirement is 503 many ordinary multiplications. This is according to the advantage described in Proposal 1. Now considering the advantage presented in Proposal 3 for the zeros in the MSBs of p, q, we require $503 \cdot \frac{512-50}{512} = 454$ many equivalent ordinary multiplications effectively.

Following [21], where no constraint on p, q, d_p, d_q has been considered, we can take that half of the bits of d_p, d_q will be 1 on an average. Thus, the number of multiplications required in such a case is $(192+97)+(192+96) = 577$ many ordinary multiplications.

Thus, with proper choice of parameters, we get an advantage of $1 - \frac{454}{577} \approx 0.21$, i.e., a reduction of 21% in CRT-RSA decryption. One may note that the final step of applying CRT is negligible [4] and its computational cost is ignored in our case as well as in the case of [21].

4. Algorithm

Based on our proposals in Section 2 and security analysis in 3, we present the following step by step algorithm. Each step of the algorithm can be executed efficiently, i.e., in poly$(\log N)$ time. One may refer to [19,26] for general background in this area. We will also discuss the requirements of several steps after presenting the algorithm.

1. Select two primes p, q of same bit size with $\gcd(p-1, q-1) = a$ (a is a positive integer) and p, q are of the form $p = 2^{l_p-1} + t_p$ and $q = 2^{l_q-1} + t_q$.
2. Calculate $N = pq$ and $\phi(N) = (p-1)(q-1)$.
3. Take two positive integers d_p, d_q which are relatively prime to $p-1$ and $q-1$ respectively with $d_p \equiv d_q \bmod a$.
4. Find d such that $d \equiv d_p \bmod (p-1)$ and $d \equiv d_q \bmod (q-1)$.
5. Calculate e using Extended Euclidean Algorithm such that $ed \equiv 1 \bmod (\phi(N))$.
6. (e, N) constitute the public key and (p, q, d_p, d_q) constitute the private key.

Since $p-1$ and $q-1$ are not relatively prime to each other one can not use Chinese Remainder Theorem(CRT) directly in step 4. However, $\frac{p-1}{a}$ is relatively prime to $\frac{q-1}{a}$. This is the reason, we need to fix some a in step 1. Let $b \equiv d_p \equiv d_q \bmod a$. Then using CRT, one can find an element d' such that $d' \equiv \frac{d_p-b}{a} \bmod \frac{p-1}{a} \equiv \frac{d_q-b}{a} \bmod \frac{q-1}{a}$. Then in step 4, the required d will be $d = ad' + b$. Also note that in step 3, one can easily find a positive integer d_p (similarly d_q) which is relatively prime to $p-1$ (similarly $q-1$) as $p-1$ has order of $\log \log p$ many prime divisors [9].

Below we present an example that is generated form our algorithm presented above.

Example 1 *We consider 512 bits p, q. The primes p, q are as follows:*
6703903964971303736254660622036150276423392402640004338115434282461219576609422853055510048460313394873443712301886423856945282172785959598605137976416640 3,
6703903964971302875715787951247286240579677788480387626327633603796171995656380658017157656223856838997458921319438120059152815699158843046220412666612637.
One can check there are 50 zeros in binary representation after the MSB of of both p, q.

The decryption exponents are 192 bits, where $d_q = d_p - 2$ and d_p is 372718544839241990754204930109522543728061571860916025808 3. *One can check weight of d_p, d_q are 60 and 59 respectively. We obtain e as an 1021 bit integer as follows:*
2145941035968320439467973871628201833765744673637798566761651091542533387882073042784070071267125062277929401734691698468266444047010687272966292711762656870293609105118165482938715610938798020928682

657580956100813397917815197599012261612749693926781882433340341199
182487883155571698142346719540209683983983 77.

5. Conclusion

In this paper we have studied how CRT-RSA decryption can be made faster for efficient implementation in the environment of RFID communication security, particularly for communication between active tags and readers. For primes p, q with a run of zeros in the MSBs and decryption exponents d_p, d_q with very few ones in the bit pattern, we have achieved this efficiency. The speed is around 21% over the state-of-the-art results for 512-bit primes and 192-bit decryption exponents.

References

[1] D. Boneh, G. Durfee and Y. Frankel. Exposing an RSA Private Key Given a Small Fraction of its Bits. Asiacrypt 1998, LNCS 1514, pp. 25–34, 1998.

[2] D. Boneh and G. Durfee. Cryptanalysis of RSA with Private Key d Less Than $N^{0.292}$. IEEE Trans. on Information Theory, 46(4):1339–1349, 2000.

[3] D. Boneh. Twenty Years of Attacks on the RSA Cryptosystem. Notices of the AMS, 46(2):203–213, February, 1999.

[4] D. Boneh and H. Shacham. Fast variants of RSA. CryptoBytes, Vol. 5, No. 1, pp. 1-9, 2002.

[5] J. W. Bos, M. E. Kaihara, T. Kleinjung, A. K. Lenstra and P. L. Montgomery. On the Security of 1024-bit RSA and 160-bit Elliptic Curve Cryptography.
Cryptology ePrint Archive: Report 2009/389,
available at http://eprint.iacr.org/2009/389.

[6] D. Coppersmith. Small Solutions to Polynomial Equations and Low Exponent Vulnerabilities. Journal of Cryptology, 10(4):223–260, 1997.

[7] E. Brier, C. Clavier, J. -S. Coron and D. Naccache. Cryptanalysis of RSA Signatures with Fixed-Pattern Padding. Crypto 2001, LNCS 2139, pp. 433–439, 2001.

[8] S. Galbraith, C. Heneghan and J. Mckee. Tunable Balancing of RSA. ACISP 2005, LNCS 3574, pp. 280–292, 2005.

[9] G. H. Hardy and S. Ramanujan. The normal number of prime factors of a number n. Quart. J. Math. 48 (1917), 76-92.

[10] N. Howgrave-Graham. Finding Small Roots of Univariate Modular Equations Revisited. Proceedings of Cryptography and Coding, LNCS 1355, pp. 131–142, 1997.

[11] E. Jochemsz. Cryptanalysis of RSA Variants Using Small Roots of Polynomials. Ph. D. thesis, Technische Universiteit Eindhoven, 2007.

[12] E. Jochemsz and A. May. A strategy for finding roots of multivariate polynomials with new applications in attacking RSA variants. Asiacrypt 2006, LNCS 4284, pp. 267–282, 2006.

[13] E. Jochemsz and A. May. A Polynomial Time Attack on RSA with Private CRT-Exponents Smaller Than $N^{0.073}$. Crypto 2007, LNCS 4622, pp. 395–411, 2007.

[14] A. May. RSA & meet-in-the-middle Angriffe. Chapter from the course "Public Key Kryptanalyse", available via
http://www.informatik.tu-darmstadt.de/KP/lehre/ws0506/vl/pkk.html.

[15] A. K. Lenstra, H. W. Lenstra and L. Lovász. Factoring Polynomials with Rational Coefficients. Mathematische Annalen, 261:513–534, 1982.

[16] A. Lenstra. Generating RSA moduli with a predetermined portion. Asiacrypt 1998, LNCS 1514, pp.1–10, 1998.

[17] A. K. Lenstra and H. W. Lenstra, Jr., The Development of the Number Field Sieve, Springer-Verlag, 1993.

[18] A. May. Using LLL-Reduction for Solving RSA and Factorization Problems: A Survey. LLL+25 Conference in honour of the 25th birthday of the LLL algorithm, 2007.
Available at http://www.cits.rub.de/personen/may.html [last accessed 23 November, 2009].

[19] A. Menezes, P. Van Oorschot, and S. Vanstone. Handbook of Applied Cryptography. CRC Press,1997.
[20] P. L. Montgomery. Modular multiplication without trial division. Math. Comp. 44(1985), pp. 519–521.
[21] G. Qiao and K.-Y. Lam. RSA signature algorithm for microcontroller implementation. CARDIS 1998, LNCS 1820, pp. 353–356, 1998.
[22] M. R. Reiback, B. Crispo and A. S. Tanenbaum. The evolution of RFID security; Pervasive Computing. IEEE Volume 5, Issue 1, January-March 2006. pp 62–69.
[23] R. L. Rivest, A. Shamir and L. Adleman. A Method for Obtaining Digital Signatures and Public Key Cryptosystems. Communications of ACM, 21(2):158–164, Feb. 1978.
[24] S. Sarkar and S. Maitra. Partial Key Exposure Attack on CRT-RSA. ACNS 2009, LNCS 5536, pp. 473–484, 2009.
[25] R. D. Silverman. Fast generation of random, strong RSA primes. Cryptobytes, 3(1):9–13, 1997.
[26] D. R. Stinson. Cryptography - Theory and Practice. 2nd Edition, Chapman & Hall/CRC, 2002.
[27] M. Wiener. Cryptanalysis of Short RSA Secret Exponents. IEEE Transactions on Information Theory, 36(3):553–558, 1990.

Radio Frequency Identification System Security
Y. Li and J. Zhou (Eds.)
IOS Press, 2010

doi:10.3233/978-1-60750-485-6-73

Fingerprinting Radio Frequency Identification Tags Using Timing Characteristics

Senthilkumar CHINNAPPA GOUNDER PERIASWAMY [a,1], Dale R. THOMPSON[a], Henry P. ROMERO[b] and Jia DI[a]
[a] *University of Arkansas, Fayetteville, USA*
[b] *University of Colorado, Boulder, USA*

Abstract. Radio Frequency Identification (RFID) has been very actively developed as an identification technology in the last ten years. The uniqueness of RFID tag's electronic product code has made it to be used as an anti-counterfeiting feature for objects attached to it. However, currently the anti-counterfeiting properties of the tag themselves and methods to prevent counterfeiting of the tags have not been established. Here we propose a physical layer fingerprinting methodology that will improve the security of RFID tags.

Keywords. RFID, Security, Authentication, Anti-Counterfeiting

1. Introduction

Radio frequency identification (RFID) tags are devices that are used to uniquely identify objects that are attached to it. EPCglobal Class 1 Generation 2 is a widely used protocol for passive RFID systems that operates in the 860 - 960 MHz frequency (UHF) range [1]. An RFID reader identifies the tag by using the 96-bit electronic product code (EPC) of the tag. There is no method specified in the standard that prevents a duplicate tag from using the EPC of the original tag to authenticate itself as the original tag. The low cost and size that is driving the technology makes the implementation of conventional cryptographic security protocols a challenge.

In order to keep the cost low, the tags are produced with a high speed manufacturing process which introduces minor variations in the hardware components of the tag. We are using a measurement of those differences in the hardware to distinguish tags. These variations are difficult to reproduce, predict or control because this type of manufacturing focuses on mass-producing the tags at a lower cost. This method will provide an additional layer of security for authenticating RFID tags.

2. Background

The EPCglobal Class 1 Generation 2 is a reader-first-talk, half-duplex protocol. The process of a reader identifying a single tag in the system is illustrated in Fig. 1. The

[1] Corresponding Author.

reader starts the process of identifying a tag by sending a *select* command. The tag does not respond to the command but changes it state to respond to the following *query* command. The reader then sends the *query* command to which the tag responds with a random number. The reader asks the tag to send the identifying information by acknowledging the random number from that particular tag. When the tag receives the acknowledgment (*ack*), it sends the EPC, Protocol control (PC) and Cyclic redundancy check (CRC). If the reader receives the information without error, it acknowledges the receipt.

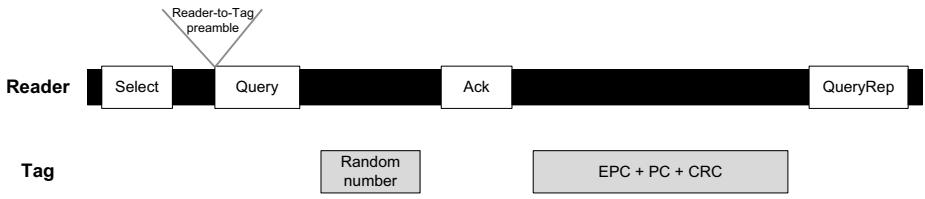


Figure 1. Reader - Tag Communication.

The reader sets the data rate of the tag-to-reader communication during the reader-to-tag preamble that precedes all *query* commands from the reader. The data rate remains the same for the entire session. The reader-to-tag preamble shown in Fig. 2 consists of a fixed-length start delimiter, a data '0' symbol, a reader-to-tag calibration symbol and a tag-to-reader calibration symbol (*TRcal*). The tag measures the length of the *TRcal* and uses it as the basis to determine the data rate of the tag-to-reader communication. The data rate for the FM0 encoding is calculated by dividing the *divide ratio* by the *TRcal*. The *divide ratio* is sent to the tag as a part of the query command based on the data rate required.

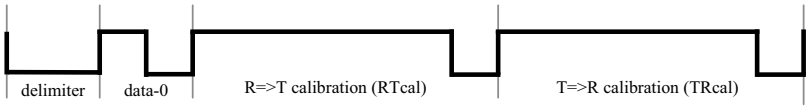


Figure 2. Reader-to-Tag preamble.

In this paper, we are using the differences in the measurement, storage, and usage of the *TRcal* value between tags to distinguish them. The differences are represented in the time required to transmit the EPC, PC and CRC.

3. Measurement

The communication signals between the RFID reader and RFID tags were captured using a Tektronix DPO70604 oscilloscope in our lab, which was a non-controlled radio frequency (RF) environment. The oscilloscope had a bandwidth of 6 GHz with a sampling rate of 25 Giga samples per second with maximum capture duration of 4 milliseconds at the highest resolution. A TagSense Micro-UHF reader was used to communicate with the tag. The reader was connected to a PC through an USB interface. This reader shown in Fig. 3 had a small form factor. A linear patch antenna

was connected to the reader to communicate with the tag. The antenna had a gain of 8 dBi.

We used this oscilloscope because the sampling rate will allow us to capture information at high resolution which was required for frequency features we were planning to analyze. However, we found that the timing feature discussed in this paper does not require such high resolution. Signals captured at a sampling rate of 25 Mega samples per second will provide enough resolution to extract the timing based features discussed in this paper.

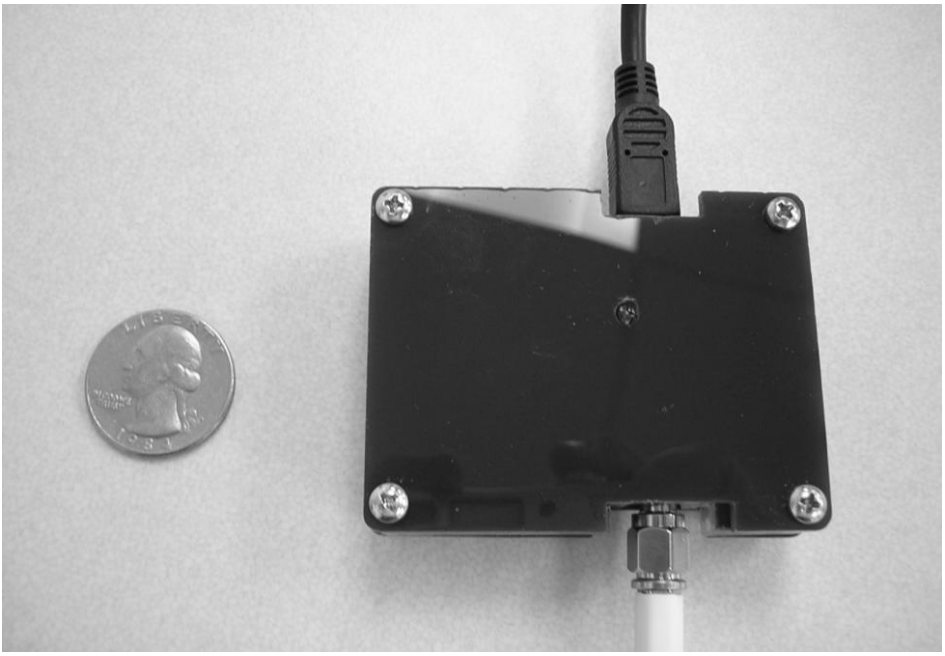


Figure 3. TagSense Micro-UHF RFID reader.

The antennas and tag were mounted to plastic stands such that they were at an elevation of 180 cm from the ground; this elevation from the ground reduced the interference caused by the reflection of RF signals. The setup had a free space of 0.5 meter around it. The frequency hopping of the reader was disabled and it was made to communicate with the tag at 915 MHz. RFID tags from three major manufacturers were used. They were entry-level passive tags with a cost of about six cents per tag. Tags from each manufacturer were taken from the same roll, which shows that they were manufactured at the same time. These models have already been deployed in both case level and item level RFID applications. We programmed all the tags with the same EPC, which was 300833b2ddd9048035050000. There was no specific reason in choosing this EPC; we just wanted all the tags to have the same EPC.

The captured communication between a reader and a single tag is shown in Fig. 3. We wanted to capture the communication between the tag and the reader such that the reader and the tag parts of the communication can be differentiated by their power level. This difference in tag to reader power was used as trigger to capture the tag communication. This enabled capturing tag communication without using a setup that

captures communication using the synchronization between the communicating reader and the capturing oscilloscope.

Each tag was measured six times initially. The tags were measured again six times after one week. The two sets were labeled *time1* and *time2* respectively. The tags were removed from the test fixture after *time1* measurements and stored. The tags were mounted again in the test fixture for the *time2* measurements. In all the measurements, the reader used double-sideband amplitude-shift keying (DSB-ASK) modulation with pulse-interval encoding (PIE) Type C encoding. The tags used ASK modulation with FM0 (bi-phase space) encoding.

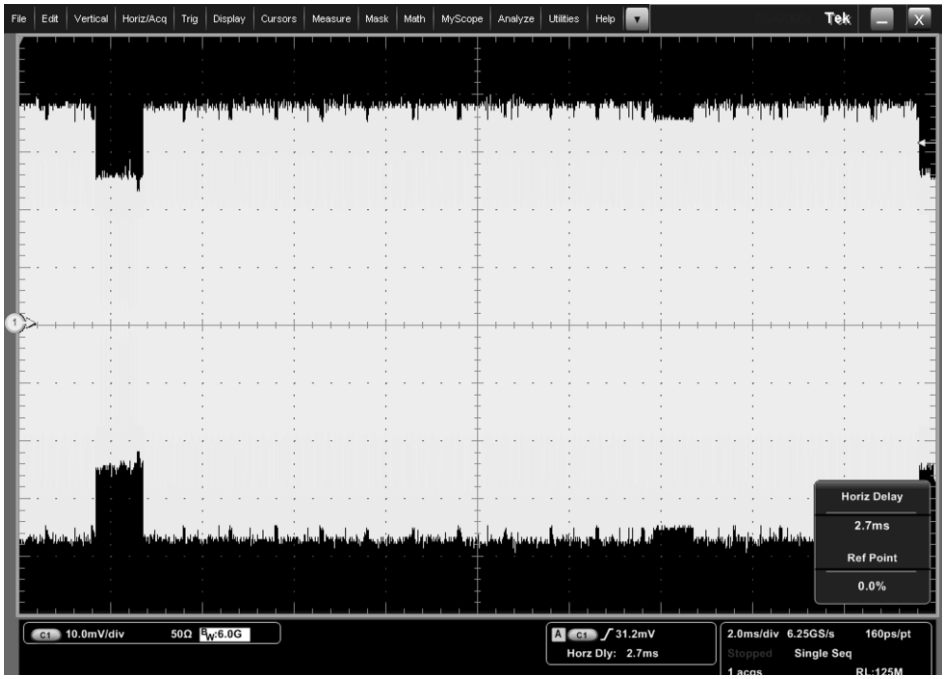


Figure 4. Reader to Tag Communication.

4. Tag Timing Response

The time required for the tag to send the EPC, PC and CRC to the reader was measured. This information was extracted from the captured signals using a MATLAB script. The script used the difference in the power level of the tag and reader communication to extract the information. The transmission time of the tags are shown in Fig. 5, Fig. 6 and Fig. 7.

K Nearest Neighbor algorithm (KNN) was used to classify the data. KNN is an instance-based classifier that classifies the current instance to the closest enrolled group by using a distance metric [2]. The classifier was trained on *time1* data which became the enrolled group. Then the classifier was used to classify the *time2* data which became the instance being classified. Tags from Manufacturer-1 were classified with a success rate of 98.44%. Manufacturer-2 tags were classified with a success rate of 96.25%. The success rate of Manufacturer-3 tags was 31.54%.

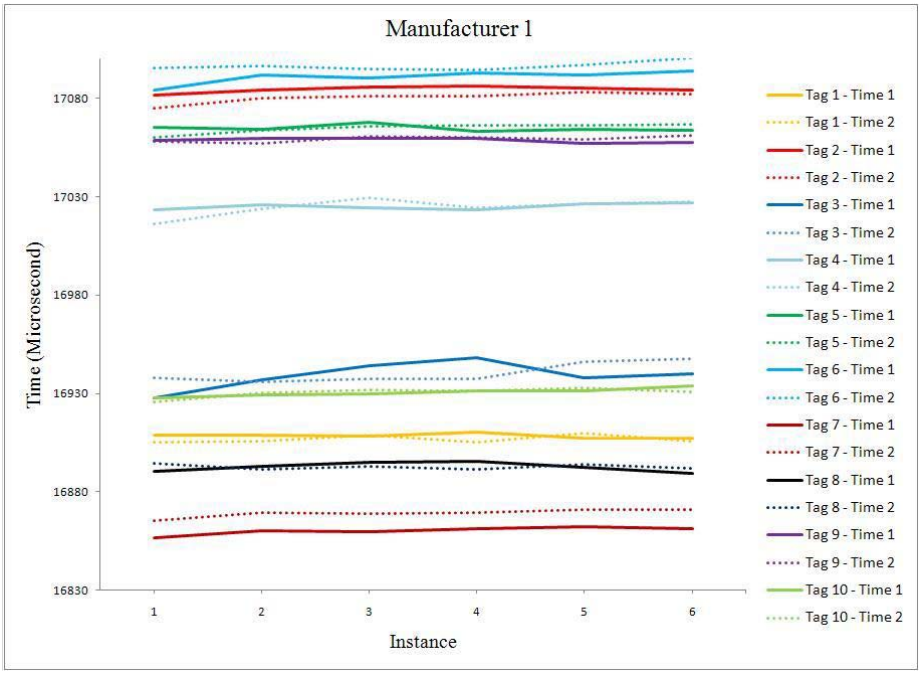


Figure 5. Tag timing response – Manufacturer 1.

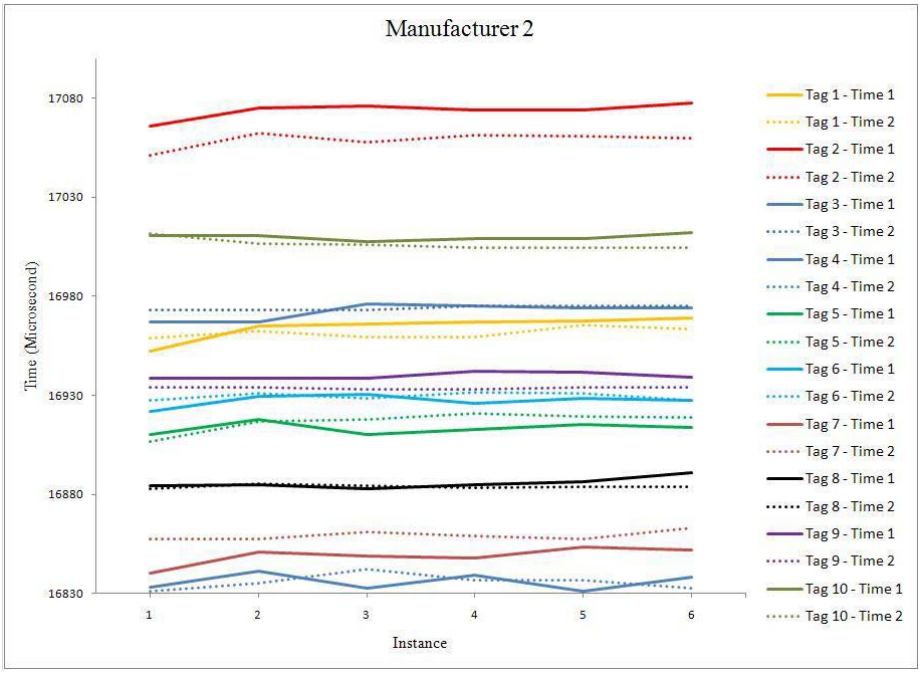


Figure 6. Tag timing response – Manufacturer 2.

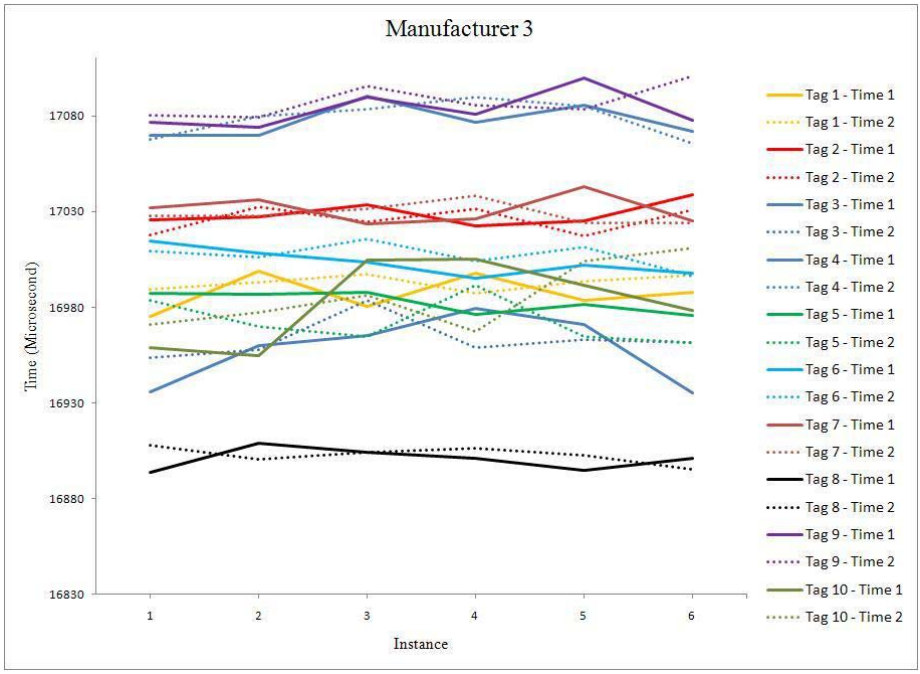


Figure 7. Tag timing response – Manufacturer 3.

Two of the tag manufacturers were repeatable and reproducible, and one manufacturer was not completely repeatable and reproducible. Investigating the behavior of manufacturer 3 tag is part of the future work. Having one manufacturer that was not repeatable leads us to believe that the final fingerprint will need multiple features, not just timing, and be ranked based on their reliability and performance.

The differences that occur during the measurement storage and the use of the *TRcal* value by the tag leads to the differences in the length of the EPC, PC and CRC transmission. It can also be observed that the reader is using the same *TRcal* and *divide ratio* by observing that the measurements are reproducible for the same tag. Therefore, it is the differences in the interpretation and implementation of the *TRcal* by the tag that produces the differences in length of the transmission.

The measurements show that the transmission time is between 16.8 milliseconds and 17.1 milliseconds which will lead to a small bandwidth and resolution when there are a larger number of tags in the system. The bandwidth and resolution can be increased when the length of the data sent from the tag is increased. Transmission of longer lengths like repeatedly sending the EPC multiple times could be done by using custom commands.

There is a possibility of adversaries building tags or circuits that can mimic the hardware feature but the feasibility of such an approach is an open research question.

5. Related Work

Several light weight cryptographic models [3-9] have been proposed to prevent unauthorized reading or cloning of tags. Although they may improve security and provide resistance to cloning after sufficient peer review [10], they face the possibilities

of attack through improper implementation, reverse-engineering [11], relay [12-14] and side-channel [15] attacks. Building a secure cryptographic protocol for a RFID tags which has only a couple of thousand gates dedicated for security is a challenge by itself, so providing an additional layer of security through hardware fingerprinting increases the reliability of the security mechanism.

Some manufacturers provide security using a unique Transponder ID (TID) for a controlled group of tags. TID is a number that is present in an Integrated Chip (IC) to identify their model and locate custom/optional commands they support. They are written in a Read Only Memory (ROM) when the tag is manufactured. Their purpose of identifying the manufacturing and specification details of the IC has been extended to identifying the RFID tag itself [16]. The TID was not created as a security feature but it is currently being used as one. This feature does not prevent an attacker from programming a non-programmed RFID tag or building a RFID tag emulator with the same TID information. Non-programmed RFID tags are currently not available in the market which may not be the case in the future [16]. Therefore, TID cannot be a reliable standalone anti-counterfeiting feature but can be used as one of the time buying and inexpensive options to prevent counterfeiting

A proposal to create a physical fingerprint of RFID tags using the initialization state of SRAM's in them was proposed in [17]. They used data from virtual tags to show that a SRAM can be used to uniquely identify tags. This research is closest to our work although it was done independently and simultaneous. The fingerprinting mechanism proposed in this paper is also based on variations that are caused due to the manufacturing process of the RFID tags. A difference between the work in [17] and our work is that their fingerprinting methodology has only been demonstrated in virtual tags and not in real tags. All of our features are measured from commonly used RFID tags without using any invasive methods.

The work in [18] investigates how RFID tags can be made unclonable by linking it to a Physical Unclonable Function (PUF). PUFs are physical structures that respond to challenges that are easy to measure but are hard to predict. In addition, PUFs are difficult to copy or clone. They are embedded in the tag such that any attempt to remove them will either destroy them or modify the values of the responses to challenges. The PUF only communicates with the chip of the tag and is inaccessible to the reader. Security protocols based on PUFs are used in [19]. PUF based security requires special circuits that needs to be built into tags when they are manufactured while our method can work on any existing tag.

Different models of high frequency (HF) RFID tags were distinguished using the differences of the waveform in [20] and [21]. The process of fingerprinting HF tag is very different from process of fingerprinting UHF tags. HF tags use inductive coupling (magnetic field) for communication while the main form of UHF tags communicate in the radiative field (electric field). In HF RFID, communication is almost instantaneous because the transmission time is a fraction of a cycle of a RF voltage. There is no separation between reader and tag communication; changes in the reader antenna induce change in the tag antenna and vice versa. This is further explained by the work done in [20] where they used changes in the reader antenna when different tags were used to fingerprint the tag. The changes in the reader were more pronounced than the changes to the tag because of the greater modulation depth. In contrast, there is distinct channel of communication between reader transmission and tag communication in UHF RFID. The longer length of the transmission path makes the signals susceptible to more loss and interference.

6. Conclusion

We have used the transmission time of EPC, PC and CRC to distinguish individual UHF passive RFID tags from two major manufacturers, with the measurements being repeatable and reproducible. This feature is appropriate to be part of a hardware-based fingerprinting system which can be used against counterfeiting of RFID tags. The hardware-based fingerprinting system when combined with other application layer security protocols will provide robust security for RFID tags. This fingerprinting method is independent of the computational capabilities and resources of the tag which enables its implementation with any tag already in the system.

All the above measurements were taken by measuring the tag by itself. There will be more variation in the hardware based electronic features when the tag is part of a product due to the dielectric, reflective and constructive properties of the product. When the tag is fingerprinted along with the product, we will create a fingerprint of the tag-product combo. This combo fingerprint can be used to prevent tag swapping on products for malicious purposes.

References

[1] *EPC Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860 MHz – 960 MHz*, ver. 1.2.0, EPCglobal Inc., Oct. 23, 2008. Available: http://www.epcglobalinc.org.

[2] D. W. Aha, D. Kibler and M. K. Albert, "Instance-Based Learning Algorithms," *Machine Learning*, vol. 6, pp. 37-66, 1991.

[3] A. Juels, "RFID security and privacy: a research survey," *Selected Areas in Communications, IEEE Journal on,* vol. 24, pp. 381-394, 2006.

[4] A. Juels, "Minimalist cryptography for low-cost RFID tags," in *International Conference on Security in Communication Networks -- SCN 2004; Lecture Notes in Computer Science,* 2004, pp. 149-164.

[5] A. Juels, "'Yoking-proofs' for RFID tags," in *International Workshop on Pervasive Computing and Communication Security -- PerSec 2004,* 2004, pp. 138-143.

[6] F. Kerschbaum and A. Sorniotti, "RFID-based supply chain partner authentication and key agreement," in *Proceedings of the Second ACM Conference on Wireless Network Security* -- WiSec'09,2009.

[7] M. Lehtonen, F. Michahelles and E. Fleisch, "How to detect cloned tags in a reliable way from incomplete RFID traces," in *IEEE International Conference on RFID -- RFID 2009*, 2009.

[8] T. Dimitriou, "A lightweight RFID protocol to protect against traceability and cloning attacks," in *First International Conference on Security and Privacy for Emerging Areas in Communication Networks -- SecureComm 2005,* Athens, Greece, September 2005.

[9] A. Juels, "Strengthening EPC tags against cloning," in *WiSe '05: Proceedings of the 4th ACM Workshop on Wireless Security*, 2005, pp. 67-76.

[10] P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estevez-Tapiador, T. Li and J. C. A. van der Lubbe, "Weaknesses in two recent lightweight RFID authentication protocols," in *Workshop on RFID Security* -- RFIDSec'09, 2009.

[11] S. Bono, M. Green, A. Stubblefield, A. Juels, A. Rubin and M. Szydlo, "Security analysis of a cryptographically-enabled RFID device," in *USENIX Security Symposium,* 2005, pp. 1-16.

[12] Z. Kfir and A. Wool, "Picking virtual pockets using relay attacks on contactless smartcard systems," in *First International Conference on Security and Privacy for Emerging Areas in Communication Networks -- SecureComm 2005,* Athens, Greece, September 2005.

[13] G. Hancke, "A Practical Relay Attack on ISO 14443 Proximity Cards," February. 2005. Available http://www.cl.cam.ac.uk/gh275/relay.pdf

[14] G. Hancke and M. Kuhn, "An RFID distance bounding protocol," in First International Conference on Security and Privacy for Emerging Areas in Communication Networks -- SecureComm 2005, Athens, Greece, September 2005.

[15] D. Carluccio, K. Lemke and C. Paar, "Electromagnetic side channel analysis of a contactless smart card: first results," in *ECRYPT Workshop on RFID and Lightweight Crypto*, Graz, Austria, July 2005, pp. 44-51.

[16] M. Lehtonen, A. Ruhanen, F. Michahelles and E. Fleisch, "Serialized TID numbers - A headache or a blessing for RFID crackers?" in *IEEE International Conference on RFID* -- RFID 2009.
[17] D. Holcom, W. Burleson and K. Fu, "Initial SRAM state as a fingerprint and source of true random numbers for RFID tags," in *Workshop on RFID Security -- RFIDSec'07,* 2007.
[18] P. Tuyls and L. Batina, "RFID-tags for anti-counterfeiting," in Topics in Cryptology - CT-RSA 2006, the Cryptographers' Track at the RSA Conference 2006; Lecture Notes in Computer Science, 2006.
[19] S. Devadas, E. Suh, S. Paral, R. Sowell, T. Ziola and V. Khandelwal, "Design and Implementation of PUF-Based Unclonable RFID ICs for Anti-Counterfeiting and Security Applications," *IEEE International Conference on RFID*, pp. 58-64.
[20] H. P. Romero, K. A. Remley, D. F. Williams and Chih-Ming Wang, "Electromagnetic Measurements for Counterfeit Detection of Radio Frequency Identification Cards," *Microwave Theory and Techniques, IEEE Transactions on*, vol. 57, pp. 1383-1387.
[21] B. Danev, T. S. Heydt-Benjamin and S. Capkun, "Physical-layer identification of RFID devices," in *Proceedings of the 18th USENIX Security Symposium* USENIX'09, 2009.

This page intentionally left blank

Radio Frequency Identification System Security
Y. Li and J. Zhou (Eds.)
IOS Press, 2010

doi:10.3233/978-1-60750-485-6-83

Security Flaws in a Recent Ultralightweight RFID Protocol

Pedro Peris-Lopez [a,1], Julio C. Hernandez-Castro [b]
Juan M. E. Tapiador [c] and Jan C.A. van der Lubbe [a]

[a] *ICT Group, Technical University of Delft, The Netherlands*
[b] *School of Computing, University of Portsmouth, United Kingdom*
[c] *Department of Computer Science, University of York, United Kingdom*

Abstract. In 2006, Peris-Lopez *et al.* [1,2,3] initiated the design of ultralightweight RFID protocols – with the UMAP family – involving only simple bitwise logical or arithmetic operations such as bitwise XOR, OR, AND, and addition. This combination of operations was revealed later to be insufficient for the intended security level [12,13]. Then, Chien proposed the SASI protocol [4] with the aim of offering better security by adding the bitwise rotation to the set of supported operations. The SASI protocol represented a milestone in the design of ultralightweight protocols, although certain attacks have been published against this scheme [5,6,7]. In 2008, a new protocol named Gossamer [8] was proposed and the scheme can be considered a further development of both the UMAP family and SASI. Although no attacks have been disclosed against Gossamer, Lee *et al.* [9] have recently published an alternative scheme that is highly reminiscent of SASI. In this paper, we show that Lee's scheme fails short of many of its security objectives, being vulnerable to several important attacks like traceability, full disclosure, cloning and desynchronization.

Keywords. RFID, authentication, ultralightweight protocols, cryptanalysis

1. Introduction

In an RFID system, objects are labelled with a tag. Each tag contains a microchip with a certain (generally limited) amount of computational and storage capabilities, and a coupling element. Such devices can be classified according to their memory type and power source. Another relevant parameter is tag price, which creates a broad distinction between high-cost and low-cost RFID tags. The *rule of thumb* of gate cost says that every extra 1,000 gates increases chip price by 1 cent [10].

In [4], Chien proposed a tag classification mainly based on which are the operations supported on-chip. High-cost tags are divided into two classes: "full-fledged" and "simple". Full-fledged tags support on-board conventional cryptography like symmetric encryption, cryptographic one-way functions and even public key cryptography. Simple

[1]Corresponding Author: Delft University of Technology (TU-Delft), Faculty of Electrical Engineering, Mathematics, and Computer Science (EEMCS), Information and Communication Theory group (ICT). P.O. Box 5031, 2600 GA, Delft, The Netherlands; E-mail: P.PerisLopez@tudelft.nl.

tags can support random number generators and one-way hash functions. Likewise, there are two classes for low-cost RFID tags."Lightweight" tags are those whose chip supports a random number generator and simple functions like a Cyclic Redundancy Checksum (CRC), but not cryptographic hash functions. "Ultralightweight" tags can only compute simple bitwise operations like XOR, AND, OR, etc.

In this paper we focus in the latter category of ultralightweight tags. These tags represent the greatest challenge in terms of security, due to their expected wide deployment and, at the same time, extremely limited capabilities.

2. Related Work

In 2006, Peris-Lopez *et al.* proposed a family of Ultralightweight Mutual Authentication Protocols (henceforth referred to as the UMAP family). Chronologically, M^2AP [1] was the first proposal, followed by EMAP [2] and LMAP [3]. Although some vulnerabilities were discovered (active attacks [11,12], and later on passive attacks [13,14]) which rendered those first proposals insecure, they were an interesting advance in the field of lightweight cryptography for low-cost RFID tags.

In 2007, Hung-Yu Chien published a striking ultralightweight authentication protocol providing Strong Authentication and Strong Integrity (SASI) for very low-cost RFID tags [4]. The SASI protocol is highly reminiscent of the UMAP family, and more concretely, of the LMAP protocol. The main difference between these two protocols is the inclusion of rotation in the set of operations supported by each tag. Indeed, the messages transmitted over the insecure channel in the UMAP family are computed by the composition of triangular-functions (e.g. addition modulo 2, bitwise OR, AND, etc.) – easily implemented in hardware – which finally results in another triangular-function [15]. A triangular-function has the property that output bits only depend of the leftmost input bits, instead of all input bits. This undesirable characteristic (lack of diffusion) greatly facilitated the analysis of the messages transmitted by the UMAP protocols, and thus the work of the cryptanalyst.

SASI represented a considerable advance towards the design of a secure ultralightweight protocol. However, certain important attacks have been published. First, Sun *et al.* proposed two desynchronization attacks. In [6], it was proposed a denial-of-service and traceability attack. Then, D'Arco *et al.* [7] proposed another desynchronization attack and an identity disclosure attack. In [16], Phan shows how a passive attacker can track tags, violating the location privacy of tags' holder. Finally, Hernandez-Castro *et al.* [17] recently proposed a full disclosure attack, but the authors assume modular rotations instead of SASI's hamming weight rotation.

In 2008, the Gossamer protocol [8] was proposed as a further development upon both the UMAP family and the SASI protocol. So far, this scheme seems the most secure ultralightweight authentication protocol for low-cost RFID tags availiable, as no attacks have been published – to the best of our knowledge. As an alternative to Gossamer, Lee *et al.* recently published a new ultralightweight RFID protocol with mutual authentication (UMA-RFID in the following) [9]. The analysis of this recent protocol is the subject of this paper.

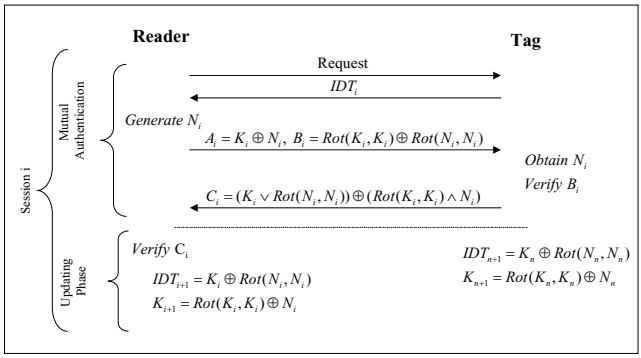


Figure 1. Ultralightweight RFID protocol with mutual authentication

3. Lee et al.'s Ultralightweight RFID Protocol with Mutual Authentication

Tag, reader and back-end database are the three entities involved in the protocol. Each tag has a static identifier (ID). A pseudonym – dynamic temporary identifier – (IDT) and a secret key (K) are shared between the tag and the reader. Indeed, the old and the potential new values of the pair $\{IDT, K\}$ are both kept in the tag to hinder desynchronization attacks. The length of the variables is 128 bits. The channel between the tag and the reader is insecure due to the open nature of the radio channel. In contrast, a secure channel is assumed for the communications between the reader and the back-end database.

Tags are limited to bitwise operations (i.e. bitwise XOR, OR and AND) and left bitwise rotation. Specifically, $Rot(A, B)$ symbolizes that the vector A is subjected to a left circular shift of n bit positions, where n is the hamming weight of vector B (i.e. $n = hw(B)$). Readers are limited to the same set of operations and have the extra capability of random number generation.

We described the messages exchanged in the protocol below (see also Figure 1). First, the reader ($\mathcal{R}$) and the tag ($\mathcal{T}$) are mutually authenticated (authentication phase). Then, the reader and the tag, respectively, update their shared private information $\{IDT, K\}$ (updating phase).

1. Authentication Phase

$\mathcal{T} \rightarrow \mathcal{R} : IDT_i$ In the session i-th, the reader sends a request message to the tag. Then, the tag backscatters its pseudonym (IDT_i) to provide anonymous identification.

$\mathcal{R} \rightarrow \mathcal{T} : A_i, B_i$ Upon receiving IDT_i, the reader looks up in the database the secret key associated to $\mathcal{T}$. Then, it generates a new random value N_i and computes the authentication messages A_i and B_i:

$$A_i = K_i \oplus N_i \tag{1}$$

$$B_i = Rot(K_i, K_i) \oplus Rot(N_i, N_i) \tag{2}$$

The reader sends $\{A_i, B_i\}$ to the tag.

$\mathcal{T} \rightarrow \mathcal{R} : C_i$ After receiving $\{A_i, B_i\}$, the tag obtains N_i' from message A_i ($N_i' = A_i \oplus K_i$) and computes its local version of B_i ($B_i' = Rot(K_i, K_i) \oplus Rot(N_i, N_i)$). If

$B_i = B_i'$, the reader is authenticated. Then, the tag computes the authentication message C_i:

$$C_i = (K_i \vee Rot(N_i, N_i)) \oplus ((Rot(K_i, K_i) \wedge N_i) \quad (3)$$

Finally, the tag sends C_i to the reader.

$\mathcal{R}$**:** Upon receiving C_i, the reader checks its correctness to authenticate the tag.

2. Updating Phase Upon the reader authentication (messages A_i, B_i), the tag updates its secret information when message C_i is sent. The updating in the reader is conditioned to the valid authentication of the tag (message C_i). Specifically, the updating phase is defined by the equations below:

$$IDT_{i+1} = K_i \oplus Rot(N_i, N_i) \quad (4)$$

$$K_{i+1} = Rot(K_i, K_i) \oplus N_i \quad (5)$$

4. Security Analysis

In this section, we show how Lee *et al.* scheme does not fulfill many of the security properties claimed in its protocol definition.

4.1. Traceability Attack

Traceability is one of the most important security threats linked to RFID technology. Location privacy is compromised when tags answer readers queries with a static value, something that, despite its well-known security shortcomings, curiously happens in numerous commercial tags. An encrypted version of the static identifier may be used for privacy protection, but an attacker could still track the tag's holder as the tag keeps on sending a constant value. So it seems necessary to anonymize tags' answers by the inclusion of nonces. However, the simple use of random numbers by itself does not guarantee that a protocol will be resistant to traceability attacks [18].

The traceability problem has attracted a lot of research. In [19], Juels and Weis give a formal definition of traceability for basic analysis of RFID systems. The same definition, though with a style more similar to that used for security protocols, is introduced by Phan in his attack against the SASI protocol [16]. The latter is used to analyze Lee *et al.*''s protocol.

In RFID schemes, tags ($\mathcal{T}$) and readers ($\mathcal{R}$) interact in protocol sessions. In general terms, the adversary ($\mathcal{A}$) controls the communications between all the participants and interacts passively or actively with them. Specifically, $\mathcal{A}$ can run the following queries:

- $\mathrm{Execute}(\mathcal{R}, \mathcal{T}, i)$ query. This models a passive attacker. $\mathcal{A}$ eavesdrops on the channel, and gets read access to the exchange of messages between $\mathcal{R}$ and $\mathcal{T}$ in session i of a genuine protocol execution.
- $\mathrm{Send}(\mathcal{X}, \mathcal{Y}, M, i)$ query. This models that the message M sends from $\mathcal{X}$ to $\mathcal{Y}$ in session i is blocked or altered (e.g. flipping one bit), preventing its correct reception.

- Test(i, $\mathcal{T}_0$, $\mathcal{T}_1$) query. This does not model any ability of $\mathcal{A}$, but it is necessary to define the untraceability test. When this query is invoked for session i, a random bit is generated $b \in \{0,1\}$. Then, the pseudonym $IDT_i^{\mathcal{T}_i}$ from the set $\{IDT_i^{\mathcal{T}_0}, IDT_i^{\mathcal{T}_1}\}$ and corresponding to tags $\{\mathcal{T}_0, \mathcal{T}_1\}$ is given to $\mathcal{A}$.

Upon definition of the adversary's abilities, the untraceability problem can be defined as a game $\mathcal{G}$ divided into the following phases:

Phase 1 (Learning): $\mathcal{A}$ can send Execute and Send queries. So, $\mathcal{A}$ eavesdrops messages – passive attack – passed over the channel and have the ability of blocking or altering – active attack – certain messages.

Phase 2 (Challenge): $\mathcal{A}$ chooses two fresh tags whose associated identifiers are $ID^{\mathcal{T}_0}$ and $ID^{\mathcal{T}_1}$. Then he sends Test(i, $\mathcal{T}_0$, $\mathcal{T}_1$) query. As result, $\mathcal{A}$ is given a dynamic temporary identifier $IDT_i^{\mathcal{T}_i}$ from the set $\{IDT_i^{\mathcal{T}_0}, IDT_i^{\mathcal{T}_1}\}$, which depends on a chosen random bit $b \in \{0,1\}$.

Phase 3 (Guessing) $\mathcal{A}$ finishes the game and outputs a bit d ($d \in \{0,1\}$) as its conjecture of the value of b.

$\mathcal{A}$'s success in winning $\mathcal{G}$ is equivalent to the success of breaking the untraceability property offered by the protocol. So the advantage of $\mathcal{A}$ in distinguishing whether the messages correspond to $\mathcal{T}_0$ or $\mathcal{T}_1$, is defined as below:

$$Adv_{\mathcal{A}}^{UNT}(t, r_1, r_2) = |Pr[d = b] - \frac{1}{2}| \tag{6}$$

where t is a security parameter (i.e. the bit length of the key shared by the tag and the reader) and r_1 and r_2 are the number of times $\mathcal{A}$ can run Execute and Send queries respectively.

Definition An RFID protocol in an RFID system (S= $\{R_i, \mathcal{T}_0, \mathcal{T}_1, ...\}$) in which an adversary $\mathcal{A}$ can invoke {Execute($\mathcal{R}$, $\mathcal{T}$, i), Send($\mathcal{X}$, $\mathcal{Y}$, M,i), Test(i, $\mathcal{T}_0$, $\mathcal{T}_1$)} in a game $\mathcal{G}$, offers resistance against traceability if:

$$Adv_{\mathcal{A}}^{UNT}(t, r_1, r_2) < \varepsilon(t, r_1, r_2) \tag{7}$$

$\varepsilon(.)$ being some negligible function.

We will show how the UMA-RFID scheme does not guarantee privacy location, thus allowing tags tracking.

Theorem 1 *The UMA-RFID protocol, on an RFID system (S= {R_i, $\mathcal{T}_0$, $\mathcal{T}_1$, ...}) in which an adversary $\mathcal{A}$ can invoke one* Execute*($\mathcal{R}$, $\mathcal{T}$, i), one* Send*($\mathcal{X}$, $\mathcal{Y}$, M,i) query, and one* Test*(i, $\mathcal{T}_0$, $\mathcal{T}_1$) query in the untraceability game $\mathcal{G}$, is vulnerable to traceability attacks, since the advantage for an adversary to win $\mathcal{G}$ is significant (in fact, maximal):* $Adv_{\mathcal{A}}^{UNT}(t, 1, 1) = 0.5 \gg \varepsilon(t, 1, 1)$.

Proof Specifically, an adversary $\mathcal{A}$ performs the following steps:

Phase 1 (Learning): $\mathcal{A}$ sends an Execute($\mathcal{R}$, $\mathcal{T}_0$, n) and a Send($\mathcal{R}$, $\mathcal{T}_0$, $A_n^{\mathcal{T}_0}/B_n^{\mathcal{T}_0}$, n) query. So $\mathcal{A}$ acquires the pseudonym $X = IDT_n^{\mathcal{T}_0}$ and prevents that $\mathcal{T}_0$ updates its internal values $\{ID^{\mathcal{T}_0}, K^{\mathcal{T}_0}\}$ because of the incorrect $A_n^{\mathcal{T}_0}$ or $B_n^{\mathcal{T}_0}$ values received.

Phase 2 (Challenge): $\mathcal{A}$ chooses two fresh tags whose associated identifiers are $ID^{\mathcal{T}_0}$ and $ID^{\mathcal{T}_1}$. Then he sends a $\mathrm{Test}(n+1, \mathcal{T}_0, \mathcal{T}_1)$ query. As result, $\mathcal{A}$ is given a dynamic temporary identifier $Y = IDT_{n+1}^{\mathcal{T}_i}$ from the set $\{IDT_{n+1}^{\mathcal{T}_0}, IDT_{n+1}^{\mathcal{T}_1}\}$, which depends on a chosen random bit $b \in \{0,1\}$.

Phase 3 (Guessing) $\mathcal{A}$ finishes $\mathcal{G}$ and outputs a bit d ($d \in \{0,1\}$) as its conjecture of the value b. In particular, $\mathcal{A}$ utilizes the following simple decision rule:

$$d = \begin{cases} \text{if } X = Y & d = 0 \\ \text{if } X \neq Y & d = 1 \end{cases} \tag{8}$$

So the adversary can associate tags's answers with its holder, with a 100% probability of success. Basically, we exploit the possibility of identifying a tag using its old pseudonym. The attack just described, is completely feasible because in the protocol definition the authors do not specify how many times a tag can be identified by using its old pseudonym. In fact, this and similar weaknesses plague the majority of RFID protocols that include an updating phase. A threshold value that guarantees the proper operation of the protocol while avoiding attacks to user's privacy location should be more carefully defined to thwart this security risk.

4.2. Full Disclosure, Cloning, and Desynchronization Attacks

The tag and the reader share a secret key. The main purpose of this key is to serve in the authentication of both entities. The key is combined with a random number to hamper its acquisition by the attacker when passed over the insecure channel. The above idea is well conceived but the protocol somehow abuses of the values $Rot(K_i, K_i)$ and $Rot(N_i, N_i)$. Indeed, this fact facilitates a sort of linear cryptanalysis of the scheme, despite the combination of triangular and non-triangular functions.

Theorem 2 *In the UMA-RFID protocol, a passive attacker, after eavesdropping two consecutive authentication sessions $\{n, n+1\}$ between an authentic tag ($\mathcal{T}$) and a legitimate reader ($\mathcal{R}$), can discover the secret key shared by these two entities by simply computing an XOR among some of the public messages transmitted over the radio channel:*

$$K_{n+1} = A_n \oplus B_n \oplus IDT_{n+1} \tag{9}$$

Proof We start describing the messages exchanged in sessions $\{n, n+1\}$:

Session n: $\{IDT_n, A_n, B_n, C_n\}$ where

$$A_n = K_n \oplus N_n \tag{10}$$

$$B_n = Rot(K_n, K_n) \oplus Rot(N_n, N_n) \tag{11}$$

Session n + 1: $\{IDT_{n+1}, A_{n+1}, B_{n+1}, C_{n+1}\}$ where

$$IDT_{n+1} = K_n \oplus Rot(N_n, N_n) \tag{12}$$

$$A_{n+1} = K_{n+1} \oplus N_{n+1} \tag{13}$$

$$B_{n+1} = Rot(K_{n+1}, K_{n+1}) \oplus Rot(N_{n+1}, N_{n+1}) \tag{14}$$

$$C_{n+1} = (K_{n+1} \vee Rot(N_{n+1}, N_{n+1})) \oplus (Rot(K_{n+1}, K_{n+1}) \wedge N_{n+1}) \tag{15}$$

The secret key of the tag in session $n+1$ is described by the equation below:

$$K_{n+1} = Rot(K_n, K_n) \oplus N_n \tag{16}$$

Finally, the attacker can acquire the actual secret key (K_{n+1}) of the tag by computing the XOR between the public messages A_n, B_n and IDT_{n+1} (see Equations (10), (11) and (12)):

$$\begin{aligned} & A_n \oplus B_n \oplus IDT_{n+1} = \\ &= K_n \oplus N_n \oplus Rot(K_n, K_n) \oplus Rot(N_n, N_n) \oplus K_n \oplus Rot(N_n, N_n) \\ &= (K_n \oplus K_n) \oplus N_n \oplus Rot(K_n, K_n) \oplus (Rot(N_n, N_n) \oplus Rot(N_n, N_n)) \\ &= (0x0) \oplus N_n \oplus Rot(K_n, K_n) \oplus (0x0) \\ &= N_n \oplus Rot(K_n, K_n) = Rot(K_n, K_n) \oplus N_n = K_{n+1} \end{aligned} \tag{17}$$

∎

RFID tags are usually not designed to be tamper resistant, because this would significantly increase their price. An active attacker may tamper with the tag in order to read from or write to its memory, in which secret values are stored. Low-cost RFID tags cannot offer protection against these sort of attacks but should be resistant, at the very least, to passive attacks. We show now how in the analyzed protocol, a passive attacker is able to clone a tag after accessing all secrets stored in its memory, but without requiring any physical manipulation.

Theorem 3 *In the UMA-RFID protocol, a passive attacker, after eavesdropping two consecutive authentication sessions $\{n, n+1\}$ between an authentic tag ($\mathcal{T}$) and a legitimate reader ($\mathcal{R}$), can clone the tag by computing:*

$$IDT_{n+2} = K_{n+1} \oplus Rot(N_{n+1}, N_{n+1}) \tag{18}$$

$$K_{n+2} = Rot(K_{n+1}, K_{n+1}) \oplus N_{n+1} \tag{19}$$

Proof From Theorem 2, an adversary can discover the actual secret key of the tag (K_{n+1}) after eavesdropping messages $\{IDT_n, A_n, B_n, C_n\}$ exchanged in session n and the dynamic temporary identifier $\{IDT_{n+1}\}$ in session $n+1$.

$$K'_{n+1} = A_n \oplus B_n \oplus IDT_{n+1} \tag{20}$$

Then the adversary can obtain the random number associated to the session $n+1$ by computing an XOR between the message A_{n+1} and the key K_{n+1}. Then, message B_{n+1} can be used to check its correctness.

$$N'_{n+1} = K'_{n+1} \oplus A_{n+1} \tag{21}$$

$$B_{n+1} \stackrel{?}{=} Rot(K'_{n+1}, K'_{n+1}) \oplus Rot(N'_{n+1}, N'_{n+1}) \tag{22}$$

Once the actual key (K_{n+1}) and the random number (N_{n+1}) linked to session $n+1$ are known by the attacker, the new state can be computed by using these values:

$$IDT_{n+2} = K'_{n+1} \oplus Rot(N'_{n+1}, N'_{n+1}) \tag{23}$$

$$K_{n+2} = Rot(K'_{n+1}, K'_{n+1}) \oplus N'_{n+1} \tag{24}$$

Finally, the attacker can copy the above values to the memory of a blank tag, which results in a successful cloning attack (having an undistinguishable copy of an authentic tag). ∎

Tags and readers have to remain in a permanent synchronization state. The authors of the protocol took the precaution of storing the old and potential new values of the pair $\{IDT, K\}$ to fight against desynchronization attacks, but in this case this well-known approach, common in the literature, is not enough. Despite of this countermeasure, an attacker is able to desynchronize a tag and a reader exploiting Theorem 2.

Theorem 4 *In the UMA-RFID protocol, a passive attacker, after eavesdropping two consecutive authentication sessions $\{n, n+1\}$ and performing a man-in-the-middle attack between an authentic tag ($\mathcal{T}$) and a legitimate reader ($\mathcal{R}$), can desynchronize these two entities by sending:*

$$A_{n+1} = K_{n+1} \oplus N^*_{n+1} \tag{25}$$

$$B_{n+1} = Rot(K_{n+1}, K_{n+1}) \oplus Rot(N^*_{n+1}, N^*_{n+1}) \tag{26}$$

$$C_{n+1} = (K_{n+1} \vee Rot(N_{n+1}, N_{n+1})) \oplus (Rot(K_{n+1}, K_{n+1}) \wedge N_{n+1}) \tag{27}$$

Proof Taking advantage of Theorem 2 any adversary, after eavesdropping messages $\{IDT_n, A_n, B_n, C_n\}$ exchanged in session n, and the dynamic temporary identifier $\{IDT_{n+1}\}$ of session $n+1$, gets the actual secret key of the tag (K_{n+1}).

$$K'_{n+1} = A_n \oplus B_n \oplus IDT_{n+1} \tag{28}$$

Then, the attacker starts the man-in-the-middle attack. Specifically, the attacker intercepts messages $\{A_{n+1}, B_{n+1}\}$ (see Equations (13) and (14)) and sends $\{A^*_{n+1}, B^*_{n+1}\}$ linked to the random number N^*_{i+1}:

$$A^*_{n+1} = K'_{n+1} \oplus N^*_{n+1} \tag{29}$$

$$B^*_{n+1} = Rot(K'_{n+1}, K'_{n+1}) \oplus Rot(N^*_{n+1}, N^*_{n+1}) \tag{30}$$

Finally, the attacker intercepts the answer C^*_{n+1} of the tag, and computes the answer C'_{n+1} to the original messages $\{A_{n+1}, B_{n+1}\}$ sent by the legitimate reader:

$$N'_{n+1} = K'_{n+1} \oplus A_{n+1} \tag{31}$$

$$B_{n+1} \stackrel{?}{=} Rot(K'_{n+1}, K'_{n+1}) \oplus Rot(N'_{n+1}, N'_{n+1}) \tag{32}$$

$$C'_{n+1} = (K'_{n+1} \vee Rot(N'_{n+1}, N'_{n+1})) \oplus (Rot(K'_{n+1}, K'_{n+1}) \wedge N'_{n+1}) \tag{33}$$

After the mutual authentication between the tag and the reader, both entities update their internal secret values:

Tag	Reader
$IDTN^*_{n+2} = K'_{n+1} \oplus Rot(N^*_{n+1}, N^*_{n+1})$	$IDT'_{n+2} = K'_{n+1} \oplus Rot(N'_{n+1}, N'_{n+1})$
$K^*_{n+2} = Rot(K'_{n+1}, K'_{n+1}) \oplus N^*_{n+1}$	$K'_{n+2} = Rot(K'_{n+1}, K'_{n+1}) \oplus N'_{n+1}$

So the adversary deceives the tag and the reader into thinking that the random number associated to the session $n+1$ is N^*_{n+1} or N'_{n+1}, respectively. Consequently, the tag a the reader lose their synchronization after the completion of the updating phase. ∎

To further clarify the attacks previously described, Figures 2(a) and 2(b) illustrate the exchanged messages.

As an alternative to the last presented attack, an adversary can desynchronize tags and readers using the non-resistance of bitwise operations to active attacks [20]. The adversary can reuse old values, transmitted in the channel, to compute new valid authentication messages. Specifically, an XOR operation between the captured value and a constant value properly selected (e.g. $A_{i+1} = A_i \oplus 0x0005$) is enough to achieve this objective.

Theorem 5 *In the UMA-RFID protocol, a passive attacker, after eavesdropping an authentication session n between an authentic tag ($\mathcal{T}$) and a legitimate reader ($\mathcal{R}$), can desynchronize these two entities by sending: $A_{n+1} = A_n \oplus C_1$, $B_{n+1} = B_n \oplus C_2$, where $\{C_i\}_{i=1}^2$ are constant values whose hamming weight is exactly 2.*

Proof First, the reader eavesdrops messages $\{IDT_n, A_n, B_n, C_n\}$ passed over the channel in session n, where

$$A_n = K_n \oplus N_n \tag{34}$$

$$B_n = Rot(K_n, K_n) \oplus Rot(N_n, N_n) \tag{35}$$

After the mutual authentication, the tag and the reader update their secret values $\{IDT_{n+1}, K_{n+1}\}$. Indeed the tag stores the old and the potential new values with the aim of preventing desynchronization attacks. However, the adversary may exploit this fact – simulating the incorrect reception of C message and thus using the old values in a new authentication – provoking a new updating in the tag but not in the reader. Specifically, the adversary follows the experiment described below:

1. Initialization. The adversary randomly selects a C_1 value, with the restriction that its hamming weigh is 2 (i.e. $hw(C_1) = 2$).

2.0. Selection of the mask. The adversary picks up a C_2 value from the subset of $x \in \{0, 1, ..., 2^L\}$ that satisfies $hw(x) = 2$, where L is the length of the variables used (i.e. $L = 128$ in Lee *et al.* protocol [9]).

2.1 Authentication. The adversary computes and sends to the legitimate tag the authentication messages:

$$A_{n+1} = A_n \oplus C_1 = K_n \oplus N_n \oplus C_1 \tag{36}$$

$$B_{n+1} = B_n \oplus C_2 = Rot(K_n, K_n) \oplus Rot(N_n, N_n) \oplus C_2 \tag{37}$$

2.2 Check of $\mathbf{C_2}$. If the tag accepts $\{A_{n+1}, B_{n+1}\}$ and replies $\{C_{n+1}\}$ to the adversary, it proves the success of the attack launched. Otherwise, the process is repeated from Step 2.0.

Table 1. Performance comparison of ultralightweight authentication protocols

	UMAP family [1,2,3]	SASI [4]	UMA-RFID [9]	Gossamer [8]
Resistance to desynchronization attacks	No	No	No	Yes
Resistance to disclosure attacks	No	No	No	Yes
Privacy and anonymity	No	No	No	Yes
Mutual auth. and forward security	Yes	Yes	Yes	Yes
Total messages for mutual auth.	4-5L	4L	3L	4L
Memory size on tag	6L	7L	5L	7L
Memory size for each tag on database	6L	4L	3L	4L
Operation types on tag	$\oplus$, $\vee$, $\wedge$, +	$\oplus$, $\vee$, $\wedge$, +, Rot	$\wedge$, $\vee$, $\oplus$, Rot	$\oplus$, +, Rot, $MixBits$

3. Check of $\mathbf{C_1}$. If Step 2 (2.0 – 2.2) completely fails, the process is repeated from Step 1.

When messages $\{A_{n+1}, B_{n+1}\}$ are accepted by the legitimate tag, the tag sends C_{n+1} and inmediately updates its secret values. However, the reader, which is unaware of the attack committed, keeps on storing its old values. So the reader and the tag lose their synchronization, and this situation is irreversible.

The remaining question is to know how efficient the attack is. C_1 is restricted to having a hamming weigh of 2 to make the hamming weigh of N_n and $N_n \oplus C_1$ unknown equal with a relatively high probability. As two bits are flipped in N_n, and N_n is an uniformly distributed random vector, the above condition is satisfied with a probability of $1/2$. Finally, the adversary has to test with different values of C_2. As the adversary does not know the hamming weight of $N_n \oplus C_1$, he can not say how many bits C_1 is rotated. However, he knows that the vector resulting from this rotation has a hamming weight of 2, which is quite advantageous. Indeed, the average number of times that the adversary has to try is $C_{L,2} = \binom{L}{2} = \binom{128}{2} = 8128 \ll 2^{128}$. ∎

Finally, a simple comparison of ultralightweight authentication protocols is shown in Table 1, where L designates the bit length of the variables used.

5. Conclusions

In this paper, we present the cryptanalysis of Lee *et al.* protocol, which is one of the most recent RFID mutual authentication protocols in the area of ultralightweight cryptography. The scheme presents noteworthy weaknesses related to most of the security properties initially required in its protocol design. Furthermore, the protocol is an excellent example of the fact that both triangular and non-triangular functions have to be combined to design secure ultralightweight protocols, but also that their combined usage, just by itself, does not guarantee any security at all.

References

[1] P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estevez-Tapiador, and A. Ribagorda. M2AP: A minimalist mutual-authentication protocol for low-cost RFID tags. In *Proc. of UIC'06*, volume 4159 of *LNCS*, pages 912–923. Springer-Verlag, 2006.

[2] P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estevez-Tapiador, and A. Ribagorda. LMAP: A real lightweight mutual authentication protocol for low-cost RFID tags. In *Hand. of Workshop on RFID and Lightweight Crypto*, 2006.

[3] P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estevez-Tapiador, and A. Ribagorda. EMAP: An efficient mutual authentication protocol for low-cost RFID tags. In *Proc. of IS'06*, volume 4277 of *LNCS*, pages 352–361. Springer-Verlag, 2006.

[4] H.-Y. Chien. "SASI: A new ultralightweight RFID authentication protocol providing strong authentication and strong integrity". *IEEE Transactions on Dependable and Secure Computing* 4(4):337–340. Oct.-Dec. 2007.

[5] H.-M. Sun, W.-C. Ting, and K.-H. Wang. "On the Security of Chien's Ultralightweight RFID Authentication Protocol". In *Cryptology ePrint Archive*. `http://eprint.iacr.org/2008/083`, 2008.

[6] T. Cao, E. Bertino, and H. Lei. "Security Analysis of the SASI Protocol". *IEEE Transactions on Dependable and Secure Computing* 6(1):73–77. Jan.-Mar. 2009.

[7] P. D'Arco and A. De Santis. "From Weaknesses to Secret Disclosure in a Recent Ultra-Lightweight RFID Authentication Protocol". In *Cryptology ePrint Archive*. `http://eprint.iacr.org/2008/470`, 2008.

[8] P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estevez-Tapiador, and A. Ribagorda. Advances in Ultralightweight Cryptography for Low-cost RFID Tags: Gossamer Protocol. In *Proc. of WISA'08*, Volume 5379 of *LNCS*, pages 56-68. Springer-Verlag, 2008.

[9] Y.-C. Lee, Y.-C. Hsieh, P.-S. You, T.-C. Chen A New Ultralightweight RFID Protocol with Mutual Authentication, In *Proc. of WASE'09*, Volume 2 of *ICIE*, pages 58-61, 2009.

[10] S. Weis. Security and Privacy in Radio-Frequency Identification Devices. In *Master Thesis, MIT*, 2003.

[11] T. Li and G. Wang. Security analysis of two ultra-lightweight RFID authentication protocols. In *Proc. of IFIP-SEC'07*, 2007.

[12] H. Y. Chien and C.-W. Huang. Security of ultra-lightweight RFID authentication protocols and its improvements. *SIGOPS Oper. Syst. Rev.* 41(4):83–86, 2007.

[13] M. Bárász, B. Boros, P. Ligeti, K. Lója, and D. Nagy. "Breaking LMAP", In *Proc. of RFIDSec'07*, 2007.

[14] M. Bárász, B. Boros, P. Ligeti, K. Lója, and D. Nagy. "Passive attack against the M2AP mutual authentication protocol for RFID tags", In *Proc. of the First International EURASIP Workshop on RFID Technology*, 2007.

[15] A. Klimov and A. Shamir. "New applications of T-functions in block ciphers and hash functions". *Proc. of FSE'05*, LNCS vol. 3557, pp. 18–31. Springer-Verlag, 2005.

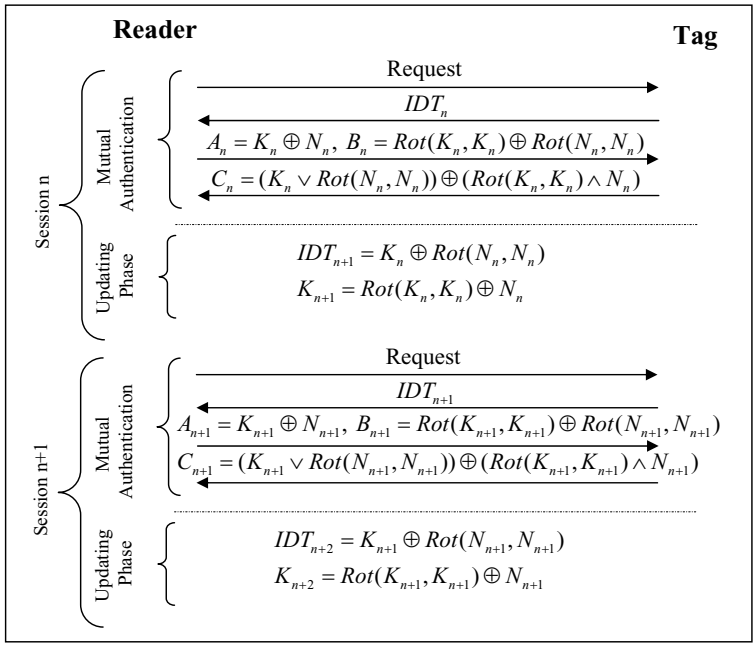
[16] R. Phan. Cryptanalysis of a new ultralightweight RFID authentication protocol - SASI. *IEEE Transactions on Dependable and Secure Computing* 6(4):316–320. Oct.-Dec. 2009.

[17] J. C. Hernandez-Castro, J. M. E. Tapiador, P. Peris-Lopez, T. Li and J.-J. Quisquater. Cryptanalysis of the SASI Ultralightweight RFID Authentication Protocol with Modular Rotations. In *Proc. of WCC'09*, Lofthus, Norway, May 10-15, 2009.

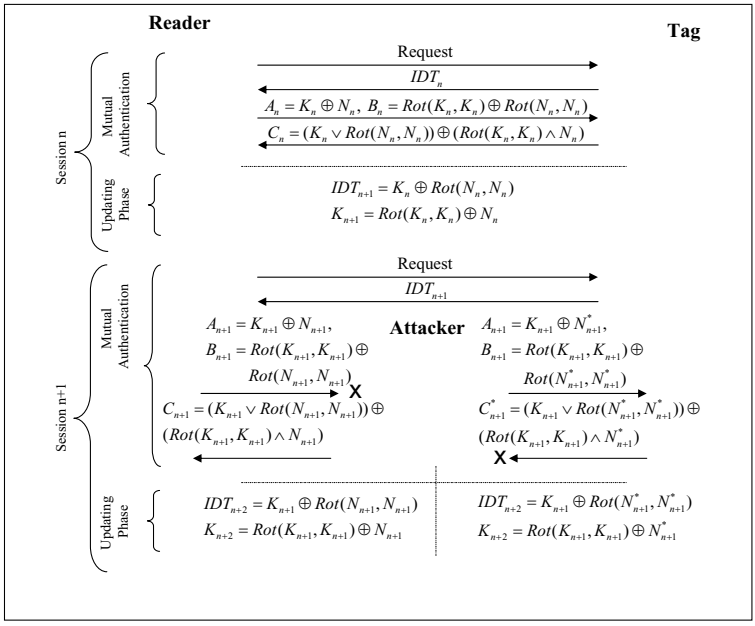
[18] P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estevez-Tapiador, T. Li and J. C.A. van der Lubbe. Weaknesses in Two Recent Lightweight RFID Authentication Protocols. In *Hand. of Workshop on RFID Security*, 2009.

[19] A. Juels and S. Weis. Defining strong privacy for RFID. In *Proc. of PerCom 2007*, pp. 342–347. IEEE Computer Society Press, 2007.

[20] B. Alomair and R. Poovendran. On the authentication of RFID systems with bitwise pperations. In Proc. of NTMS'08, pages 1–6, 2008.



(a) Full disclosure and cloning attacks



(b) De-synchronization attacks

Figure 2. Passive and active attacks

Radio Frequency Identification System Security
Y. Li and J. Zhou (Eds.)
IOS Press, 2010

doi:10.3233/978-1-60750-485-6-95

Semantic Access Control Model for RFID-enabled Supply Chains

Zang Li, Chao-Hsien Chu* and Wen Yao
College of Information Sciences and Technology
The Pennsylvania State University
University Park, PA 16802

Abstract. Radio frequency identification (RFID) has been considered as a viable solution for automatic data capture, information sharing and collaboration between enterprise partners. Along with the advantages of sharing information, it comes with the challenges of securing data and trade secret. Access control is an important method for securing data storing and sharing within and across partners in supply chains. In this paper, we propose a concept-level authorization model, aiming at addressing the challenges for securing RFID-enabled supply chains. We analyze the data characteristics and summarize common authorization challenges for RFID data from application perspectives. We propose to use concepts to reduce the size of policy repository and eliminate the structural heterogeneity across the information sources. We then focus on the details of semantic propagation rules that reflect the logical, time and spatial relationships. Other ideas such as policy groups and ontology merging techniques are employed to manage the authorization repository in an efficient manner.

Keywords: Semantic access control, RFID; authorization, supply chains

Introduction

Radio frequency identification (RFID) has been widely adopted to automatically identify, capture, and transmit information from tagged objects to enterprise systems, such as supply chain management (SCM), enterprise resources planning (ERP), and inventory management, via radio waves. Especially in SCM, large retailers like Wal-Mart, Albertsons and Target have begun employing RFID systems in their warehouses and distribution centers, and are requiring their suppliers and logistics partners to attach RFID tags to their products at the pallet and case levels. With the increased deployment of RFID in supply chains, how to manage the data flows over supply chains effectively while secure information storing and sharing between multiple data sources becomes a challenging task. Access control concerns the design and determination of whether a user can perform a specific operation on a resource.

As the basis of electronic supply chain solutions, RFID technologies serve a great mission of bringing together suppliers and buyers in supply networks into a global world wide web. The EPCglobal network aims at providing standard services for trading partners to discover RFID-associated information, and the EPC Information

* Corresponding Author. Tel: +1 (814) 865-4446; Fax: +1 (814) 865-6429; e-mail: chu@ist.psu.edu

Services (EPCIS) [1] serves as a standard query interface for supply chain partners to efficiently exchange information. However, this environment brings new challenges to the access control of RFID data sources from different partners:

- The multi-domain RFID-enabled systems involve more roles (users and organizations); thus, the number of authorization polices is expected to grow fast with the expanding of the data records. It is necessary to find an efficient way to solve the storage requirement by allowing the system to deduce authorizations from explicitly stored authorizations.
- RFID objects and data are defined and used by different organizations in their systems; thus, authorizations are required to reflect the access restrictions from different aspects such as physical alignments and time restrictions.
- Although the players may use the same methods for querying data across the supply chain with the help of EPCIS, their heterogeneous underlying databases with different data types and structures become a major obstacle for sharing and securing information. The access control mechanism should consider the semantic heterogeneity across various information sources in the partner organizations.

There has been some prior work on access control for federated databases [2], but they still cannot handle the challenges brought by RFID data and provide the supply-chain-wide access control services. Considering the above challenges, we develop a semantic access control model to manage authorizations of RFID data by using concepts to abstract heterogeneous data and structures and developing semantic propagation rules to reflect the logical, time and spatial relationships.

1. Related work and Preliminaries

1.1. Related Works

Access control (or called authorization) is a major issue for computer and network security in general and information security in particular. [3, 4] introduced early authorization concepts and models that the information system authorized or denied the access request. Role based access control (RBAC) model [5] is a sophisticated security model, which simplifies administration by assigning roles to users and then assigning permissions to those roles.

XML is one of the most popular languages to facilitate the sharing of structured or semi-structured data across different information systems. The use of access control mechanism for XML-based information has been discussed in a number of studies [6]. Most of these works provide language to define policies but do not offer mechanism to authorize access. Meanwhile, many researchers have studied security and privacy issues of RFID applications for several years. [7] provided a detailed survey on RFID security and privacy. Most of them, however, are studied from the perspective of communications between RFID tags and readers. Some studies have dealt with the access control issue of RFID information systems. For instance, [8] proposed an approach that exploits randomized read access control and thus prevents hostile tracking and man-in-the-middle attack; [9] proposed an access control and authorization model for security of RFID multi-domain based on Security Assertion Markup Language (SAML) and eXtensible Access Control Markup Language

(XACML). However, no existing works provided ways to express the unique characteristics embedded in RFID information services over supply chains and offered solution for solving the accompanied access control challenges, which is the focus of our study.

1.2. Node-level Authorization Model

Before we present our proposed model for RFID data in supply chains, it is important to describe why a traditional access control model would fail on such data. Here the traditional models are those in which the protected data nodes within XML documents are indicated by path-like structures. We call these models "Node-level Authorization Model (NAM)". As an example, we analyze the fine-grained access control model proposed by [6], which is considered as one of the classic ones in this field. The model defines the access right of users to elements and attributes within an XML document based on user's identity and policies, called authorizations. The access authorization for RFID information system can be represented as a 5-tuple, ar = {sub, obj, a, s, op}, where "sub" is the subject, "obj" is the object of the authorization, referenced by means of path expressions using XML Path Language (XPath), "a" is the action types such as "Read", "Write"; "s" refer to the decision of the policy either being authorized or forbidden; "op" contains such options as "DTD-level authorization" and "local authorization", etc. Suppose we need to create some policies to forbid the managers of the retailers from viewing the BOMs of all kinds of electronic products of the manufacturing companies or departments. The administrator who is in charge of policy making for the RFID data services may face some challenges.

- First, to write the XPath-based rules, the administrator has to be familiar with the structures of the bill of material (BOM) documents of different manufacturing companies or departments. Additionally, the knowledge of structures is also required for designing the propagation rules for the policies.
- Secondly, the number of electronic products is huge and the administrator needs to design policies for all the data documents.
- Thirdly, the same policies must be assigned for different data nodes with the same meaning, because they appear in different documents or locations.

Therefore, we need a more powerful model capable of handling these challenges.

2. RFID Data Characteristics

Several studies have discussed the types of RFID data and queries [1]. As a departure from these schemas, we provide a new classification scheme based on the semantics in the practical applications across the supply chain. We summarize the data types at a higher abstraction level without considering the data at the primitive level such as RFID readings. Moreover, besides the descriptive information for the data documents and elements, we examine four types of data nodes found in RFID data documents in supply chains.

2.1. Category/Product/Item

Category/product/item is the most popular structure for business data, especially inventories. This structure is primarily about specifying the class and type of product items. For inventory management, the products are usually classified into different categories. Products in a category will be differentiated by a unique code such as UPC (Universal Product Code), which identifies the type and brand of the product. On the other hand, in RFID era, EPC Network scheme includes four fields such as the version, the manufacturer, product type and a serial number unique to an item. That is, every item has its own identity, the so-called "Item-level" information. This makes the access to individual item possible. Figure 1 shows a simplified example of inventory.

```
<Computers>
  <Laptops>
    <name>A60</name>
    <Manufacturer>DEC</Manufacturer>
    <Price>DEC</Price>
    <Laptop EPC = "urn:epc:1.10.123.2">
      <Madedate>3/8/2008</Madedate>
      <Color>White</Color>
    </Laptop>
    <Laptop EPC = "rn:epc:1.10.123.4">
      <Madedate>3/8/2008</Madedate>
      <Color>Black</Color>
    </Laptop>
  </Laptops>
</Computers>
```

Figure 1. Illustration of category/product/item

The possibility of managing items' information brings challenges to authorization components. Besides the fact that we need to separately define the authorization polices for product- and item-level information, the number of potential objects to be protected becomes very large. These new objects include individual items and various semantic categories.

2.2. Spatial Relationships

The physical alignments among RFID objects are also important for access policy making and propagation. Containment is a hierarchical relationship between contained objects and containing objects. RFID tagging at different container levels, such as pallet-, case- and item-level, has been adopted to improve SCM by retailers and logistics. In an RFID information system, it is necessary for a logistics company to be able to access product information from the truck (or container) level or pallet level to the case level; however, for the retailers, this needs to be extended to the item level. Therefore, how to define access control rules for providing multiple views on different levels to satisfy information needs and authorization requirements is a critical issue for RFID systems, especially in SCM and logistics information management.

BOMs are hierarchical with the top level representing the end-item and the remaining indicating a "part list" needed to complete this end-item. In terms of RFID systems, the end-item and its components in a BOM file are usually identified by their EPCs. BOM information is transferred among enterprises and may serve in different

areas of the product life cycle along the supply chain such as design, production planning, product assembly/disassembly, pricing, sales service and product maintenance. According to different usages and users, different views of BOM information will be provided by the information system. Developing information sharing methods and access authorization mechanisms for multiple views is critical for authorization components.

2.3. Timestamped Records

In RFID applications, a large volume of timestamped records are generated based on the primitive tuples with the form {*EPC; location; time*} from RFID readers. Here we focus on the two major types of timestamped records, tracking historical data and transaction record. Timestamped records are common for RFID enabled asset tracking. At every transportation station, manufacturing or logistics facility reads the RFID tags on items or containers, and then update the location status and timestamps within the database. Moreover, keeping the timestamped location history (which is also called "trace") for an item is necessary. Especially, a logistics system provides a mechanism to trace the route of a pallet, case or shipment as it moves along the supply chain. Similarly, everyday there are various business transactions happening in a supply chain, such as purchase orders, advance shipping notices etc. For a transaction record, it often includes a transaction ID, together with a tag data of the issuing entity, and importantly, the date-time-stamp at which the transaction takes place.

For the timestamped data, the RFID information system needs to be able to provide different views in terms of the time, the role of users as well as the current stage along the supply chain. On one hand, a user is only allowed to access the information in a certain time interval. For example, a retailer can only obtain the trace information from distributer centers to the retailer. On the other hand, we may need to design different access control policies for different transactions according to the date-time-stamp information. For example, a seller is only allowed to see the transaction records of the current selling season.

3. The Proposed Access Control Model

In this section, we introduce a flexible authorization model with a compact policy repository to resolve the access control challenges of RFID data sources. This model, named as Concept-Level Authorization Model (CAM), defines the access right of users to data within the XML database based on subject's identity and semantics of the protected object.

3.1. Ontology-Level Access Control Model

Definition 3.1 Concept-Level Authorization Model. *Concept-Level Authorization Model* is denoted as a triple ($\mathbb{C}$, $\mathbb{R}$, $\mathbb{P}$), where $\mathbb{C}$ is a set of concepts, $\mathbb{R}$ is a set of relationships among concepts, and $\mathbb{P}$ is a set of authorization policies.

A concept here is an abstract idea or a mental symbol usually defined as the data types, a group of items, records and even attributes. Generally a concept can be named as a term of a specific domain, for example, electronics. Also, we can use the concepts

for special purpose, such as *the parts of a computer*, *the white computers* etc. Every concept is linked to an individual node or a set of nodes in XML documents.

Definition 3.2 Mapping. A *mapping m* is defined as a 2-tuple $\{c_i, (l_1, l_2, \ldots, l_n)\}$, where $c_i \in \mathbb{C}$, $(l_1, l_2, \ldots, l_n)$ is a set of data nodes in path-like expression. With the mapping, c_i is *defined by* (using the notation '$\dashv$') $(l_1, l_2, \ldots, l_n)$, denoted as $c_i \dashv (l_1, l_2, \ldots, l_n)$, or we say $(l_1, l_2, \ldots, l_n)$ is mapped to (using the notation '$\vdash$') c_i, denoted as $(l_1, l_2, \ldots, l_n) \vdash c_i$.

With the definition of mapping, for any $c_i, c_j \in \mathbb{C}$, if $c_i \dashv (l_1, l_2, \ldots, l_n)$, $c_j \dashv (l_1, l_2, \ldots, l_n)$, then $c_i = c_j$; if c_i, c_j are defined by different sets of data nodes, then $c_i \neq c_j$. For simplicity, our model requires every concept in the knowledge base is logically unique; that is, for any $c_i, c_j \in \mathbb{C}$, if $i \neq j$ then $c_i \neq c_j$.

Definition 3.3 Binary Relationship. A *relationship r* is defined as a 4-tuple, $r = (src, tgt, tp, \delta)$, where *src*, *tgt* are the two related concepts ($src, tgt \in \mathbb{C}$), *tp* stands for the type of relationship, δ refers to the propagation strength of this relationship, $\delta \in \{3/2, 1/2, 0, -1/2, -3/2\}$. The concept *src* and *tgt* are also named as *source concept* and *target concept*.

For a relationship *r*, we use functions **src**(*r*), **tgt**(*r*), **tp**(*r*) and $\boldsymbol{\delta}(r)$ to return its source concept, target concept, type and propagation strength. We denote it as $c_i \rightarrow c_j$ if there is a relationship from concept c_i to concept c_j, otherwise, $c_i \nrightarrow c_j$. The access control policy outlines the authorizations on systems and data. A policy *p* is defined as a triple, $p = (sub, obj, s)$, where *sub* is the subject to whom the authorization is applied, $obj \in \mathbb{C}$, which is the object defined at the concept level, *s* stands for the sign of the policy which can be positive or negative, $s \in \{+, -\}$. For a policy p, we use functions **sub**(*p*), **obj**(*p*), **sign**(*p*) to return its subject, object and sign.

3.2. Subjects and Groups

Usually, subjects are referred to the entities who initiate queries for RFID information. For our supply-chain-wide authorization problem, users and groups together with the membership relationship can be expressed in a hierarchical structure (See Figure 2). Every element in the hierarchical tree represents an *authorization subject group*, which may be mapped to an organization or a physical location. These groups form partially ordered sets, including many *hierarchies*.

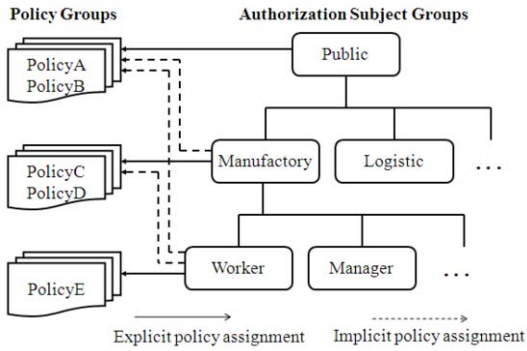


Figure 2. Policy groups and authorization subject groups

Definition 3.4 Hierarchy. A hierarchy is a triple (*s1*, *s2*, $\preccurlyeq$), where $\preccurlyeq$ represents a partial order on *s2*, indicating *s1* is the set of minimal elements of *s2* with respect to the partial order. It is also expressed as $s1 \preccurlyeq s2$.

We can also group the sets of policies within the application models as *policy groups*. With the hierarchical structure for authorization subject group, we can conveniently define policy groups. Policies may be assigned to appropriate policy groups and then authorization subject groups can subscribe to one or more of these policy groups. We use the notation $G1 \mapsto S1$ to express "policy group *G1* is assigned to subject group *S1*". With the policy groups and subject groups, we can deduce implicit policies by using the following rule:

Rule 1. The authorizations specified for an authorization subject group are applicable to all the descendent subjects; that is, for two subject groups S_i, S_j, if $\exists\ S_i \preccurlyeq S_j$, then for any policy group G with the assignment $G \mapsto S_j$, the assignment $G \mapsto S_i$ can be deduced.

3.3. Objects and Concept-Level Policies

In a traditional node-level authorization model, an object is a resource to which access is controlled. To create an access control policy, the policy maker must indicate its object by means of XPath in accordance to an XML document's structure. However, in practical uses, the potential authorization object of a concept-level policy may exist in different locations within a data source or even across different data sources. Moreover, the objects appearing in multi-domain data sources may have the same meaning and need to be assigned the same authorization policies.

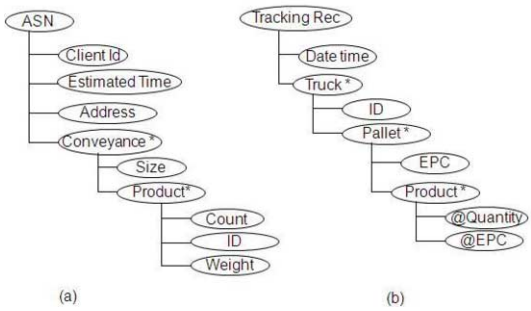


Figure 3. Illustration of structural heterogeneity between XML documents

Figure 3 depicts two XML schemas for simplified ASNs and simplified tracking records. The two schemas are presented as trees and the data are stored in different data sources (e.g., ASNs are managed by distributed center, and tracking records are managed by logistics). Moreover, ASNs are generated and received by traditional information systems such as ERP systems, whereas tracking records are the formatted reports from the new RFID asset tracking system. Consequently, the two information systems may use different terms describing the same attribute data. For example, the old system uses *count* whereas the newer uses *quantity*. The ASNs use the general word *conveyance* whereas the tracking records indicate the type of conveyance tools such as truck, train or airplane. Usually, for security purpose, it is reasonable and

necessary that we design the same policy for these objects with different names within heterogeneous structures. Suppose the requirement is "Logistics companies are not allowed to view quantity information of the products within conveyances". The traditional authorization system will define two XPath based policies in the two schemas:

(*Logistics*, */ASN/Conveyance/Product/Count*, −)
(*Logistics*, */Tracking Rec/Truck/Pallet/Product/@Quantity*, −)

Obviously, the traditional method of object definition on heterogeneous data sources brings a nontrivial burden to policy makers, not to mention that the policy makers have to be familiar with the structures of these heterogeneous data sources. These shortcomings are well handled by our semantic access control model by defining policy's objects at the concept level, so that policy maker can manage policies in a global-level without considering the structures of all the local-level data sources. For example, if we define a mapping:

{/ASN/Conveyance/Product/Count,/TrackingRec/Truck/Pallet/Product/@Qua}⊢Products in conveyances. Quantity

Then we have a concept policy:

(*Logistics*, *Products in conveyances. Quantity*, −).

A concept can be generated as any cognitive unit of meaning. For implementation, the model employs a *mapping table* to store mappings from XML source to concepts. The mapping table is maintained by software tools, which automatically link schema structures to ontologies, or by schema designers.

3.4. Ontology and Relationship Types for RFID data

In our model, the concepts are defined by ontologies that are controlled vocabularies for our specific problem domain – RFID information in supply chain networks. Following is the formal definition of *ontology*.

Definition 3.5 Ontology. An ontology $\mathbb{O}$ is a formal representation of a set of concepts within a domain and the relationships between those concepts; that is, $\mathbb{O}$ =($\mathbb{C}$, $\mathbb{R}$), where $\mathbb{C}$ is a set of concepts, $\mathbb{R}$ is a set of binary relationships among the concepts in $\mathbb{C}$.

Figure 4 shows the common relationships around RFID items. As shown, the concepts defined for categories, products, a set of items (or an individual item), and timestamped records are connected via various relationships which are derived from the summarization of structures introduced in section 3. Considering the tri-dimension relationships among RFID items: logical organizations, spatial alignments and timestamped records, we identify several useful relationship types among these concepts:

- *instanceOf*. A relationship between concept c_i and c_j (c_i,c_j∈$\mathbb{C}$) has the instanceOf type, iff the items denoted by c_i is an instance of the product or category denoted by c_j.
- *subclassOf*. A relationship between concept c_i and c_j (c_i,c_j∈$\mathbb{C}$) has the subclassOf type, iff any instance of concept c_i is also an instance of concept c_j.

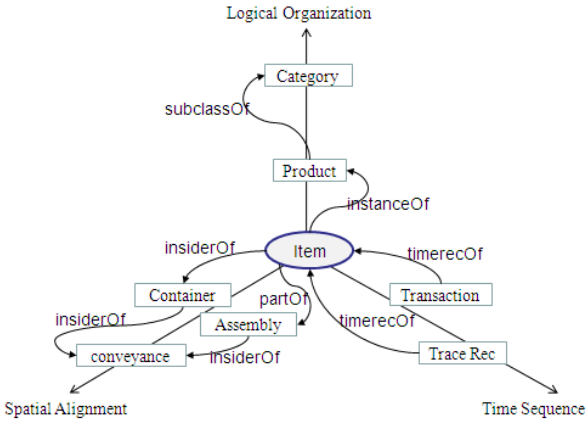


Figure 4. Relationships in different dimensions

- *partOf*. A relationship between concept c_i and cj (c_i,cj∈ℂ) has the partOf type, iff the objects denoted by c_i are the parts of the object denoted by c_j.
- *insiderOf*. A relationship between concept c_i and c_j (c_i,c_j∈ℂ) has the insiderOf type, iff the items denoted by c_i are spatially insider of the item denoted by c_j.
- *timerecOf*. A relationship between concept c_i and c_j (c_i,c_j∈ℂ) has the timerecOf type, iff the record set denoted by c_i are the timestamped information of the items denoted by c_i.
- *attributeOf*. A relationship between concept c_i and c_j (c_i,c_j∈ℂ) has the attributeOf type, iff the record set denoted by c_i are the attribute data of the items denoted by c_j.

Among which, the *subclassOf*, *partOf*, *insiderOf* relationships are transitive. That meant, given two triples *(Computers, subclassOf, Electronics)* and *(Laptop, subclassOf, Computers)*, we can infer the triple *(Laptop, subclassof, Electronics)*.

3.5. Propagation Factors

For any relationship *r*, we analyze how the access decision of *tgt* can be propagated to *src*. The access decision d_c indicates whether a positive authorization (+), a negative authorization (−), or no authorization (ε) should be applied to the requested concept *c*. The access decision d_c can also be written in a function format **d**(*c*).

The $|\delta|$ is the strengths of relationships. As we will see in section 4.2, suppose the strength of any explicit policy for making the final access decision is 1 (representing *regular*) and the strength of no propagation of policy is 0, then the strength of a relationship *r* with $|\delta|$= 3/2 is *strong* and the propagation of policy along *r* may take precedence over the policies on the *src*. Reversely, if relationship *r* is defined as weak ($|\delta|$= 1/2), the propagation of policy along *r* may be overridden by the policies on the *src*. On the other hand, the sign of δ can classify the relationships into three groups:

inferable, non-inferable, and reverse-inferable. Given a relationship $r = (c_i, c_j, f, \delta)$, if $\delta > 0$, then the relationship is inferable, denoted as $c_i \rightarrow_{\mathrm{I}} c_j$; if $\delta < 0$, then the relationship is reverse-inferable, denoted as $c_i \rightarrow_{\mathrm{IR}} c_j$; if $\delta = 0$, then the relationship is non-inferable, denoted as $c_i \rightarrow_{\mathrm{N}} c_j$.

4. System Implementation

The proposed semantic access control model has been carefully implemented and verified. The major components include 1) an efficient algorithm to extract the onto-inference graph from the ontology graph, 2) a procedure to identify applicable policy, 3) the rules to resolve conflict, 4) the rule to propagate policies, and 5) the process of computing access decision. In this section, we focus our attention on the extraction of onto-inference graph and mechanism of the authorization system.

4.1. Authorization Evaluation Process

In our model, the authorization evaluation process is operated on an ontology graph. For explicitly purpose, ontology graph are usually employed to represent an ontology.

Definition 4.1 Onto-inference Graph. Given a vertex v in ontology graph $\mathbb{G}=(\mathbb{V},\mathbb{E})$, its Onto-Inference graph $\mathbb{G}_{\mathrm{I}}(v)$ can be described as a 2-tuple $\mathbb{G}_{\mathrm{I}}=(\mathbb{V}_I, \mathbb{E}_I)$, where $\mathbb{V}_{\mathrm{I}}$ is the set of related concepts which are identified by function *Related-Concept*(v), and $\mathbb{E}_{\mathrm{I}}$ is a set of directed edges, representing the inferable or reverse-inferable relationships among the concepts in $\mathbb{V}_{\mathrm{I}}$.

To obtain the onto-inference graph, the model conducts a depth-first expansion. The algorithm first locates the node which represents the object concept of the request in ontology graph, and defines the node as the root of the onto-inference graph. Then the onto-inference graph expands by following its inferable outgoing edges to the parent nodes of the root node. The incoming edges and non-inferable edges are ignored since they have nothing to do with propagations. For each parent node, the algorithm adds it into the onto-inference graph by storing its relationship to the current node. This process continues until it hits some nodes that has no parents. Then the search backtracks and starts propagating the access decisions along these backtracking paths.

Now we discuss how to compute the access decision based on policy options and conflict resolution rules. We first introduce the definition of weighted decision pair.

Definition 4.2 Weighted decision pair. A *weighted decision pair* (Ω) is defined as a 2-tuple (θ, ω), where θ is an access decision label ($\theta \in \{+, -, \varepsilon\}$), ω refers to the weight of the decision, $\omega \in \{3/2, 1, 1/2, 0\}$.

We also define functions $\boldsymbol{\theta}(\Omega)$ and $\boldsymbol{\omega}(\Omega)$ to return the access decision label and weight. Now let us consider the problem we have: for a request initiated by requester rq, the model computes the access decision (denoted as d_f). There are four steps to decide the access decision on a concept c: 1) *Identify applicable policies*. Identify the set of authorization policies (denoted as $\mathbb{P}_{rq,c}$), which are defined on c and are applicable to rq; 2) *Resolve conflicts*. Resolve the conflicts of policies in $\mathbb{P}_{rq,c}$ and generate a weighted decision pair Ω_c; 3) *Propagate policies*. Translate access decisions of parents

and corresponding relationships into weighted decision pairs, based on which a new weighted decision pair $\Omega_{\wp}$ is generated. 4) *Compute access decision*. By combining all Ω_c and $\Omega_{\wp}$, compute d_f.

Notice that a weighted decision pair Ω_c is defined to describe the result of conflict resolution. If the $\mathbb{P}_{rq,c}$ is empty, the weight of Ω_c of is specified as 0; otherwise the weight is set as 1.

Step 1: Identify applicable policies. In this step, the model finds the corresponding policy set according to the requester *rq* and the requested concept. Let $\mathbb{P}_{rq,c}$ be the set of applicable policies, we can compute the set using the following process:

$$\mathbb{P}_{rq,c} = \{p = (subject, object, sign) \mid p \in \mathbb{P},\ rq \preccurlyeq subject, c = object\} \quad (1)$$

Step 2: Resolve conflict. In this step, the model handles the situations that several authorization policies resulted in different rights in $\mathbb{P}_{rq,c}$. Let $\mathbb{P}_{rq,c} = \{p_1, p_2, \ldots, p_n\}$. We need to employ some rules to choose a policy (denoted as p_f) which will be used to make the final access decision for the request. Let '$\wedge$' denote the action choose between two policies (notice that $p_i \wedge p_j = p_j \wedge p_i$), the conflict resolution can be expressed as: $p_f = p_1 \wedge p_2 \wedge \ \ldots \ \wedge p_n$. Although several conflict resolution rules from previous research works [10] can be used, for simplicity, our model employs the following conflict resolution rules for two conflicting policies defined on c:

Rule 2. Given conflicting policies p_i and p_j, if $\mathbf{sub}(p_i) \preccurlyeq \mathbf{sub}(p_j)$, then $p_i \wedge p_j = p_j \wedge p_i = p_i$.

Rule 3. Given conflicting policies p_i and p_j, if $\mathbf{sub}(p_i) = \mathbf{sub}(p_j)$ and $\mathbf{sign}(p_i) =$ '$-$' , then $p_i \wedge p_j = p_j \wedge p_i = p_i$.

With the conflicts rules, the calculation of Ω_c can be complete as:

$$\Omega_c = \begin{cases} (\varepsilon, 0), & \text{if } \mathbb{P}_{rq,c} = \emptyset \\ (\mathbf{sign}(p_i), 1), & \text{if } \mathbb{P}_{rq,c} \neq \emptyset \text{ and } no\ conflicts \\ (\mathbf{sign}(p_1 \wedge p_2 \wedge \ldots \wedge p_n), 1), & \text{if } \mathbb{P}_{rq,c} \neq \emptyset \text{ and } conflicts \end{cases} \quad (2)$$

Step 3: Policy propagation. Suppose the set of parents of c is $\{\wp_1, \wp_2, \ldots, \wp_m\}$, for any $\wp_I$ $(1 \leq i \leq m)$, the access decision of $\wp_i$ is denoted as $d_{\wp i}$, the inferable or reverse-inferable relationship from *c* to $\wp_i$ is denoted as $r_{\wp i}$. The model generates the set of weighted decision pairs $\{\Omega_{\wp 1}, \Omega_{\wp 2}, \ldots, \Omega_{\wp m}\}$ based on the following rule, where $\Omega_{\wp i} = (\theta_{\wp i}, \omega_{\wp i})$, $1 \leq i \leq m$.

Rule 4. Given a relationship $r = (c_i, c_j, f, \partial)$ and the access decision $\mathbf{d}(c_j)$ on c_j, a weighted decision pair (θ, ω) is generated as:

$$\theta = \begin{cases} \mathbf{d}(c_j), & \text{if } \boldsymbol{\delta}(r) > 0 \\ \sim\mathbf{d}(c_j), & \text{if } \boldsymbol{\delta}(r) < 0, \\ \varepsilon, & \text{if } \boldsymbol{\delta}(r) = 0 \end{cases} \qquad \omega = |\boldsymbol{\delta}(r)| \quad (3)$$

Here the operator $\sim$ is defined on $\{+, -, \varepsilon\}$, the truth table can be referred to Figure 5.

A	~A
ε	ε
−	+
+	−

A	B	A⊙B
ε	ε	ε
ε	−	−
ε	+	+
−	ε	−
−	−	−
−	+	−
+	ε	+
+	−	−
+	+	+

Figure 5. Truth tables

We can then combine all of these weighted decision pairs by using the following formula:

$$\Omega_{\wp}=\Omega_{\wp 1}\otimes\Omega_{\wp 2}\otimes\ldots\otimes\Omega_{\wp m} \tag{4}$$

Here, $\Omega_{\wp}$ represents the combination result of all the weighted decision pairs, and the operator $\otimes$ is defined as:

$$\Omega_{\mathrm{i}}\otimes\Omega_{j}=\begin{cases}\Omega_{\mathrm{i}}, & \text{if } \boldsymbol{\omega}(\Omega_i)>\boldsymbol{\omega}(\Omega_j)\\ \Omega_{j}, & \text{if } \boldsymbol{\omega}(\Omega_i)<\boldsymbol{\omega}(\Omega_j)\\ (\boldsymbol{\theta}(\Omega_i)\odot\boldsymbol{\theta}(\Omega_j),\boldsymbol{\omega}(\Omega_i)), & \text{if } \boldsymbol{\omega}(\Omega_i)=\boldsymbol{\omega}(\Omega_j)\end{cases} \tag{5}$$

Where, $\odot$ is defined on the 3-values set {+,−, ε} and the truth table can be referred in Figure 5. Notice that A⊙B=B⊙A, and accordingly $\Omega_i\otimes\Omega_j=\Omega_j\otimes\Omega_i$. As we can see from the formula, the weight in a weighted decision pair represents the priority of a pair. Moreover, the model always stays on the safe side and applies "denials (−) take precedence over permissions (+), and permissions take precedence over no-authorization (ε)".

Step 4: Access decision. In this step, the model calculates the final access decision by comparing the weight of Ω_c and the weight of $\Omega_{\wp}$. The one with bigger value of weight will be defined as the winner and the access decision label of the winner will be the final access decision d_f. Notice that $\boldsymbol{\omega}(\Omega_{\wp}) \in\{1/2, 3/2\}$ because relationships between any parent $\wp_i$ should not be non-inferable, and $\boldsymbol{\omega}(\Omega_c) \in\{0, 1\}$. Therefore, $\boldsymbol{\omega}(\Omega_{\wp})\not\equiv \boldsymbol{\omega}(\Omega_c)$. Also, if $\boldsymbol{\omega}(\Omega_c) > \boldsymbol{\omega}(\Omega_{\wp})$, then $d_f = \boldsymbol{\theta}(\Omega_c)$; if $\boldsymbol{\omega}(\Omega_c) < \boldsymbol{\omega}(\Omega_{\wp})$, then $d_f = \boldsymbol{\theta}(\Omega_{\wp})$. By using the operator $\otimes$, we know:

$$d_f = \boldsymbol{\theta}(\Omega_c\otimes \Omega_{\wp})= \boldsymbol{\theta}(\Omega_c\otimes\Omega_{\wp 1}\otimes\Omega_{\wp 2}\otimes\ldots\otimes\Omega_{\wp m}) \tag{6}$$

4.2. System Components and Work Flow

Figure 6 presents a solution of integrating our concept-level authorization model with RFID information systems. As shown, the system includes four major components: data collector, data processor, authorization engine, and ontology analyzer. Data collector collects RFID events and gets master data from external sources. Data processor is responsible for parsing queries, exchanging information with the authorization component, and responding user with pruned data. The authorization engine plays a key role in authentication and authorization process. Authentication

functions choose the authentication method and determine whether a requested service is allowed to access or not based on the identity established during authentication. Authorization function involves gathering concept-level policies, granting access based on the policies defined on related concepts. The ontology analyzer executes the transformations between concepts and XML data and fetches related concepts and relationships from ontology knowledge base for the authorization process. Figure 6 also shows basic work flow of policy making and evaluation, where, the information server is built upon distributed data sources, which are managed by different organizations. For each data source, a local ontology repository and mapping table are maintained.

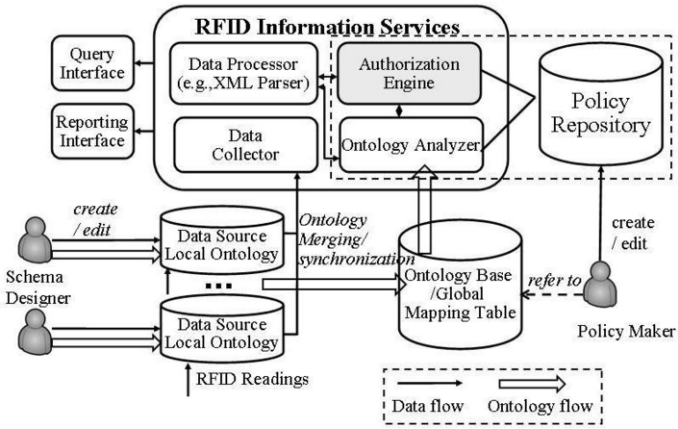


Figure 6. System architecture and work flow

The process of policy making can be summarized as follows:

- The schema designers abstract security sensitive information into concepts when the XML schemas are created or modified. Accordingly, the entries in local mapping table are created or modified. A software tool can be developed to automatically edit the ontology and mapping table according to the changes of XML schema.
- The process of ontology merging and synchronization takes multiple local ontologies as inputs and outputs the integrated ontology into the global ontology base. If two concepts from two local ontologies have the same semantics, and importantly, the same requirements of access protection, then they can be merged into a concept; that is, an entry in the global mapping table is generated or updated by combing the according entries from the two local mapping tables.
- The policy maker creates and edits concept-level policies with reference to the global ontology base, and then the policies are stored in policy repository.

As we can see, the ontologies for data sources are created by the schema designers who are most familiar with the structures of local data documents. The ontology knowledge base and mapping table change as the schema changes, therefore this can be done automatically by editing tools, and will not bring heavy burden to the designers. Thus, the policy makers can design the policies based on semantics of concepts without knowing where the concepts are and what the structures they have.

5. Conclusion

Access control can be viewed as the central element of information security for RFID information services, of which the mission is to serve data sharing within and across enterprises in supply chains. In this paper, we propose the concept-level authorization model as a mechanism of access control for RFID information services. We first discuss the RFID data characteristics from different application perspectives and summarize common authorization challenges for RFID data in supply chains. A concept level authorization model is then introduced based on the following two key ideas: (1) Conceptualize authorization objects based on RFID data features and (2) Design implicit authorizations based on relationships among concepts and organization hierarchies.

Besides the notation system of our model, some ideas such as generating concepts and describing relationships from multiple dimensions of RFID data are suggested. Additionally, we propose to borrow ideas (such as ontology merging) from frameworks of data integration and apply them to manage authorizations of distributed data sources. Furthermore, we provided computational methods for the authorization evaluation process. To our best knowledge, this is the first paper systematically providing a mathematical foundation for the authorization evaluation process of semantic authorization model.

Our proposal leaves space for future research works, either from theoretical aspect or application aspect. We are developing a prototype based on existing open-source projects such as *KAON Ontology Framework*. Based on the prototype, some issues can be investigated and tested, for example, the consideration of evaluating XPath queries, the integration of NAM and CAM, and the optimizations of the authorization evaluation algorithms.

References

[1] M. Harrison and others, "EPC information service-data model and queries," *Auto-ID Center White Paper*, 2004.

[2] K. Taylor and J. Murty, "Implementing role based access control for federated information systems on the web," in *Proceedings of the Australasian information security workshop conference on ACSW frontiers 2003-Volume 21*, 2003, pp. 87–95.

[3] D. E. R. Denning, *Cryptography and data security*, 1982.

[4] S. Castano, M. Fugini, G. Martella, and P. Samarati, *Database security*: Addison-Wesley Reading, MA, 1995.

[5] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman, "Role-based access control models," *IEEE computer*, vol. 29, pp. 38–47, 1996.

[6] E. Damiani, S. C. Di Vimercati, S. Paraboschi, and P. Samarati, "A fine-grained access control system for XML documents," *ACM Transactions on Information and System Security (TISSEC)*, vol. 5, pp. 169–202, 2002.

[7] A. Juels, "RFID security and privacy: A research survey," *IEEE Journal on Selected Areas in Communications*, vol. 24, pp. 381–394, 2006.

[8] X. Gao, Z. Xiang, H. Wang, J. Shen, J. Huang, and S. Song, "An approach to security and privacy of RFID system for supply chain," in *IEEE International Conference on E-Commerce Technology for Dynamic E-Business*, vol. 2004, 2004, pp. 164–168.

[9] D. S. Kim, T. Shin, B. Lee, and J. S. Park, "Access Control and Authorization for Security of RFID Multi-domain Using SAML and XACML," *Lecture Notes in Computer Science*, vol. 4456, pp. 887, 2007.

[10] S. Jajodia, P. Samarati, M. L. Sapino, and V. S. Subrahmanian, "Flexible support for multiple access control policies," *ACM Transactions on Database Systems (TODS)*, vol. 26, pp. 214–260, 2001.

Radio Frequency Identification System Security
Y. Li and J. Zhou (Eds.)
IOS Press, 2010

doi:10.3233/978-1-60750-485-6-109

Security in the Internet of Things

Manfred Aigner [a]
[a] *IAIK, Graz, University of Technology, Austria, Manfred.Aigner@iaik.tugraz.at*

Abstract. In this paper we illustrate security problems coming up with the new concept of the Internet of Things. Passive RFID tags will make up the majority of devices participating in this network. Since passive RFID tags are the devices with the least computing power in the IoT, we focus our investigations on this technology. If we can provide a proper protection for those devices, it will also be possible to use the same security mechanisms on other technologies. In the following section we explain the relevance of passive RFID and security for the upcoming IoT. Later we will provide information about security issues for this technology and explain the current state-of-the-art in research.

Keywords. IoT, RFID, Security

1. Passive RFID Technology and the Internet of Things

The term "Internet of Things" has meanwhile become a scientific discipline that deals with the interlink of physical objects and the Internet. Various technologies are discussed as building blocks of this development, like RFID, short-range communications, ultra wide band communications, real-time localization, sensor networks or ad-hoc networks. In 2008 the first version of Conference "Internet of Things" took place. New business models, technologies and applications were discussed. The European Commission organized a series of workshops to develop a research road-map that should ensure proper development.

The Internet of Things will be characterized by communication between objects without human interaction, rather than data transfer between computers that are operated by persons. In the traditional Internet, as we know it today, most communication is triggered by human interaction, e.g. when we open a web-page or send an email with attachments to the mailbox of a recipient. Communication in the Internet of Things will be enabled when e.g. objects enter into the area that is covered by a reader device. While present within the reading range it may connect to a server for inventory or object tracing purposes in a supply-chain. An autonomous node in a network of sensor nodes might issue an alarm to a controlling system as soon as it measures a critical value.

Although we can foresee some scenarios for applications using the Internet of Things, it is impossible to correctly predict typical applications or killer applications of this upcoming network. Similar to the development of the Internet we will face new classes of applications that are hard or even impossible to forecast (remember e.g. the development of Web 2.0 that is based on user generated content). Nevertheless, already

today we can foresee that a big number of application will need security measures as prerequisite for successful launches. We can easily compare the situation with the world wide web (WWW) in its early years. At the beginning the Internet was a network that combined servers from various universities. Facilitated data exchange was the first use case for the network. All involved parties basically trusted each other and first applications did not really require protection, but the goal was to establish a connection for exchange of data. When CERN came up with their first version of the WWW the basic idea was to have a presentation platform for research results to interested audience. Newspapers started to use the web as platform and companies used it to provide information about there products. More advanced applications like electronic banking or webshops were not possible until protection of the communication was established. Due to the high risk of fraud no company would provide access to critical data via the web without secure authentication of the servers and encrypted communication between server and clients. Looking at current killer applications in the Internet, most of them use some sort of security measures. Especially for commercial services it is necessary to achieve the secure authentication of communication partners and protection of the stored data against attacks. The short history of the Internet has shown that it is necessary to protect any server that hosts valuable data or services, we can assume that this will also account for the Internet of Things. To illustrate this statement, we distinguish three different classes of protection for applications in the Internet:

Protection of servers against intrusion and denial of service: Any server holding valuable data needs to be protected against modification of its content. Although it is often not visible to a standard user, companies (like Google or Yahoo) need to protect their servers against illicit access and denial of service attacks. Strict access control, firewalls and other measures are used to protect the servers. The main motivation for protection is the commercial value of their service and data. If databases or web-pages were modified by attackers their business would seriously suffer.

Authenticated access: Many applications require authentication of the client to grant access to specific data and services. Typically the end-user needs to log in, to receive personalized services or data. As example you can think of web-based email or platforms to share photos. The service provides only the data that the logged-in end user should have access to. On the other hand the authentication protects the end-user's data from access by other, probably unknown users. Nobody would use e.g. a web-based email service that would automatically allow any other users to access personal data. It is therefore necessary for the service provider to protect the servers in a similar way as above, but additionally to demonstrate to the end-users that their personal data is protected from access by others. In some applications like e.g. platforms for sharing photos or videos, endusers may grant access to other users or groups of users.

Encrypted connection: The third class of applications requires more protection. In some cases access control to data is not enough, but the transferred data is critical. The application needs to make sure that the data is not modified or eavesdropped on it's was between the server and client. Any webshop with payment facility falls into this category. The end-user wants to be sure that information about his credit card is not eavesdropped nor modified when transferred to the server. Additionally to protection of the servers against illicit access and authentication with proper management of access rights, this class of application requires encryption of the transferred data and integrity checks.

During the development towards the current Internet, important but insecure services were replaced step-by-step with protected versions. Instead of remote login via telnet or rlogin, nowadays ssh (secure shell) is typically used to establish a remote connection to a computer.

The development of the Internet of Things is still in an early phase. First applications are currently developed, most of them operate in a closed loop environment, without public access. Technology to interact with the network is not yet pervasively installed, but already available. Based on the development in the Internet we can foresee similar security requirements for applications in the new network.

1.1. The role of passive RFID in the IoT

The term RFID isn't well defined nor consistently used in the scientific literature. RFID stands for Radio Frequency Identification, so in principle for any technology that transmits an identifying information via radio-frequency. Friend or foe identification systems for aircrafts are presumed to be the predecessors of modern RFID systems. Modern RFID systems consist of tags or transponders which are able to store a identifying number. As soon as they enter the range of a reading device they transfer their ID. Transponders are separated into active and passive and semi-passive ones. Active tags use their own power supply, therefore they can actively send information which leads typically to a longer reading distance. Passive tags draw the power for operation from the reader's electromagnetic (EM) field. High volume production allows to produce such tags at very low cost. The functionality of passive tags is rather restricted due to the low power that can be extracted from the surrounding EM-field. Semi-passive tags are a hybrid of the previously described ones. They communicate passively but they have their own power supply, typically used for additional functionality. Currently they do not play an important role but with upcoming applications for sensor tags they will become more relevant in future RFID applications.

In the context of this paper passive RFID tags are the most interesting ones. Due to their low production costs and they can be attached to basically any object in our surroundings. A variety of such passive RFID transponders exist. The most powerful ones in terms of computation capabilities are currently contact-less smart cards. This devices communicate with readers in a reading distance of up to some centimeters. They are often used as security devices for e.g. access control systems or in ticketing systems. The short reading distance is not a restriction, but actually a feature - the user is forced to bring the card close to an access point to trigger deliberately a transaction. Due to the short reading distance the contact-less device receives enough power from the reader so that even complicated cryptographic operations are possible. The contact-less interface replaces the contact-based smart card interface, instead of sliding the card in the smart card reader it is just brought close to the reading point. This improves both, user convenience and transaction speed.

Another sort of tags are designed for longer reading distances. Such tags are typically used in supply chain automation. They are optimized for lowest power consumption and low cost, since this type of applications require a very high number of tags. In the following paragraphs we will point out why this sort of RFID tags are interesting on the context of the Internet of Things. Although this tags look similar to contact-less smart cards in their physical appearance, their implementation is very different due to

the different design requirements for reading distance and unit costs. In the remainder of this paper, we refer with the term *RFID tag* or *RFID transponder* to this specific sort of low cost tags which are built to operate in higher reading distances.

The main goal during development of such RFID tags is cost reduction. The silicon area of the tag's chip is an important cost factor for such tags. The costs for development are insignificant in the cost calculation for a single tag, since a very high number of tags is produced.

We can assume that low cost RFID tags will make up the majority of devices in the Internet of Things. Those tags - when attached to a physical object - can provide an electronic identity of objects. Due to the low cost of these tags we can assume that a very high number of things will be tagged and therefore enabled to participate in the network. RFID tags are nowadays used in applications for supply chain automation. Currently approximately one billion of tags is produced worldwide per year. The costs for such tags is already below five US cents.

The future Internet of Things will allow homogenous access to tags from and to the traditional Internet, as well as compatibility to ad-hoc sensor networks and other networks of mobile devices (e.g. devices operating in GSM or UMTS networks). In this network the passive tags of will be the devices with least performance and therefore mark the lower barrier for data throughput. If we compare passive RFID tags with other wireless connected devices like sensor nodes we realize a dramatic difference in the power consumption of the circuits. With given voltage levels, the power consumption directly translates to the continuous current consumption of the circuits. While battery operated devices or passive smart cards typically have an average continuous current consumption between 10 mA and 100 mA. The circuits of passive RFID reset themselves when the power consumption exceeds 0.01 mA - 0.02 mA. The low level of energy available in the reader's EM field at the maximal reading distance and the minimal size of the chip and antenna prevents higher power levels on RFID tags. We observe a difference of power levels of factor higher that 1000 between passive RFID tags and battery powered devices. This will not change in the next years, since the given parameters for reading distance and antenna or chip size will not change in near future. The power consumption of the circuits themselves will be reduced following Moore's Law. Nevertheless, it will take very long until today's RFID tags will provide similar capabilities as today's sensor nodes or similar battery operated devices.

1.2. Security for RFID

In 1999 the Auto ID Center at MIT started to promote their vision towards "The Networked Physical World" [16]. One of their starting assumptions was that only inexpensive tags can enable the development of RFID technology. [17]. Automation of supply chain management by tagging of every produced good was defined as the application for this new network. They developed a system based on very limited functionality provided by the tags, to allow production of such low cost tags for a price below 5 cents. All intelligence was considered to be computed by the network to keep the chip area, and therefore the cost for a tag as low as possible. Security requirements for the tags were not considered, but the tags should hold a permanent ID and EPC and transfer it whenever requested by a reader.

A consequence of the necessary unique identifer (UID) for each tag and tagged object as electronic product code (EPC), an extensive discussion on privacy implication

occurred. When the unique ID of a tag or an object can be directly of indirectly mapped to a person who carries the object the data protection problem is indeed significant, because the information of the tag has then to be treated as "personal data".

1.2.1. Kill command

A straightforward solution to this privacy problem is to disable the tags at the POS (point-of-sale), so that tags are not operating anymore when the end-consumer takes over the tagged goods. The so-called kill-command was introduced in the tag communication protocol. When a tag receives such a command it disables itself unrecoverable. This approach solves the privacy problems for supply chain application, but it raises others:

No after POS application of tags: When tags are killed at the POS they cannot be used in other applications. Often tags are integrated in the objects, so they could be easily used in processes like warranty refund or disposal. Due to the use of the kill command to tackle the privacy problem we loose a lot of additional benefits of tagged objects.

Unwanted execution of the kill command: A supply chain management system that relies on the information of the tags can be jammed by executing kill commands for tags while they are still in the supply chain, e.g. during loading or unloading of goods to from and to a lorry. For most tags, the kill command is protected by a password that needs to be sent to the tag before it accepts the command. Managing such passwords is quite complex, if each tag uses a different password. On the other hand, it is easy to eavesdrop and reuse passwords, in case that it is used for more than one tag.

1.2.2. Blocker tag

The blocker tag is a suggestion by RSA Security to cope with the privacy problem [8]. It is a tag that can be attached to a shopping bag to protect the end consumer from unwanted reading. As long as the blocker tag is in the same reader field as other tags, the reader is not able to communicate with the tags, because the blocker tag prevents the resolving of collisions. The solution is effective but has several drawbacks. It is possible that a tag in the pocket is within the reading distance of a reader while the blocker tag of the bag is not. In this case it is still be possible to read certain tags. For many products it might be impossible or not desirable to put a bag around it (wrist watch) so an end consumer would need to apply a blocker tag near working tags to be protected. Additionally, a blocker tag will also block wanted RFID communication, if it is unintentionally brought into the reader field, of e.g. an automatic cashier reader. As final point we can observe that the solution to apply blocker tags puts burden and potential costs on the end-consumer to protect his privacy.

1.2.3. Proprietary crypto on tags

Due to the very restricted requirements for low cost RFID tags it was often proposed to use proprietary protection measures for RFID tags. This argumentation follows an assumption that published algorithms might use higher resources during execution and that undisclosed encryption algorithms might provide an accepted security level. Very successful RFID-tag products have been produced following this assumption. One example is the first version of Philips' MIFARE tags which used the undisclosed CRYPTO1 algorithm [14], another TI's DST (Digital Signal Transponder) [9] using DST40 as cryptographic primitive. Both approaches were meanwhile broken by academic groups [6] [1].

When proprietary algorithms are used, one needs to consider that the chance for successful attacks gets very high as soon as secret details about the encryption algorithm become public. The decision of such an approach may be reasonable for a specific application, application of undisclosed primitives for protection of an open network is not meaningful.

1.2.4. Pseudonyms for tags

A tag pseudonym is a changing ID that appears to a non authorized reader as random number. Only authorized readers should be able to dissolve the pseudonym to find out the real ID of the tag. Typically, a central computing centre is involved that dissolve the pseudonym for the distributed readers. To fully tackle the privacy problem it is necessary that the ID changes every time when a tag is read so that tracing for collected pseudonyms gets useless. Various schemes to generate pseudonyms on tags and to find their real ID efficiently were published in the recent years [12]. The centralized nature of the computation centre is feasible for e.g. application in closed loop system like libraries, but prevents many possible applications in the open Internet of Things.

1.2.5. Standardized crypto tags

Meanwhile, it is established as good practice to use standardized cryptographic primitives and protocols for security operations in the Internet. Following Kerckhoffs' principle all parts of the cryptographic system, except the key, must not rely on secrecy. During the standardization phase of public evaluation of the algorithms or protocols is carried out so that potential security holes can be corrected. Design of open systems prevents application of secret algorithms and protocols, since it would be very likely that attackers would get access to the secret it it is know by many parties.

Comparing standardized solutions with ad hoc solutions for a specific applications they often seem to be less efficient. After a closer look, often new flaws are detected in the ad-hoc solutions and after fixing them the overall performance gain is lost. For many years it was argued that implementation of standardized cryptographic primitives is not possible on RFID tags, due to the very restricted power budget. Our results show that current technology allows implementation of such algorithms without any restriction of the reading distance. The possible gain of chip area by application of proprietary algorithms does in our opinion not justify the arising risks.

We think that use of established and standardized cryptographic protection measures for RFID tags is the best and most efficient way towards secure RFID technology.

1.3. Security for the Internet of Things

For proper establishment if an open Internet of Things we foresee security as a service enabler. As in the Internet there will be also application without security requirements. Nevertheless, especially commercial applications will require protection of the data that is communicated. When we state protection we do not only refer to encrypted communication, but also other security features. To protect the privacy of people who carry tagged goods encryption of the tag's communication is certainly an important feature, but it is not the only one. Data integrity in end-to-end connections with passive tags is becoming an important issue. When an application communicates with a remote tag, via a reader

and other servers that are not under own control it will be required to detect if the data originating from the tag was not changed on its way. Vice versa it will be important that data that is supposed to be stored on a tag was not modified on its way to the tag.

As soon as networked readers, that are not under sole control of the application provider, are used in applications, we will need to check the authenticity of the data that should be stored on the tag. Authentication of tags to readers will be an important feature to protect IoT applications from data introduced by fake tags.To avoid illicit changes of the data on the tags, reader authentication will be necessary in every scenario where tags are reconfigured. Tags with signature functionality will allow to ensure the integrity of data originating from them.

2. Security for passive RFID tags

In the early years of passive RFID technology, security was not considered as important topic. Passive RFID tags were considered to replace bar codes for applications in the supply chain. While security measures for smart-card technology were applied on contact-less smart cards, the primary goal for design of RFID tags was cost saving, and therefore minimal functionality of the tags themselves. The advantages over bar codes, like improved reading distance, bulk reading and no need for direct line of sight, were addressed. The resulting security concerns (tag cloning, eavesdropping etc.) were not taken into consideration. This chapter will outline the development of the "security" topic in the context of RFID applications. We will focus on passive RFID technology for longer reading distance without discussing the development of contact-less smart card technology.

2.1. The privacy discussion

Soon after the presentation of first ideas for applications of RFID technology, an intense public discussion about the technology's privacy implications emerged. This public discussion was mainly driven by privacy activists and consumer advocates and was based on a lot of factoids. RFID industry ignored the discussion for a rather long time. They did not react on the statements but they claimed that RFID technology does not have any privacy implication, since RFID tags do not hold personal data. This statement is true but only for a very specific selection of applications. In the closed supply-chain scenario this argument is true as long as it is guaranteed that the tag is removed or killed when the item is handed over to the end consumer. Over years the discussion was held active, with arguments like RFID chips are "the mark of the beast" [11] from Biblical prophesies or that RFID tags can be read from satellites.

In fact, the ongoing discussion lead to a situation which turned out to be critical for development of the technology. One famous incident was the announcement of a hugh RFID project performed by Benetton. Philips Semiconductors published that Benetton was about to order a very high number of tags, to introduce item level tagging for improving their supply chain.The project was in a very early stage and far from introduction into the productive system, but vague ideas for use after the point of sale were already discussed (e.g. the RFID enabled washing machine). In reality, no item level tagging was considered in the first stage of the project, pallet level tagging was planned as a start.

In this scenario no privacy problems would appear for the end customers because the purchased goods would not carry RFID tags. Only when the first stage of the project was considered successful, item level tagging was considered later on. Directly after publication of the story by Philips, Benetton was flooded with a very high number of complaints and requests for statements, but they were not prepared for such a situation at this moment. No privacy impact evaluation or similar was done at this stage of the project. A organization called CASPIAN (Consumers Against Supermarket Privacy Invasion and Numbering) called for a boycott of Benetton products and got a very high appearance in press. Due to the high pressure, Benetton had to back off the project, and oviously also the order of RFID tags. So far the project was not implemented, or at least no public information about the results and reasons to stop the project are available. It is obvious that Benetton had to stop the project and since then abstains from RFID technology for the following years as consequence of the story.

In 2006 the European Commission started activity to solve the privacy discussion in RFID technology. A series of public RFID consultations was organized together with a platform for public discussion. During a series of workshops experts were invited to contribute to the topic.

In May 2009 the *Commission Recommendation on the implementation of privacy and data protection principles in applications supported by radio-frequency identification* was published. This document provides a basis for RFID industry to develop and implement applications and gives the end consumers the confidence that their privacy is not violated by the technology or the applications. According this recommendation, future RFID applications which involve processing of personal data require a privacy impact assessment (PIA) before roll-out. Requirements for this PIA are currently defined by all stakeholders in close cooperation with the European Commission. RFID industry has also reacted on the ongoing discussion.

2.2. Added value due to security services

Often security requirements for RFID systems are discussed as "non-functional requirement". This approach does not consider benefits to the systems functionality triggered by embedded security features.

We suggest to discuss security functionality as a service that enables new application areas for a system or that raises the value of the system due to its protection. This approach facilitates the justification of the additional costs arising by the protection. When the argumentation demonstrates benefits, the arising costs can be compared with the assumed arise of income. We think that a service oriented approach is necessary for proper development of secure RFID technology.

In the Internet the introduction of the ssl, or the secure version of http — shttp — enabled a multitude of new applications that were not possible before (online banking, eGovernment, eCommerce, ...). The same can happen as soon as RFID systems can provide a proper level of security.

In the following, we provide a non-exhaustive description of security services for RFID systems:

Proof of origin - anti-cloning: Tags with encryption functionality and a stored secret key can perform a challenge-response authentication protocol. Cryptographic authentication can be performed between tags and readers.

When such a tag is attached to a product this mechanism can be used as protection against product cloning. Companies can personalize the tags in their products with secret keys. An authentication server with public reading access can provide the necessary authentication data to customers or to any other party who intends to check the origin.

Protection against introduction of wrong or fake data into back-end services: When RFID systems are integrated into automated business or production processes, the data inserted into these systems is critical. As for any other part of the IT system, protection is necessary to avoid financial loss due to successful attacks.

In closed-loop RFID applications which operate inside the premises of a company, protection against introduction of illicit RFID tags, and wrong or fake data from those tags, can be achieved by physical protection of the area around RFID readers.

As soon as the system expands outside the company, attackers can easily introduce data via the unprotected tag-reader link with either faked RFID tags, or RFID tag spoofing devices. It is easy to understand that wrong data in IT-systems for production or other critical business processes can produce considerable damage. A possible attack got wide attention after its publication [15] under the buzzword RFID virus. RFID tags with cryptographic authentication feature can prevent such attacks without the need for physical protection of the area around the readers.

Data integrity protection for data from tags: In current RFID systems the transponders do not hold a lot of data, but their main functionality is to transmit their UID. Future RFID systems will include tags with increased memory; therefore readers will be store data on tags, or tags will generate data by themselves (e.g. tags with temperature sensor or light sensor).

When such tags are read and the data is relevant for the RFID application, it is important to check the integrity of the data coming from the tag. A very illustrative example is a temperature sensor tag which is used to record the permanent temperature characteristic of an object in a cold chain. In case that anybody can change the temperature logging data coming from the tag, the logging is useless. A sleazy operator of a could chain could easily alter the logging data to hide discontinuances under his responsibility. For food products this leads to inconveniences because the goods go bad earlier. For pharmaceutical or chemical products such an interruption can cause severe damage with relatively high compensation claims.

If data on tags is not protected, an attacker can harm the system by simply changing the logging data of correctly reposited goods, which would force a cold chain operator to dispose non-expired goods due to the forged logging data.

Access control for the tag's memory or commands: When data is written to tags, or the configuration of tags can be changed by a reader outside a trusted environment (e.g. execution of the "kill" command), it is necessary to protect from unwanted access.

In current tags the write commands or the execution of the kill command is protected by passwords (access password or kill password in Gen2 tags). This is only a weak protection due to the rather short length of the password (32 bit) and the possibility to eavesdrop and reuse the password.

Especially when write access to tags should be performed outside protected areas, another form of protection is necessary. A tag can request proper authentication of the reader before it grants access to critical memory areas or commands. If only authenticated and trusted readers get write permission to write data to the tag or to execute critical commands, a broad band of attacks is repelled.

In future open RFID systems it will be additionally necessary to hand over the control of the tag's content. When a tagged object changes its owner, is is also necessary to perform a so-called transfer of ownership of the access rights to the tag's memory. After the transfer, only the new owner can alter the tag's data and configuration. This can be performed by exchange of the cryptographic keys used in the reader authentication process.

Protected or encrypted transactions between reader and tag: To repel eavesdropping attacks on the wireless channel between tag and reader, the exchanged data can be encrypted. A pre-requirement for useful encryption is authentication, because the party who encrypt the data needs to decide which key should be used. Even if a session key for the encryption is generated by so called "key agreement schemes", both parties need to authenticate to avoid man-in-the-middle attacks.

If correctly established between authenticated parties, an encrypted channel between tag and readers prevents illicit tracking and tracing of RFID tags. No information about the content is revealed to an eavesdropper, but the exchanged data seems like a stream of random data.

Privacy protection for the owner of the tagged object is not the only motivation for encryption of RFID transactions. The data transferred between tags and readers during an inventory process can be a interesting target for industrial espionage.

2.3. Wrong assumptions for secure RFID tags

Some wrong assumptions about the feasibility of cryptographic primitives during the early years of the RFID hype delayed the development of proper protection mechanisms. In this section we provide some information about these wrong assumptions, we try to correct them and describe the effects that those assumptions had on the work of the research community.

First wrong assumption: Implementation of real crypto on tags is technically not possible. With the idea to use passive RFID tags as bar code replacement the first killer application for this technology was born. For this application, maximum reading distance and minimal tag costs are the major requirements. The tag's power consumption determines the reading distance, the area of the chip influences the costs of the tag.

When AUTO-ID labs came up with their vision of automated supply chains on basis of passive RFID tags in 1999, it was still a very challenging task to realize the basic functionality of the tags so that they still operate in the required reading distance. In the early years, it was therefore believed that it is technically not feasible to include crypto functionality on the tags without reduction of the maximal reading distance.

This assumption was arising from estimations by downsizing the power and area consumption of existing modules from smart card technology, to newer silicon production technologies. Different throughput requirements of RFID tags and smart card chips were not considered during this estimation.

The results of our research proofed this assumption wrong. In [4] the authors presented for the first time an RFID tag architecture with an integrated AES module. The crypto module fulfills all requirements for application on passive RFID tags in terms of power consumption and chip-area.

Based on the assumption that crypto modules for tags are not feasible, a lot of alternative solutions for protection measures on the NW-layer were proposed, which do not

require crypto functionality on the tags. Considering the expected high number of tags in RFID applications, such approaches get soon inefficient, due to the high amount of entries in distributed databases.

Second wrong assumption: Hash modules are less power and area consuming than encryption modules. When the first researchers started to accept that RFID tags might be able to compute cryptographic functionality a lot of proposals were made based on hash primitives. It was believed that hardware implementation of hash primitives are less resource consuming that encryption primitives.

For experts with background in power aware development and implementation of digital circuits in HW, it was obvious that hash primitives will consume more power and area in dedicated HW implementations, due to the required storing of rather long initial and chaining values. Although the combinational effort of hash algorithms is usually less than for encryption algorithms, the overall power and area consumption is higher. When the assumption was published in [18] the authors did not consider that storing intermediate values requires more power and area than low-power implementation of combinatorial logic.

The authors of [5] clearly show that "*current standards and state-of-the-art low-power implementation techniques favor the use of block ciphers . . . instead of hash functions . . . for secure RFID protocols.*"

Third wrong assumption: A tag with crypto results in a contact-less smart card. Contact-less smart cards can look very similar to RFID tags. The main difference between those devices are the operation distance. For most applications where contact-less smart cards are used, a very short reading distance with a well defined operation area is intended. For payment or access control applications it is required that only cards that are very close to a terminal are accepted.

The shorter reading distance leads to following differences. Firstly, the power available closer to a reader is by a factor of 1000 higher than in the maximum distance of passive RFID tags. While digital parts of contact-less smart cards consume currents around 20mA, the available power for digital circuits on RFID-tag is less than 0.015mA. Secondly, the possible number of devices in the reader field is very different.

Smart cards, which are used in security critical operations must fulfil very high requirements for implementation security. During a certification process it is evaluated that card's secret does not leak, even if a very high number of cryptographic operations are analyzed with side-channel analysis (SCA) techniques (some ten million samples are meanwhile standard). Typical RFID tags do not require to perform than many cryptographic operations in their lifetime, therefore the number of operations a tag performs with one key can be restricted which facilitates the design of SCA countermeasures.

Chip costs is a critical factor for both technologies, but RFID tags are much cheaper than smart cards. While smart cards chips have a size of around $10mm^2$, the silicon area of RFID tag chips is around $\frac{1}{2}mm^2$. Any similar additional component added to the basic functionality, influences the RFID tag's overall size — and consequently its costs — with a much higher percentage.

We think that future RFID tags will provide a very restricted and well chosen set of cryptographic functions, that results in a marginal controlling effort.

2.4. Security flaws in existing products

Already today RFID tags are used in security critical applications. In difference to our suggestion, most of this applications rely on proprietary cryptographic solutions. To achieve the power requirements for tags, dedicated algorithms that promised to result in a low power design were developed. In many approaches the used key length does not fulfill established cryptographic standards and therefore brute force attacks become feasible.

In the following we provide information about recently published attacks on RFID applications with secure tags.

Texas Instruments – DST: The DST can perform a cryptographic challenge-response protocol to authenticate tags to a reader. As underlying cryptographic primitive the proprietary custom block cipher DST-40 is used. Those tags were designed for use in an electronic car immobilizer system, with protected passive tags which are embedded in the car key.

At the time when the tags were designed, it was assumed that a 40-bit key provides sufficient protection against car-theft. Designers of the system claim, that at the time when the tags were developed, it was technically not possible to implement protection measures with longer bit length with the available silicon technology. In a later application the RFID tags from the car keys were used to authenticate clients at payment terminals of gas stations (*Speedpass by Exxon Mobil*).

In [1] a group of students and researchers from Johns Hopkins University present a hack which exploits severe vulnerabilities of the tag. They used information from a presentation held by one of the developers of DST-40 to reverse engineer the proprietary and confidential encryption primitive. Knowing the algorithm, they implemented an FPGA based key-search engine which is able to search the secret key for a given challenge-response pair within an hour. Once they know the secret key, they can copy it to a tag-emulation device to authenticate illicitely to the electronic car immobilizer system or to a payment terminal. While it is possible to check for duplicate tags in the online system of the payment application and to defeat this attack on this level, the electronic protection of the car key is completely broken by the attack. The researchers produced and published videos where they showed that they could make a purchase on an electronic terminal and start a car using their emulation device. The necessary budget for the devices used in the attack was below US $ 10.000.-

This incident was the first published attack on a protected RFID application. It received worldwide press coverage.

The attack shows that application of proprietary custom algorithms for protection is very critical. As soon as details about the implementation become public, attacks become probable.

NXP – The Mifare Incident: Mifare describes a series of contact-less smart card products from NXP. Different versions of Mifare with different security levels are available, the one we refer to here in this section is Mifare Classic, which uses the proprietary CRYPTO-1 algorithm as security primitive. Mifare was also licensed to other chip producers. Currently Mifare is the market leader and the de-facto standard for contact-less ticketing system. It was also discussed as protection of RFID applications.

The functionality of CRYPTO-1 was reverse engineered with home equipment by graduate students. During the Chaos Communication Congress in December 2007

they presented their results. Later on they published their analysis results as research paper[13]. During their work, not only the functionality of the proprietary and undisclosed algorithm was revealed, but also other severe weaknesses of the design were detected. Soon after disclosure of CRYPTO-1 a very efficient attack that reveals the secret key within some minutes followed.

At the same time a second academic team was investigating the security of Mifare cards. Soon after the first presentation of successful attacks on Mifare, researcher from Radbound University published a paper [2], where successful attacks on Mifare cards and their applications were presented. After these publications it was clear that the protection of Mifare Classic cards was completely broken.

It is out of doubt that Mifare products are a commercial success. When the successful hacks were published, newer versions with stronger, and above all, standardized cryptographic features were already available as products. Nevertheless, Mifare Classic cards are still used in many applications. The incidents shows again the risk of using proprietary cryptographic primitives. As soon as the secret algorithm was disclosed, successful attacks became feasible.

Keeloq – A successful attack using SCA: Keeloq is available as product for active (battery powered) and passive transponder devices. The main application area are battery powered (active) Keeloq transponders for remote key-less entry systems.

In early 2008 a cryptanalytic attack on the Keeloq encryption algorithm was presented [7]. This attack requires exhaustive computation but is technically feasible. The impact on practical implementations was considered minor due to the necessary computational effort of an attack.

Shortly after publication of the cryptanalytic attack, researchers from the University of Bochum analyzed several "high secure" key-less entry systems with Keeloq protection by application of SCA methods. This class of attacks uses physical characteristics of the encryption device, e.g. the power consumption of the device during operation, to reveal the secret key. After analysis of several products they completely broke the system [3]. In a follow-up paper they explain how to reveal the secret key of a remote device after eavesdropping only two cipher-texts from the device[10]. Additionally, this attack allows to prevent access for legitimate devices, while the illegal attacker can still enter. These attacks pose a serious threat for all installed systems which are protected by Keeloq.

Compared to the value that is protected by the devices, it is possible to break it with rather low effort. Interesting is that this attack relies on an attacking method that was not known when the system was developed.

Future Attacks: It is important to consider the previously described incidents for future developments. The life time of tags in the field is rather long, compared to software products. The academic community has interest in breaking RFID systems. There is no reason to assume that other protected RFID tag solutions provide a better protection. The design of currently available tags date back to a time when the necessary protection level for the applications was underestimated and when silicon technology for tags did not allow better protection.

2.5. Trade-offs for security implementation on tags

From the experiences in the traditional Internet we learn that secured communication will enable a variety of applications for new technology. We are confident that proper

protection of future RFID systems and the IoT requires also protection of passive RFID tags. In this section we discuss the possibilities to provide those protection functionality on future tags. Due to the high restrictions of resources like chip area and power supply, alternative trade-offs need to be taken to come up with proper solutions.

Computation time - clock cycles for computation: Often, tags are quite long in the field. In some applications tags need to answer rather fast to a initial request of the reader (inventory command), but this is not the general case. For cryptographic operations we can assume that only a very limited amount of data will be encrypted by the tags. Consequently, the required throughput for the cryptographic circuit is quite low; a lot of clock cycles can be used for for the operation. Traditionally, throughput optimization is the reason to embed cryptographic hardware primitives, therefore the hardware designers have to re-think their approaches when designing for RFID tags.

Often the area and power consumption of circuits can be significantly improved when all available time for the required computation is used. Parallel execution or pipelining are typical strategies that are applied to improve throughput of crypto HW. Both strategies are counter productive when designing for low area and low power consumption.

Clock frequency: The used clock frequency is a critical factor for the power consumption of the digital circuit. The lower the clock frequency, the lower is the overall consumption of the circuit since the CMOS structures use most of the energy at the moments when the clock signal changes. Different clock domains can be defined for the overall RFID tag. Separating clock domains of circuits with high computational activity from those of register files can help to meet the limits for the overall power consumption. Nevertheless, separation of clock-domains requires additional effort for synchronization and clock generation, which adds up to the overall chip area.

Silicon area is becoming less critical: Production technology for silicon chips is steadily improving. Newer production processes allow smaller on-chip structures leading to lower area- and power consumption for the same functionality. Production on newer process facilities is more expensive, but the expected high number of tag chips justifies to migrate to newer production technologies.

The chip alone does not yet make up the tag; the single tag-chips need to be cut from the die and every single chip is then mounted to the package with the antenna. The smaller the chips get, the more expensive is the handling of them before and during the connection with the packages and antennae. The percentage of area loss during the cutting of the tags from the die increase, because the active chip area shrinks, but the loss from cutting stays constant. Recent estimations forecast that a chip area around $\frac{1}{2}mm^2$ results in minimum overall costs. Currently, tag chips are just below $1mm^2$, with the next shift to a newer technology the standard functionality will be possible on a silicon area close to this value. Making the chips even smaller would then result in higher effort during handling costs and increased loss during cutting.

For improving the power consumption, moving to newer technologies still makes sense. This development will than lead to a situation that additional functionality can be implemented without resulting cost increase during production. The area overhead due to cryptographic functionality will then be less critical.

Pre-computation: When a tag enters a reader field it is automatically powered by the EM-field. As completely passive device it stays idle until the reader sends a request. Whenever multiple tags are in the field and the reader detects a collision during the

time when the tags are supposed to answer, it deselects a certain number of tags so that the probability of a further collision in the next try is reduced. This operation is repeated until all collisions are resolved. For further operation with one specific tag, a single tag is selected by direct addressing. All other tags are then disabled. While this procedure can be rather intense for the reader, the tags themselves are mainly idle, actively operating only in a rather short time interval. This idle time could be used for cryptographic operations. Random numbers can be derived from a pre-stored seed value with a pseudo-random number generator. Pre-computations can be done for computation intense operations. Clearly, the protocols and algorithms need to be designed in a way so that this pre-computation is possible. So far this was not considered during development and design of RFID protocols.

Computation charging and split computing: RFID tags are powered by the surrounding EM field from the reader. The reader additionally performs modulation of the carrier to communicate with the tags. The tag's answers are de-modulated by the reader. Whenever a tag receives the carrier signal with enough field-strength it is powered up an waits for communication started by the reader. In principle it is possible that a tag starts computation without trigger from outside. Instead of starting computation intense operations after a request from the reader, the tag can pre-compute intermediate values before getting the input value. Clearly, the used protocol and underlying primitives need to support this feature. In case that protocols allow pre-computation computation charging stations can be designed on the application level. Those stations would be reduced readers, that only provide a RF-field in situations where tags are placed stationary for a longer time (e.g. in warehouses). The tags start computing new coupons when they are supplied with the EM field, and they store the new generated coupons when they have finished computing. More advanced tags could even store intermediate values of computation intense operations so that they do not have to start the whole operation when they get interrupted before finishing (split computation).

Acknowledgements

The opinion documented in this paper is based on ongoing research activity in the area of RFID security at IAIK, TU Graz. Many arguments are based on the results end experience gained in the research projects ART (Authentication for long-range RFID Technology) and SNAP (Secure NFC Applications). Currently, we are working on integration of asymmetric crypto on tags in the project CRYTPA (Cryptographic Protected Tags for new RFID Applications). All those project were sponsored by the Austrian funding programme FIT-IT. On international level, we are researching in funded research projects of the European Commission, e.g. BRIDGE, C@R or SMEPP (all funded under FP6).

References

[1] S. Bono, M. Green, A. Stubblefield, A. Juels, A. Rubin, and M. Szydlo. Security Analysis of a Cryptographically-Enabled RFID Device. In *USENIX Security Symposium, Baltimore, Maryland, USA, July-August, 2005, Proceedings*, pages 1–16. USENIX, 2005.

[2] G. de Koning Gans, J.-H. Hoepman, and F. Garcia. A practical attack on the MIFARE Classic. In *8th Smart Card Research and Advanced Application Workshop (CARDIS 2008)*, volume 5189 of *Lect. Notes Comp. Sci.*, pages 267–282. Springer, 2008.

[3] T. Eisenbarth, T. Kasper, A. Moradi, C. Paar, M. Salmasizadeh, and M. T. M. Shalmani. On the power of power analysis in the real world: A complete break of the keeloqcode hopping scheme. In *CRYPTO*, pages 203–220, 2008.

[4] M. Feldhofer, S. Dominikus, and J. Wolkerstorfer. Strong Authentication for RFID Systems using the AES Algorithm. In M. Joye and J.-J. Quisquater, editors, *Cryptographic Hardware and Embedded Systems – CHES 2004, 6th International Workshop, Cambridge, MA, USA, August 11-13, 2004, Proceedings*, volume 3156 of *Lecture Notes in Computer Science*, pages 357–370. Springer, August 2004.

[5] M. Feldhofer and C. Rechberger. A Case Against Currently Used Hash Functions in RFID Protocols. In S. Dominikus, editor, *Workshop on RFID Security 2006 (RFIDSec06), July 12-14, Graz, Austria*, pages 109–122, July 2006.

[6] F. D. Garcia, G. Koning Gans, R. Muijrers, P. Rossum, R. Verdult, R. W. Schreur, and B. Jacobs. Dismantling mifare classic. In *ESORICS '08: Proceedings of the 13th European Symposium on Research in Computer Security*, pages 97–114, Berlin, Heidelberg, 2008. Springer-Verlag.

[7] S. Indesteege, N. Keller, O. Dunkelman, E. Biham, and B. Preneel. A practical attack on keeloq. In N. P. Smart, editor, *EUROCRYPT*, volume 4965 of *Lecture Notes in Computer Science*, pages 1–18. Springer, 2008.

[8] A. Juels, R. L. Rivest, and M. Szydlo. The Blocker Tag: Selective Blocking of RFID Tags for Consumer Privacy. In *10th ACM Conference on Computer and Communication Security, Washington, DC, USA, October 27-30, 2003, Proceedings*, pages 103–111. ACM Press, October 2003.

[9] U. Kaiser. *RFID Security, Techniques Protocols and System-on-Chip Design*, chapter Digital Signal Transponder, pages 177–190. Springer, 2008.

[10] M. Kasper, T. Kasper, A. Moradi, and C. Paar. Breaking keeloq in a flash: On extracting keys at lightning speed. In B. Preneel, editor, *Progress in Cryptology - AFRICACRYPT 2009, Second International Conference on Cryptology in Africa, Gammarth, Tunisia, June 21-25, 2009. Proceedings*, volume 5580 of *Lecture Notes in Computer Science*, pages 403–420. Springer, 2009.

[11] L. M. Katherine Albrecht. *The Spychips Threat: Why Christians Should Resist RFID and Electronic Surveillance*. Nelson Current, 2006.

[12] D. Molnar. Security and Privacy in Two RFID Deployments, With New Methods For Private Authentication and RFID Pseudonyms. Master's thesis, University of California Berkeley, 2006.

[13] K. Nohl, D. Evans, S. Starbug, and H. Plötz. Reverse-engineering a cryptographic rfid tag. In *SS'08: Proceedings of the 17th conference on Security symposium*, pages 185–193, Berkeley, CA, USA, 2008. USENIX Association.

[14] NXP Austria GmbH. Website mifare.net - contactless smart cards. http://www.mifare.net.

[15] M. R. Rieback, B. Crispo, and A. S. Tanenbaum. Is Your Cat Infected with a Computer Virus? In *4th Anual IEEE International Conference on Pervasive Computing and Communications – PerCom 2006, Pisa, Italy, 13-17 March, 2006, Proceedings*, pages 169–179. IEEE Computer Society, March 2006.

[16] D. L. B. . K. A. Sanjay Sarma. White paper: The networked physical world. MIT-AUTOID-WH-001.pdf, 10 2000.

[17] S. Sarma. White paper: Towards the 5 cent tag. http://www.autoidlabs.org/singleview/dir/article/6/197/page.html, 11 2001.

[18] S. E. Sarma, S. A. Weis, and D. W. Engels. RFID Systems and Security and Privacy Implications. In B. S. K. Jr., Çetin Kaya Koç, and C. Paar, editors, *Cryptographic Hardware and Embedded Systems – CHES 2002, 4th International Workshop, Redwood Shores, CA, USA, August 13-15, 2002, Revised Papers*, volume 2523 of *Lecture Notes in Computer Science*, pages 454–470. Springer, August 2

Radio Frequency Identification System Security
Y. Li and J. Zhou (Eds.)
IOS Press, 2010

doi:10.3233/978-1-60750-485-6-125

Securing RFID-supported Supply Chains

Florian KERSCHBAUM [a] and Manfred AIGNER [b]
[a] *SAP Research, Karlsruhe, Germany*
[b] *IAIK, Graz University of Technology, Austria*

Abstract. Modern RFID-supported supply chains envision a seamless sharing of item-level data across multiple supply chain participants in the "Internet of Things". However, many companies are reluctant to propagate large amounts of their track and trace information to others, as they fear the uncontrolled disclosure of vital business intelligence. Without built-in safeguards, such systems thus run the risk of hindering the adoption of efficient supply chain management infrastructures.

In this paper we will define the cornerstones of a cryptographically sound security architecture for RFID-supported supply chains that will enable efficient logistical management with minimal data disclosure. We propose to replace the common centralized track and trace approach with an architecture that makes use of strong cryptographic primitives and secure storage on the tag and builds on top of those enhanced authentication and key-agreement protocols. The architecture will thus span the entire technology range from the RFID tag and its network infrastructure to the back-end system that is storing the supply chain information.

Keywords. RFID, Cryptography, Supply Chain Management, Track and Trace

1. Introduction

RFID is becoming a prevalent technology in supply chains. In order to gain the full benefit of this technology companies must share item-level reading data, so called events. A set of standards is emerging for gathering and sharing events across the Internet: the EPCglobal network. This future standard will allow the discovery of events without any security constraints, such that it is possible through repetitive querying to obtain the basic information of organization, time and identifier of any event. Additional event data is supposed to be protected by (role-based) access control, but traditional access control faces several problems related to item-level event data. First, the principals of access control are not always known to the protecting parties, e.g. if they are separated by two stages in the supply chain, and second, each item needs an individual access control policy even in the case of role-based access control, such that the sheer number of policies becomes unmanageable. These unresolved security and privacy issues lead to a reluctance of companies to share data [12,14]. According to a recent study by the University of Freiburg, 29% out of 102 RFID industry users consider unresolved security issues to be a problem ("high" and "relatively high" importance). A further 32% of the companies state that they face serious privacy concerns among customers. Both findings are particularly relevant since 41 out of 106 identified in-house RFID applications are also potentially suitable for cross-company use (such as Material Flow Control, Kanban, Anti-theft Protection, Maintenance etc.) [15].

In this paper we attempt defining an architecture that integrates the real world of the "Internet of Things" in a supply chain with the security objectives of the players. In particular we address the most pressing security concerns of

- *confidentiality*: as already mentioned companies want to reveal data only selectively, remain in control of the access decision and base the decision on sound identification of actors. This concerns are generally address by the security technologies of access control model (what to disclose), access control mechanism (how to enforce) and authentication. In this paper we outline a general model of access control in supply chains, a novel authentication mechanism and rely on distributed database with locally enforced access control.
- *integrity*: decision will increasingly based on supply chain data. Imagine verifying the authenticity of a prescription drug by retrieving the pedigree information in the supply chain of that particular item. If the supply chain data can be tampered with, a competitor may be able to prevent sales of a specific drug. This problem is particularly challenging, since the information is distributed across a number of parties. In this paper we propose a mechanism that ensures the integrity and authenticity of supply chain event data based on the use of enhanced RFID tags. This information can then later also be used to ensure confidentiality.

The remainder of the paper is structured as follows. In Section 2 we outline the envisioned architecture with its structural components and considered applications. Section 3 lists the properties we intend to ensure and achieve by this architecture. In Section 4 we describe the principles and mechanism that can be used to implement the architecture and sketch some initial protocols. Our conclusions from the work so far and a number of possible avenues for future work are listed in Section 5.

2. Architecture

The basic architecture is depicted in Figure 1. Goods equipped with RFID tags pass from the supplier to the buyer. Each company reads the RFID tags and stores related information, an event, in its local database. Later the companies exchange that information in order to run advanced applications. Some applications include:

- *Estimated Time of Arrival*: Based on the events of suppliers the buyer estimates the time of arrival and eventually triggers correction actions.
- *Product Recalls*: Stored events can help with targeted recalls limited to the minimal set of products that needs to be recalled.
- *Benchmarking*: Item-level events allow for the first time precise, supply-chain wide, inter-organizational benchmarks [6] to be computed that could not be computed before, e.g. percentage of returned items per supplier [3].
- *Anti Counterfeiting*: Supply-chain wide tracing allows the identification of counterfeit products and their identification at the point of sale.

Our future security architecture will consist of three interrelated pieces. Our centrepiece is a cryptographically verifiable event stored in a distributed database. If the information collected in an event and obtained from the tag, the company and the environment is authenticated and verifiable by a third party, it can then be used to secure the data exchange, e.g. as basis for a key exchange.

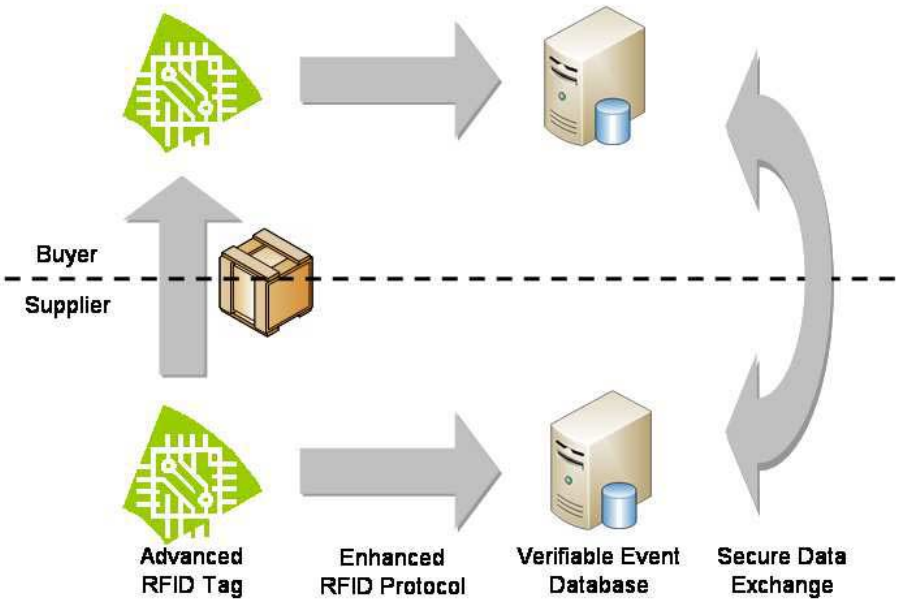


Figure 1. Basic Architecture of Track and Trace

Advanced RFID Tags: In order to guarantee end-to-end security we will need to integrate novel computational capabilities into RFID tags. These include implementations of the cryptographic functionality and new architectures of tag controllers that enable secure integration of additional modules and sensors. Those future tags will implement advanced security services to be used in applications that require signatures by objects, or integrity of sensor data. The security services will also be used to perform secure handover of tags without completely deactivating the tag by the kill command.

Enhanced RFID Protocols: Future RFID communication protocols need to secure communication between passive tags. This will include authenticating the information from the tag and making it usable in subsequent protocols, such as key-agreement protocols or cryptographically enforced access control.

Secure Data Exchange: The information systems at the back-end that collect RFID data need to be interconnected and offer the necessary services for performing the intended applications. We need to solve a novel authentication problem in supply chain back-ends. One has to prove the identity of the company in conjunction with the identity of the item. A recent development is RFID-based authentication and key agreement [11] in which information is passed along the supply chain and then later used by the companies to establish secure session keys for exchanging tracking data.

3. Challenges

The following properties should guide the design of the architecture:

- *Secure*: A formal assessment of the architecture and its components with stated trust assumptions is necessary. It should clearly derive the provided security guarantees proven security.
- *End-to-End*: The architecture spans multiple layers and multiple applications and technologies at each layer, but it should ensure the security and privacy of the data from the gathering at the device to the use within the application. It is therefore necessary that the components are compositional and integrated as well.
- *Flexible*: The architecture needs to be able to cater for different trust levels and apply different security mechanism depending on the business needs.
- *Decentralized*: The architecture should minimize the use of centralized systems, such as trusted third parties. Instead each party should remain independent and in control of its data.

4. A Simple Approach to Cryptographically Verifiable Events

4.1. Enhancing Tag Capabilities with Public-Key Crypto

Due to steady advances of silicon technology, the computational capabilities of tags are steadily rising. The minimal die area of tag chips is limited due to the fact that smaller chips produce higher costs during handling, packaging, and cutting. With current technology the basic functionality of Gen2-tags can be implemented on an area that is close to this limit. Migration to newer technology still makes sense, since the power consumption of the tags (and therefore their reading distance) can be improved by smaller silicon structures. This means, that future tags will provide additional area to implement more functionality without adding additional costs. This area can be used for implementation of cryptographic functionality. The feasibility of symmetric cryptographic operation with state-of-the-art security level is demonstrated in [4]. Newer publications show that it is even possible to implement public-key algorithms under the given requirements for power consumption, area consumption, and throughput [7] [2].

Future RFID tags will differ from contact-less smart cards by their reading distance and the set of cryptographic features they provide. While smart-cards typically provide a rather powerful selection of different cryptographic features, the capabilities of RFID tags will be very restricted to a small set of cryptographic functions. This restriction is necessary due to the available power budget, but also due to the rather high controlling effort that comes with combination of different cryptographic algorithms. Execution of different cryptographic procedures using their associated credentials in a way that security holes are avoided, requires rather complex controlling that will not be achievable on RFID tags in next future.

The execution of a cryptographic algorithm alone does not yet make up a security token that can be used in the proposed system. A private key needs to be stored in a secure way so that is available for the cryptographic operation, but not for an attacker. This memory-area requires secure access control for storage of key-material to avoid illicit access during personalization or key-exchange.

Side-channel analysis (SCA) poses a serious threat which requires additional protection [13]. In the suggested application scenario, the number of executions of the cryptographic operations with one specific key can be limited to a reasonable value, therefore the countermeasures against SCA can be scaled in a effective manner. Additionally to the private key, the tags need to store their public key in form of a certificate that ensures the binding between cryptographic key and the tag's public ID. This certificate needs to be transmitted to a reader before an authentication procedure can take place.

To execute the cryptographic operations additional commands need to be integrated into the tag to reader protocol. Currently, security standardization for RFID protocols is ongoing within ISO. They follow a service-oriented approach which allows a tag to offer available security services to a reader. The reader can decide to use the tag with its capabilities in a secure application or not. Uncritical operations are still possible, even if a tag (or a reader) does not offer advanced services. We expect that successful security standardization, together with foreseeable development of chip technology, paves the way for adoption of public-key crypto tags in commercial wide-scale applications.

4.2. Context-Based RFID Authentication

Existing RFID authentication protocol can be used to securely and privately identify RFID tags. They are secure, because they ensure that only a tag that possesses a secret identifier can successfully authenticate. They are private, because they do not reveal that identifier to rogue readers who are not already in possession of that identifier.

When using RFID in supply chains, neither property is overly important. RFID tags per se provide little resistance to physical cloning and therefore are infrequently used for product authentication. Other mechanisms that can rely on cheaper tags using the collected trace data can be used. Goods in supply chains are not tied to personal data, such that privacy is of little relevance. This changes once the good has been delivered to the customer, but different mechanisms exist for dealing with this problem including the unpopular *kill* function.

The prevalent approach for handling RFID data in supply chains are so-called events. At its very basic an event is a triple $\langle object, time, location \rangle$ stored each time an RFID tag is read.

The $object$ identifier is stored on the RFID tag and the secret information used in existing RFID authentication protocols. We can therefore use existing RFID authentication protocols to effectively ensure the confidentiality and integrity of this information. The question is how do we ensure the confidentiality and integrity of the entire event?

Given RFID tags with public-key cryptographic capabilities there is a very simple secure (non-private) RFID authentication protocol. The reader simply sends a challenge r to the tag which responds with its signature. Assuming a public-key infrastructure for the RFID tags, one can verify the identity of the tag. Obviously this simple signature verification protocol must be extended for practical use, but we will use its principle throughout this section.

The $location$ identifier can have different degrees of granularity. At a very coarse-granular level it can be just the identifier of the company handling the item. At a very fine-granular level it can be the identifier of the reader. An RFID authentication protocol that also supports reader authentication could be suitable for ensuring integrity of the pair of event data.

Unfortunately this brings along with it a major access control management and key management problem. The tag needs to decide to have a notion of which reader is allowed to read it and this notion must change as it proceeds through the supply chain. Furthermore there are certain limitations to the security any context-based RFID authentication protocol can provide without the use of physical security. As in many uses of trusted hardware, the beneficiaries of the use are not the actual users, such that its acceptance may be low.

Instead we propose to use the company identity as location in our events. Each company – similar to each RFID tag – has its own public-, private-key pair. A challenge is issued to the company which can then prove its identity by signing the challenge.

Given secure (tamper-proof) hardware one can try to generate reliable locations even at finer granularity. Secure hardware equipped with localization technology, such as GPS, can be used to verify its location if it is mobile. In other cases, the hardware can be installed permanently provide other means of authentication. The secure hardware could even be an RFID tag itself and use the same type of authentication as above. Of course, one then needs to ensure that the tag is not removable in addition to being tamperproof. Yoking proofs [8] can provide the assurance that both tags have been read together.

The third piece of information in event is the $time$. This is a global time and we assume synchronized clocks in order to rely on this time information. A trusted source of time might be a time server that issues signed timestamps. This timestamp can then be used in the stored event.

Our enhanced, context-based RFID protocols need to integrate confidentiality and integrity of all three pieces of information in an RFID event.

4.2.1. Tying The Pieces Together

So far we have been able to attest the integrity of each individual piece of information in an RFID read event, but our goal must be to ensure the integrity of the event triple. We will outline a simple technique here, that can be used to tie pieces.

Recall, each party – RFID tag, reader (company) and time server – receive a challenge and return it signed. The basic idea is to have each party issue and sign the challenge for the next party.

We start an RFID read event by contacting the time server T. It will issue a signature $S_T(time, r_T)$ where r_T is a fresh challenge issued by the time server.

Once the company is in possession of the item and the RFID tag, it can then issue this challenge r_T to the RFID tag R.

The RFID tag will respond with a signature $S_R(r_T, r_R)$ where again r_R is a fresh challenge, but this time issued by the RFID tag.

This challenge r_R is finally signed by the company C. One obvious attack remains, since the company can request the timestamp from the time server early and then delay processing. But we can use the same technique in order to have the time server sign a challenge by the company. The company produces $S_C(r_R, r_C)$ and sends r_C to the time server. The time server responds with $S_T(time', r_C)$. The time the RFID tag has been read is now bound between $time$ and $time'$.

4.2.2. Limitations

Our protocols follow the security model of distributed systems, i.e. there are n distinct parties. Collusion is an attack no protocol, even given physical security, can prevent from.

Assume an attacker controls parties A and B. He can always create an RFID read event for locations in A while the item is physically in a location of B. Imagine a device that simply relays signals from the reader over the network to a remote RFID tag. This device could trick even a trusted reader into creating a read event for a remote item. More simply the attacker could ship the item for A to B, but also no protocol can prevent relaying messages between A and B, such that an attacker can simply perform the attack on the RFID authentication protocol. This implies that the location of an event can only be as precise as the sphere of control which is our reason for choosing the company identity as granularity for the location. We assume that companies are less inclined to collude.

4.3. Authentication Using Verifiable Events

Each party stores its RFID events in a database. In order to perform the applications mentioned above the parties need to exchange the event data. This process generates "supply chain visibility". Nevertheless, the gathered data reveals sensitive information about a company's operation. Companies are therefore reluctant to share this data. Fine-granular access control may help to mitigate the problem.

The access control matrix for event data consists of events associated with an item times the supply chain partners. This access control matrix can become huge, since the number of items is continuously growing. Manageability is therefore key to the database owners.

Access control is usually performed on identity or via indirection on roles. Given that a party has access to all events or type of events, e.g. for a specific product, based on its identity there are many possible inferences. In particular a company can spy on its competition sourcing from or delivering to the same partner. Therefore a company should set its access control specific only to the items shared with a partner.

In order to authenticate for access to data based on shared items one needs to prove possession of an item. The advantage of our context-based RFID authentication protocol is that it produces a verifiable event. The integrity of such an event is ensured, such that other parties can verify and trust its correctness. Then the event can be used in order to authenticate. To request access to event data for item h, a party A simply presents a verifiable event $\langle h, time, A\rangle$ and proves its identity. The queried party can then grant access to data for item h only.

5. Conclusions and Future Work

Our proposed architecture fulfils its set goals of confidentiality and integrity at the very least to the minimal extent necessary. We provide a stronger access control using verifiable events which is enforced locally for each part of the distributed database. The integrity of each event is ensured using cryptographic mechanisms.

Our architecture is therefore *secure*. It involves all components from the tag to the item to the server hosting the database and is therefore *end-to-end* within the supply chain. Each party hosts its own data and entire system is *distributed*.

Although the access control policies may enable a large degree of flexibility, they are limited by the basic principle of access control of the decision to disclose or not to disclose and their enforcement mechanism. In order to increase the flexibility of the

architecture and cater for a wider range of use cases we propose a number of avenues for future research.

5.1. Releasing Aggregate Information

The design of supply chain information with events as building blocks does not consider the information protection needs and desires of the supply chain decision makers. It is entirely unclear the information contained within an event or set of events, e.g. by inference even with previous knowledge [18]. So how is a company supposed to make a decision whether to reveal that information or not? In many cases, it is nevertheless possible to decide on releasing aggregate information; in particular if that information, is necessary or derivable from an application. Imagine only releasing the bit whether a product has been recalled or not. If a company implements recalls – and it might be obliged to by law –, this information is entailed within the application. A decision maker can then easily compare the risks of disclosure with the expected business benefit.

The technical challenge is in implementing applications that are capable of enforcing this type of access control on aggregate (or computed) information. Given a central database this could be enforced by an application tier between user and database (similar to current enterprise systems), but in the supply chain scenario the data is distributed. Now a number of parties each unwilling to disclose its information first has to collaborate in this application.

Secure Multi-Party Computation (SMC) [1,5,17] offers a solution from cryptography. In SMC a number of parties compute a function on joint input, such that no party learns anything about the input of the other parties, but each party learns the output. Completeness theorems prove that this can be done for any function. Nevertheless these general constructions are prohibitively slow, such that researchers are developing special solutions since almost two decades. Selected applications in RFID-supported supply chains, such as batch recalls [16] and computation of key performance indicators [10], have already been proposed, but future research for more general concepts is necessary.

When using SMC new security challenges arise: nobody can verify the correctness of input data of other parties, since it remains confidential. In this way confidentiality and integrity become opposing objectives. An interesting new feature would be to integrate our proposed verifiable events into SMC, such that other parties could verify the authenticity of events without disclosing them. This could be implemented as a SMC itself or as a Zero-Knowledge-Proof. Also anonymous credentials may offer many mechanisms that can help.

5.2. Remote Enforcement of Access Control

Our proposed architecture is centered around the principle of distributed databases. In many systems RFID events are processed as streams, e.g. in publish-subscribe networks. In a publish-subscribe network data sources send events – as soon as they occur – to data sinks. In these cases one cannot rely on local enforcement of access control any more, but the data is disseminated within the network once it has been released by the source and may sooner or later reach any participant. One approach to handle this situation is to encrypt the data and only selectively release the key. This is called cryptographically enforced access control.

Attribute-based encryption [9] offers a mechanism to handle the keys, but it relies on a central key distribution center. Instead it would be necessary to distribute the keys between the supply chain partners (and the items), such that each party can only reveal data for its RIFD tags.

This leads to yet another interesting concept. Assume a central storage of information, e.g. with product information and history, and a physical object equipped with an RFID tag, i.e. the item in the supply chain. Can we control access to the central repository using the RFID tag, such that it is ensured that only physical possession of the RFID tag grants access? In this case the RFID tag could be used a physical token that is passed around for controlling access. Our enhanced RFID tags using public-key cryptography can help again, since they can respond to a challenge by signing it. The central repository can verify the signature and the freshness of the challenge and then grant access. This could also be a novel way for remote customer service where the customer even remains anonymous.

References

[1] M. Ben-Or, S. Goldwasser, and A. Wigderson. Completeness theorems for non-cryptographic fault-tolerant distributed computation. *Proceedings of the 20th annual ACM symposium on Theory of computing*, 1988.

[2] H. Bock, M. Braun, M. Dichtl, E. Hess, J. Heyszl, W. Kargl, H. Koroschetz, B. Meyer, and H. Seuschek. A Milestone Towards RFID Products Offering Asymmetric Authentication Based on Elliptic Curve Cryptography. Invited talk at RFIDsec 2008, July 2008.

[3] S. Chopra, and M. Sodhi. Looking for the Bang from the RFID Buck. *Supply Chain Management Review*. Available at http://www.scmr.com/article/CA6444375.html, 2007.

[4] M. Feldhofer, J. Wolkerstorfer, and V. Rijmen. AES Implementation on a Grain of Sand. *IEE Proceedings on Information Security*, 152(1):13–20, October 2005.

[5] O. Goldreich, S. Micali, and A. Wigderson. How to play any mental game. *Proceedings of the 19th ACM Symposium on Theory of Computing*, 1987.

[6] W. Hedgepeth. RFID Metrics. *CRC Press*, 2007.

[7] D. Hein, J. Wolkerstorfer, , and N. Felber. ECC is Ready for RFID Ű- A Proof in Silicon. In *Workshop on RFID Security 2008 (RFIDsec08)*, July 2008.

[8] A. Juels. "Yoking-Proofs" for RFID Tags. *Proceedings of the 1st International Workshop on Pervasive Computing and Communication Security*, 2004.

[9] A. Juels, and M. Szydlo. Attribute-Based Encryption: Using Identity-Based Encryption for Access Control. Manuscript, 2004.

[10] F. Kerschbaum, N. Oertel, and L. Weiss Ferreira Chaves. Privacy-Preserving Computation of Benchmarks on Item-Level Data Using RFID. *Proceedings of the 3rd ACM Conference on Wireless Network Security*, 2010.

[11] F. Kerschbaum, and A. Sorniotti. RFID-Based Supply Chain Partner Authentication and Key Agreement. *Proceedings of the 2nd ACM Conference on Wireless Network Security*, 2009.

[12] C. Kuerschner, F. Thiesse, and E. Fleisch. An analysis of data-on-tag concepts in manufacturing. *Proceedings of the 3rd Konferenz Ubiquitäre und Mobile Informationssysteme*, 2008.

[13] T. Plos. Susceptibility of UHF RFID Tags to Electromagnetic Analysis. In T. Malkin, editor, *Topics in Cryptology - CT-RSA 2008, The Cryptographers' Track at the RSA Conference 2008, San Francisco, CA, USA, April 8-11, 2008, Proceedings*, volume 4964 of *Lecture Notes in Computer Science*, pages 288–300. Springer, April 2008.

[14] B. Santos, and L. Smith. RFID in the Supply Chain: Panacea or Pandora's Box? *Communications of the ACM 51(10)*, 2008.

[15] J. Strüker, D. Gille, and Titus Faupel. RFID-Report 2008 – Optimizing Business Processes in Germany. *IIG-Telematik, Albert-Ludwigs-University Freiburg, VDI Nachrichten*, 2008.

[16] . L. Weiss Ferreira Chaves, and F. Kerschbaum. Industrial Privacy in RFID-based Batch Recalls. *Proceedings of the IEEE International Workshop on Security and Privacy in Enterprise Computing*, 2008.

[17] A. Yao. Protocols for Secure Computations. *Proceedings of the IEEE Symposium on Foundations of Computer Science*, 1982.

[18] D. Zanetti, and S. Capkun. Protecting Sensitive Business Information While Sharing Serial-Level Data. *Proceedings of the IEEE International Workshop on Security and Privacy in Enterprise Computing*, 2008.

Radio Frequency Identification System Security
Y. Li and J. Zhou (Eds.)
IOS Press, 2010

doi:10.3233/978-1-60750-485-6-135

On Mitigating Covert Channels in RFID-Enabled Supply Chains

Kirti Chawla[1], Gabriel Robins, and Westley Weimer

Department of Computer Science
University of Virginia, Charlottesville, VA 22904, USA
{kirti, robins, weimer}@cs.virginia.edu

Abstract. In a competitive business environment, RFID technology can help a business to optimize its supply chain. However, it may also enable an adversary using covert channels to surreptitiously learn sensitive information about the supply chain of a target business. We argue that the tracking of tags and the compromising of readers can create covert channels in the supply chain and cause detrimental market effects. To mitigate such attacks, we propose a framework that enables a business to monitor its supply chain in a fine-grained manner. We model the supply chain as a network flow graph, select key nodes to verify the tag flow, and actively search for covert channels. We note that optimal checkpoint node selection is NP-Complete, propose node selection and flow verification heuristics with various tradeoffs, and discuss appropriate countermeasures against covert channels detected in the supply chain. These practical methods can be implemented economically using current RFID technology.

Keywords. RFID, Covert Channel, Supply Chain, Network Flow

1. Introduction

Radio Frequency Identification (RFID) enables items to be tracked via attached tags, which respond to radio fields emitted by readers in their vicinity. A business can use this technology to make its internal processes more efficient and optimize its supply chain. RFID technology can streamline all phases of the production cycle, including pre-production activities, asset management, inventory control, production tracking, shipping, recalls and warranty authorization [1, 2].

However, the pervasive nature of RFID technology can also help adversaries glean sensitive information about the internal processes of a target business [6]. An adversary can track and/or modify existing tags, inject duplicate tags into an existing item flow, and compromise RFID readers in the supply chain of a target business. We say that such activities "*taint*" the flow of the information, and constitute covert channels in the supply chain of a target business [8]. These covert channels can surreptitiously reveal item flow patterns, including segregation, assimilation sites, site-specific inventory, delivery schedules, and other valuable sensitive information. An adversary can use this illicitly obtained information to gain an unfair (and not necessarily even illegal)

[1] Corresponding Author. This research is supported in part by a U.S. National Science Foundation grant CNS-0716635 (PI: Gabriel Robins).

marketplace advantage over a target business. Given such possible threats, it is important for a target business to verify the information flow in a fine-grained manner in order to detect the presence of covert channels and mitigate their effect.

Our contributions towards these goals are as follows. We analyze the threat sources in an RFID-enabled supply chain by enumerating four representative (but not exhaustive) attacks which an adversary can use to track the supply chain of a target business. We consider, both qualitatively and also using simulations, the ability of such attacks to affect market change. We model supply chains using network flow graphs, where nodes represent the sites and edges model the flow of items among sites. We select key nodes of the supply chain flow graph to verify the information flow. We call these selected nodes the "*taint checkpoints*", and refer to the process of their optimal selection as the "*taint-check cover*" problem, which we note is NP-Complete.

We propose taint-check cover heuristics based on various tradeoffs, such as the number of desired taint checkpoints. We propose verification algorithms that verify the flow of information, both locally and globally, in the supply chain. Our algorithms provide user-controlled tradeoffs between the strength of the verification results versus the time required to compute them. This enables post-detection actions to be taken by the target business either at a local site or along global paths. Finally, we evaluate our algorithms using a supply chain simulator, and provide a set of remedies that a target business can utilize to mitigate the effect of the discovered covert channels.

This paper is organized as follows. In section 2 we present the threat model chosen to analyze the RFID-enabled supply chain, and enumerate four possible attacks on such supply chains. In section 3 we describe some likely market change scenarios as a direct outcome of possible attacks. We discuss potential candidate models for supply chains and propose using network flow graphs in section 4. In section 5 we show that determining the optimal taint-check cover is NP-Complete, present taint-check cover heuristics, and describe verification algorithms to detect the presence of covert channels in a supply chain. We evaluate our algorithms by developing a supply chain simulator, as described in section 6. We detail possible mitigating contingencies in section 7, and conclude with future directions in section 8.

2. Threat Perception in RFID-enabled Supply Chain

In this section, we discuss the threat model chosen to analyze RFID-enabled supply chains, and present four possible supply chain attacks.

2.1.Threat Model

We present a motivating example to highlight the underlying assumptions used in the proposed threat model. Consider two competing businesses, each developing largely-interchangeable products, such as cellular phones. These businesses differentiate their products via competitive pricing and/or features, and are subject to user preferences and brand loyalty. For the sake of simplicity, assume that both businesses are competing in the same markets and target the same consumer base.

To remain competitive, these two businesses strive to optimize their internal processes to make their supply chains more efficient. When the target business invests in a new technology such as RFID, it examines the associated costs and benefits. While the benefits may be obvious in terms of efficient inventory control, production tracking, warranty authorization, etc., the cost of such a technology may involve more than just the direct cost of RFID equipment and processes.

The adversary business may also adopt RFID technology to remain competitive with respect to the target business. However, the adversary can also exploit the pervasive nature of this technology to clandestinely learn patterns of item flow in the supply chain of the target business. This can be construed as a form of industrial or economic espionage, wherein an adversary can use such discovered patterns in time-sensitive way, to provide lower consumer prices, or flood its products into selected regions or stores while the products of the target business become scarcer. Such practices can significantly reduce the profitability of the target business. The resulting profit drop can be viewed as hidden cost, which the target business would find difficult to anticipate, or even to correctly identify. Recent advances in RFID technology and the proliferation of its usage can thus give an adversary selective "*insider access*" to the target business supply chain, without direct access to target business' physical premises.

2.2.Attacks

A target business supply chain can inadvertently reveal its item flow to an adversary in a number of conceivable ways. We enumerate four representative (although not exhaustive) possible attacks, some of which have already received attention from the security research community [6, 14]. We explain the significance of these attacks when applied to a supply chain scenario, discuss the potential implications, and present possible ways to mitigate such attacks. Although such attacks are not dependent on any given RFID standard, for the sake of concreteness this paper assumes the EPC Gen2 standard [11, 12].

Tag tracking: In this attack, the adversary tracks the existing tags over the supply chain of a target business. We note that tags can be applied at the item-level or case-level. We assume that a target business assembles the finished product at its factory, attaches the tag at the case-level, and then ships them to geographically-separate warehouses. Upon arrival, these cases are organized into different batches and delivered to various retail stores. An adversary can learn the item-flow by copying the information stored in some case-level tags, and then querying them at different places in the supply chain. We note that such copied case-level tags constitute a covert channel, as they leak item-flow information from the target business to the adversary, while traveling unobtrusively through the supply chain of the target business.

Tag duplication: In this attack, an adversary copies the information stored in an existing tag and constructs a duplicate tag. Consequently, the adversary attaches this duplicate tag to a different case, enabling it to become part of the supply chain of the target business and thus a covert channel source. The adversary then queries the cases at different points of the supply chain to determine the item flow (e.g., if the adversary sees both the duplicate case-level tags at a warehouse then they aggregate at that

warehouse starting from different locations). This attack scenario is stronger than the previous tracking-only attack, since here the adversary is required to inject duplicate tags into the supply chain. Tag duplication hardware is relatively inexpensive and easily available; thus, an adversary can mount such an attack with modest effort [18].

Tag modification: In an EPC Gen2-compliant tag, there are four memory banks – Reserved, EPC, TID and User. The inventory process in a target business supply chain primarily uses the EPC portion of a tag's memory, and typically ignores the contents of the other memory banks. Therefore, an adversary can modify the information in the writable portions of other memory banks, which can then serve as a covert channel source. Independently, it has been suggested that the unused portion of memory of a tag can be utilized to conceal information [6, 17]. Such a vulnerability can be an attractive target to an adversary, due to its potentially large payoff versus relatively low effort to exploit.

Reader compromise: With rapid advances in RFID technology, various types of RFID readers are available in a wide variety of form-factors, hardware/software combinations, and use-case scenarios (i.e., handheld, rack-mountable, battery-powered, etc.). Many of these readers are deployed in supply chains in a manner that enables an adversary to compromise them (e.g., snooping on a wireless reader transmission, compromising the on-board software of a mobile reader, etc.). We differentiate two variations of this attack, as described in Figure 1.

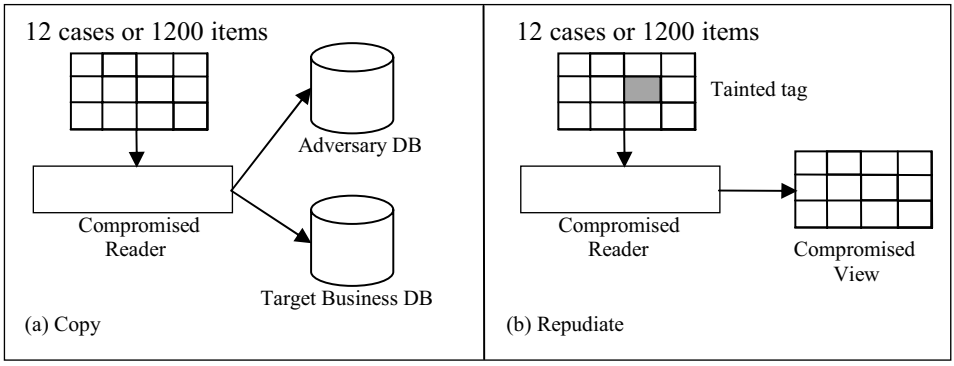


Figure 1: Reader compromise attack: (a) a compromised reader makes a copy of case-level tags; and (b) a compromised reader repudiates presence of covert channel.

In Figure 1(a), a compromised reader copies a limited number of case-level tags, and provides its information to an adversary. In Figure 1(b), an adversarial compromised reader selectively ignores the presence of any duplicate or modified tags. The reader's compromised view enables a covert channel to exist unobtrusively in the supply chain of the target business. We note that such a reader-compromise attack subsumes the tag duplication and modification attacks in terms of potential risks to the target business. From the adversary's perspective, in order to ensure a successful attack, while at least one compromised tag at the case-level is necessary, it may not be sufficient, since that tag may fail or become undetectable. Thus several compromised (i.e., duplicated and/or modified) tags should be used at the case-level (e.g., three tags per 100). On the other hand, if an adversary deploys too many compromised tags (e.g.,

half of the total), the adversary's exposure risk also increases dramatically. Moreover, an adversary may not need to track item-flow information at the item-level, since case-level tracking is sufficient for that purpose. The above types of attacks create covert channels that are said to "*taint*" the supply chain of a target business.

3. Projections for Market Change Scenario

In this section we examine two possible market scenarios to illustrate the potential impact of the RFID tag attacks described above. We note that a supply chain involves business-related variables such as stock levels at factories and retailers, delivery schedules from raw-material site to warehouses, numbers of back-orders, etc. Such strategic business information leak can occur as a result of attacking the supply chain, which can enable an adversary to engage in unfair competitive practices. Furthermore, an adversary can affect negative market changes by knowing the business practices of its competitors. We used the Anylogic supply chain model simulator [13] to obtain sample projections which, while simplistic, can still qualitatively describe possible outcomes of such attacks.

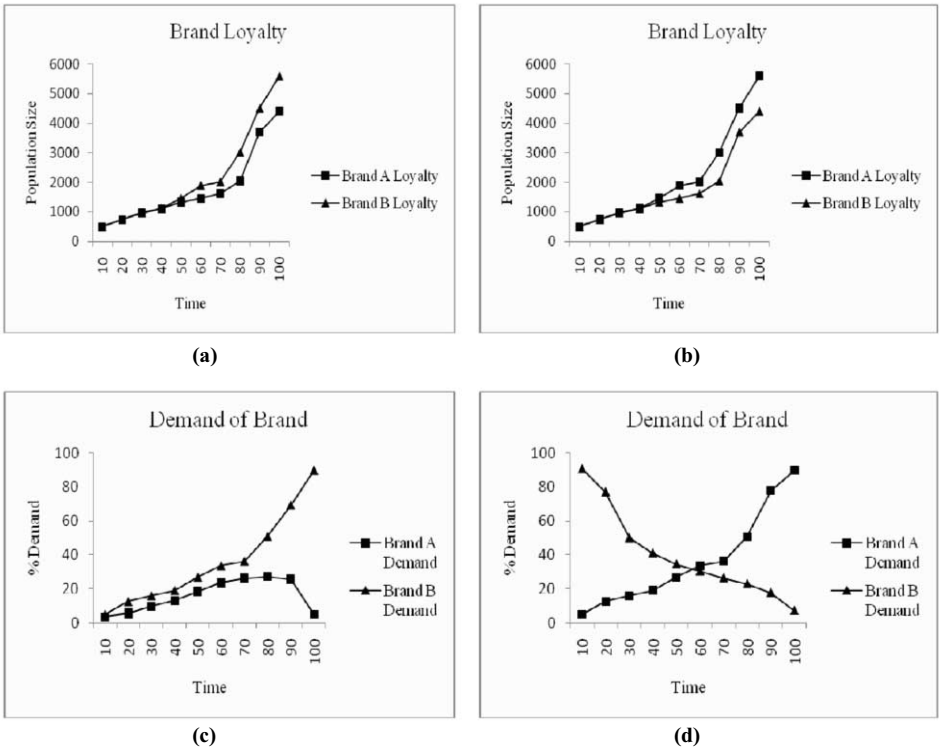


Figure 2: Market change projections using the Anylogic supply chain simulator [13]: (a) Consumers prefer brand B over brand A; (b) Consumer switch from brand B to brand A; (c) Brand B enjoys more demand than brand A; and (d) Brand A demand increases, while brand B demand decreases.

3.1. Brand Loyalty Switch

In the first scenario (Figure 2(a) and 2(b)), we consider two businesses serving a population of 10,000 consumers with brands A and B, respectively. We assume the two brands to be interchangeable, and have the same retail price. The business with brand B is the target business, while the business with brand A is the adversary. Consumers must purchase either brand A or brand B every time unit (i.e., the product is a staple item). In Figure 2(a), consumers are projected to prefer brand B to brand A by 55% to 45% (i.e., whenever a consumer arrives at a store, he chooses a product at random from the set of available equivalent products, preferring brand B slightly over brand A). However, by carefully timing its production so that more brand A products are available at a time when few brand B products are stocked or available, the adversary can induce consumers to switch brands. In Figure 2(b), the adversary has succeeded in inducing the consumers to switch brands, now favoring A over B by 57% to 43%.

3.2. Brand aversion

In the second scenario (Figure 2(c) and 2(d)), we consider a neighborhood store served by two businesses A and B, as before. Stores often stock products that enjoy consistent demand, in order to maintain profitability. Initially, the store stocks both items in equal amount. However, at a later point, as shown in Figure 2(c), brand B (i.e., the target business' product) is projected to have a higher demand than brand A by 89% to 5%. There is typically a demand threshold below which it will become non-profitable to stock a brand (i.e., "*brand aversion*"). An adversary aiming to bolster its own shelf presence may resort to illegitimately acquiring sensitive supply chain information of the competitor's business. Figure 2(d) projects such a scenario, when the adversary engages in supply chain attacks to obtain time-sensitive information about a target business, and use it to manipulate the market.

3.3. A Note on the Projected Market Change Scenarios

Enabling a supply chain with RFID technology entails attaching RFID tags at the item-level or case-level and tracking them throughput the supply chain using RFID readers. The target business keeps track of items starting from the purchase phase (e.g., in raw material form) through the distribution phase (i.e., in finished product form, stored at different warehouses or retail outlets). An adversary can use the possible attacks described above in order to learn vital strategic information, resulting in the projected market scenarios, which are detrimental to the target business.

If the benefits to an attacker are higher than its incurred costs, the adversary has strong motivation (i.e., economic incentive) for perpetrating such attacks. We believe that such attacks are viable in an RFID-enabled supply chain, given the potentially high payoff to an adversary, although specific occurrences of such attacks seem to have not yet been publicly reported. While we have argued that the exposure of only a few business variables to an adversary can result in an unfair (and not necessarily even illegal) marketplace advantage, it would be interesting to study more elaborate and detailed marketplace scenarios and projections. Such "*what-if*" scenarios can stimulate further discussions regarding the associated risks as well as the effectiveness of possible solutions in RFID-enabled supply chains.

4. Supply Chain Model

In this section, we focus on the problem of modeling a supply chain, towards the goals of preventing an attack or mitigating its effects. A supply chain typically spans multiple geographically dispersed sites and involves numerous phases that include the sourcing of raw-materials, processing and storing the end-product, and delivering the product to markets and consumers [6]. Supply chain models can be categorized as deterministic models, stochastic models, hybrid models, economic models, and IT-driven models [1, 2]. While these models intend to capture many aspects of a supply chain in great detail, our aim is to construct a simpler model that enables us to focus on the fundamentals and roots of potential attacks.

In any supply chain, there are item-flows between sites (e.g., raw materials moving among various locations), however in a RFID-enabled supply chain, item-flow between sites is analogous to "*tag-flow*", since RFID tags are attached to each item. The supply chain consists of multiple phases, wherein each phase is a collection of sites. Furthermore, to detect the presence of duplicate tags, modified tags, and compromised readers, we need mechanisms to track item-flows between supply chain phases. With these three key observations in mind, we have developed a model based on network flow graphs [10], which we call "*supply chain flow graphs*".

4.1. Phases

A supply chain can be broadly divided into three phases: the purchase phase, the production phase, and the distribution phase (e.g., sites associated with the production phase are involved primarily in manufacturing a product). Each phase of the supply chain is a collection of interconnected sites with an item-flow among them. We define the supply chain flow graph $G = (V, E)$ as a directed connected graph, where a node p corresponds to a site and an edge (p, q) models a connection between the two sites. Each edge $(p, q) \in E$ has a positive item-flow capacity $C(p, q) > 0$, while "*non-edges*" have 0-capacity: $\forall\ (p, q) \notin E.\ C(p, q) = 0,$. There are two special nodes called the "*source node*" (S) and the "*sink node*" (T). We partition the supply chain flow graph into three sub-graphs, corresponding to the purchase phase, production phase, and distribution phase, respectively.

Network flows are subject to the usual constraints on edge capacity and flow conservation at nodes [10]. We propose an additional property, namely the node maximal outgoing flow, which will enable us to address issues related to attacks. There are typically multiple paths for item-flow in a supply chain. A "*critical node*" or "*critical edge*" may experience more item-flow than other paths. To model this characteristic in the supply chain, each node keeps track of its maximum outgoing flow. If two nodes have the same maximal outgoing flow, we resolve the tie by giving precedence to the node having a higher flow value predecessor. Supply chain flow graphs with such criticality labels facilitate reasoning about issues related to possible attacks and item-flow inspections.

4.2. Taint Checkpoints

A direct approach for detecting covert channel attacks can entail inspecting for tainted RFID tags at every node of the supply chain. However, this would be prohibitively expensive and time consuming. Instead, we propose to select a subset of nodes, called "*taint checkpoints*", verify the item-flow at these selected locations, and report the presence of any discovered covert channels in the supply chain flow graph. When RFID tags are attached to items by the target business in the early phases of the supply chain, the information present on them is recorded in order to track inventory. In subsequent phases of the supply chain, this information is available to taint checkpoints for the purpose of inspection and verification. This verification process involves comparing the information present on the currently viewable RFID tag with trusted, stored information. Any mismatch may indicate the presence of covert channels or other tampering. Figure 3 illustrates a supply chain flow graph, including several taint checkpoints where item-flow is inspected.

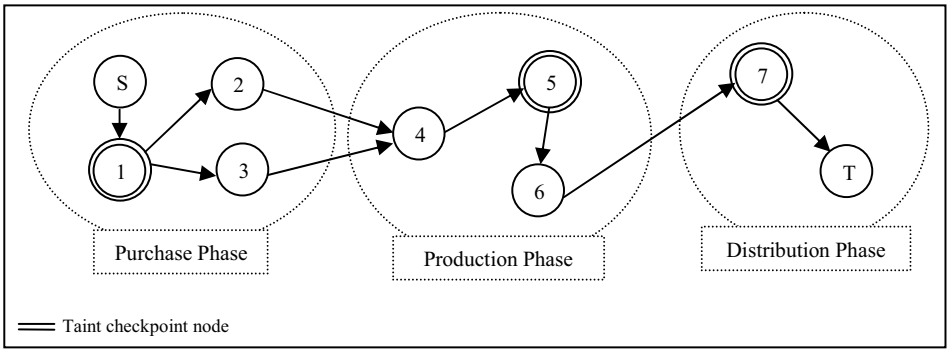


Figure 3: A supply chain flow graph with three taint checkpoints.

5. Taint Check Cover Generation and Verification Algorithm

In this section, we formulate the problem of optimal selection of taint checkpoints in the supply chain flow graph, observe that it is NP-Complete, and suggest heuristics to generate good approximate solutions.

5.1. Taint Check Cover Problem Statement

To ensure the absence of covert channels in the supply chain, the taint checkpoints should provide broad coverage for the entire graph. The associated optimization problem is to select as few taint checkpoints as possible, while providing broad coverage for the entire supply chain flow graph. Thus we seek a "*taint check cover*" V' of the supply chain flow graph $G_U = (V, E)$, where $V' \subseteq V$ and such that every edge of E has at least one of its end points in V'. Note that we may choose to only cover some critical subset of the flow graph's nodes, rather than the entire graph. Either way, this objective corresponds to the classical graph vertex cover problem, which is known to be NP-complete [10].

5.2. Heuristic Taint Check Cover Generation

There is a simple efficient heuristic for vertex cover that produces solutions of size no worse than twice the optimal [10]. This heuristic selects an arbitrary graph edge, adds its two endpoints to the growing vertex cover solution, eliminates this edge and its endpoints from the graph, and iterates until the graph is exhausted. To see that this scheme produces a 2·OPT solution, we observe that one of the two nodes of each removed edge must be present in any optimal solution. Given the high degree of freedom in how edges (and thus nodes) are selected in constructing such a heuristic taint check cover solution, a target business may introduce different selection criteria, based on practical, economic, or strategic considerations.

Parameters: A target business may wish to limit the number of taint checkpoints, seek tradeoffs between the efficiency of its supply chain versus the coverage provided by taint checkpoints, consider checkpoint selection criteria based on the specific structure of the supply chain, etc. To address these considerations, we introduce two parameters:

1. **Taint checkpoint to nodes ratio (or *TNR*):** This is defined as ratio of taint checkpoints to graph nodes in the supply chain flow graph, and enables the target business to control the number of taint checkpoints:

$$TNR = \frac{|V'|}{|V|} \qquad \text{where, } |V| \neq 0 \qquad (1)$$

2. **Coverage to efficiency ratio (or *CER*):** The ratio ε of coverage and efficiency provides a tradeoff to balance the quality of item-flow inspection against the overall operational efficiency of the supply chain:

$$CER = \varepsilon \qquad \text{where, } \varepsilon > 0 \qquad (2)$$

Heuristic template: When determining a taint check cover, the target business may choose from a continuous tradeoff between efficiency and coverage. This can be achieved by using the parameters *TNR* and *CER* to determine from which subset of the flow graph nodes (i.e., V') a taint check cover will be selected (using, e.g., the 2·OPT node cover heuristic [10], the techniques described in [20], or any other node cover heuristic). We can also presort the node selection pool by increasing node maximal outgoing flow values, in order to give higher priority to high-flow nodes. Alternatively, the nodes can be permuted in some other manner (e.g., by aggregate product value, time-criticality, or even randomly), in order to capture topological or economic considerations during the construction of a taint check cover. In summary, our template is quite general in that it can utilize (based on the above parameters) any reasonable criteria to determine which node subset will be used from which to select a taint check cover (using an arbitrary node cover heuristic).

We note that not nearly every node and/or edge in the flow graph must necessarily be covered (i.e., imbued with taint-checking capability). This is because tainted tags that are missed at some points along the graph will likely be discovered at subsequent locations downstream. On the other hand, including any flow graph "*cut*" in the taint check cover can ensure that every tainted tag will be discovered in at least one location.

An alternative taint check cover can therefore entail selecting a small (but somewhat redundant) set of cuts across the flow graph. This can insure at relatively low infrastructural cost that any tainted tags moving in the graph will eventually be detected. The taint checkpoints chosen in Figure 3 demonstrate such a cut-based cover. Choosing low-cost (or even optimal) graph cuts can be accomplished using well-known min-cut algorithms [19].

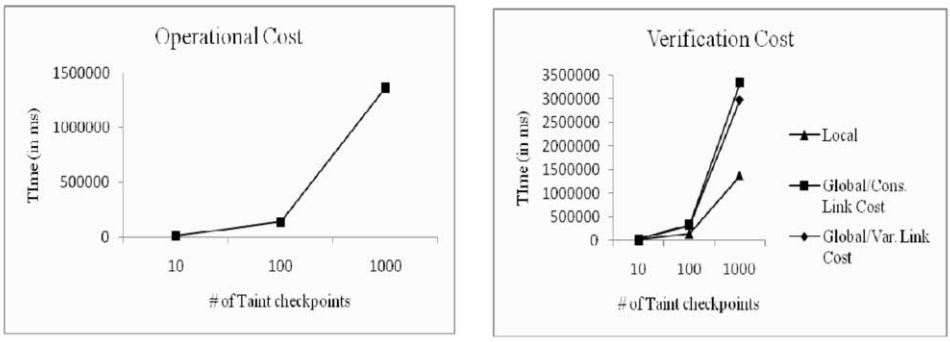
5.3. Verification Algorithm

Each node in the taint check cover (i.e., each taint checkpoint) is responsible for inspecting and verifying the item-flow passing through it. Each item in this flow has a unique RFID tag ID. If a taint checkpoint reads multiple counts of the same tag ID, or the system detects the same tag ID at two different places simultaneously, then a duplicate tag has been detected. By comparing the information present on each viewable tag with data stored a priori in a trusted tag database, modifications to tags can be detected at taint checkpoints. Item-flow verification can be performed "*locally*" at a given taint checkpoint or "*globally*" across a given path or cut, as checkpoints accumulate, exchange, and compare tag information.

6. Evaluation

We used simulations to evaluate our proposed approaches. We assume a base supply chain flow graph configuration of 2000 nodes, and selected between 10 and 1000 nodes to be taint checkpoints. Each checkpoint verifies 1000 cases of 100 items at each time interval. We assume each checkpoint has direct access to a trusted database implementing a tag lookup service. In our first simulation, we measure the relationship between the number of taint checkpoints and cumulative time required to perform local verification. Figure 4(a) shows that as the number of taint checkpoints increase, there is a corresponding increase in the time to locally verify the item-flow.

Our second simulation evaluates our global verification algorithm, which collects local verification results from taint checkpoints. The cost of the collection process depends on the underlying speeds of the network links.



(a) **(b)**

Figure 4: (a) cumulative local verification time as a function of the number of taint checkpoints; and (b) local and global verification costs as a function of the number of taint checkpoints.

Figure 4(b) shows the simulated verification cost when the link cost is either a constant (500 ms) or a variable time window (ranging from 2 to 1000 ms), based on the node's geographical distance from the central database server. We thus explored the verification communication cost as the number of taint checkpoints increases. We note that the communication cost can grow rapidly in the more realistic scenario where taint checkpoints are at large variable distances from the central node.

7. Responses to Covert Channels

In this section, we enumerate some possible response actions available to the target business when the covert channels are detected in its supply chain. Note that the presence of covert channels in the supply chain can never be completely ruled out, even when privacy-preserving algorithms are used in the underlying RFID technology [4, 5].

Passwords: According to EPC Gen2 standard, an RFID tag is required to support password protection for read or write access to the tag. The systematic use of passwords can mitigate tag tracking, tag duplication, and tag modification attacks. However, this requires that all RFID hardware in the supply chain support and conform to the same password scheme.

Pseudonyms: An RFID tag using pseudonyms transmits a slightly different ID each time it is queried [3]. This can prevent the adversary from discovering patterns in a supply chain, but requires the target business to accommodate the pseudonym scheme in its tracking logic. Burmester *et al.* describe an unlinking technique that can also be used to prevent tag tracking attacks [15].

Re-encryptions: The use of encryption to conceal the tag data still allows the adversary to track the static encrypted tag over the supply chain. To defeat such an attack, the tags can be re-encrypted after each phase of the supply chain, in order to prevent the adversary from modifying or tracking the tags.

Direct Mitigation: Rieback *et al.* describe a device that can be used for sweeping and preventing reader compromise attacks [7]. When a covert channel source is discovered, we can physically clear the operating environment while temporarily altering the flow of items. Oua *et al.* present a path checking technique that can trace tags following the altered route [16].

Physically Unclonable Functions (or *PUF*s): PUFs are hardware random number generators that rely on inherent wire-delays and process variations [9]. PUF-based privacy-preserving algorithms provide a way to build message authentication codes to ensure data integrity and aid in preventing tag modification attacks.

8. Conclusion

In this paper, we discussed and analyzed vulnerabilites in RFID-enabled supply chains, and enumerated possible attacks that can be mounted with relatively modest effort. We have shown that an adversary can learn item-flow patterns in the RFID-enabled supply chain of a target business, which may result in harmful market change scenarios. We proposed a concise model for reasoning about supply chain flow and

RFID attack mitigation, and demonstrated that attacks can be detected and addressed at a few select nodes in the supply chain. For the NP-complete problem of checkpoint selection we presented a practical heuristic template that can trade off attack coverage for efficiency. We simulated and analyzed these algorithms, and enumerated possible responses by a target business to covert channels. While this work is preliminary, we view it as an important step toward the analysis and mitigation of attacks on RFID-enabled supply chains. Possible future research directions include extending the basic model to include additional practical considerations, fine-tuning the heuristics to take these additional practical considerations into account, and further study the tradeoffs between coverage and efficiency.

References

[1] H. Min and G. Zhou, Supply Chain Modeling: Past, Present and Future, *Journal of Computer and Industrial Engineering*, Elsevier Science Direct, Volume 43, Issue 1-2, pp. 231-249, July 2002.

[2] R. Angeles, RFID Technologies: Supply-Chain Applications and Implementation Issues, *Information Systems Management*, 22:1, pp. 51-65, 2005.

[3] D. Molnar, A. Soppera and D. Wagner, A Scalable, Delegatable Pseudonym Protocol Enabling Ownership Transfer of RFID Tags, *Selected Areas in Cryptography*, Ontario, Canada, 2005.

[4] D. V. Bailey, D. Boneh, E. Goh and A. Juels, Covert Channels in Privacy-Preserving Identification Systems, *14th ACM International Conference on Computer and Communication Security*, Alexandria, Virginia,pp. 297-306, 2007.

[5] S. L. Garfinkel, A. Juels and R. Pappu, RFID Privacy: An Overview of Problems and Proposed Solutions. *IEEE Security and Privacy*, Volume 3, Issue 3, pp. 34-43, May 2005.

[6] A. Mitrokotsa, M. R. Rieback and A. S. Tanenbaum, Classification of RFID Attacks. *International Workshop on RFID Technology*, Barcelona, Spain, pp. 73-86, June 2008.

[7] M. R. Rieback, B. Crispo and A. S. Tanenbaum, RFID Guardian: A Battery-Powered Mobile Device for RFID Privacy Management, *Lecture Notes in Computer Science*, Springer, Volume 3574, pp. 184-194, July 2005.

[8] I. S. Moskowitz and M. H. Kang, Covert Channels – Here to Stay, *9th IEEE International Conference on Computer Assurance*, pp. 235-243, July 1994.

[9] L. Bolotnyy and G. Robins, Physically Unclonable Function-Based Security and Privacy in RFID System, *5th International Conference on Pervasive Computing and Communications*, New York, USA, pp. 211-128, March 2007.

[10] T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, *Introduction to Algorithms, Third Edition*, MIT Press, Cambridge, 2009.

[11] EPCGlobal, UHF C1 G2 Air Interface Protocol Standard http://www.epcglobalinc.org/standards/uhfc1g2/uhfc1g2_1_1_0-standard-20071017.pdf

[12] EPCGlobal, Tag Data Standards Version 1.4, Revision June 11, 2008 http://www.epcglobalinc.org/standards/tds/tds_1_4-standard- 20080611.pdf

[13] Anylogic Professional 6. AB-SD Supply Chain Model Simulator, http://www.xjtek.com

[14] G. Avoine, C. Lauradoux, and T. Martin, When Compromised Readers Meet RFID, *Workshop on RFID Security*, Leuven, Belgium, 2009.

[15] M. Burmester and J. Munilla, A Flyweight RFID Authentication Protocol, *Workshop on RFID Security*, Leuven, Belgium, 2009.

[16] K. Oua and S. Vaudenay, Pathchecker: An RFID Application for Tracing Products in Supply-Chains, *Workshop on RFID Security*, Leuven, Belgium, 2009.

[17] A. Karygiannis, T. Phillips, and A. Tsibertzopoulos, RFID Security: A Taxonomy of Risks, *Conference on Communications and Networking in China,* Beijing, China, pp. 1-8, 2006.

[18] J. Mandel, A. Roach, and K. Winstein, MIT Proximity Card Vulnerabilities, *Technical report*, MIT, March 2004. http://web.mit.edu/keithw/Public/MIT-Card-Vulnerabilities-March31.pdf

[19] M. Stoer and F. Wagner, A Simple Min-Cut Algorithm, *Journal of the ACM*, Vol. 44, Issue 4, pp. 585-591, July 1997.

[20] J. Chen, I. A. Kanj, W. Jia, Vertex Cover: Further Observations and Further Improvements, *Journal of Algorithms*, Publisher: Elsevier, Vol. 41, Issue 2, pp. 280-301, November 2001.

Radio Frequency Identification System Security
Y. Li and J. Zhou (Eds.)
IOS Press, 2010

doi:10.3233/978-1-60750-485-6-147

Anonymous RFID Yoking Protocol Using Error Correction Codes

Chin-Feng LEE[a], Yu-Chang CHEN[b], Hung-Yu CHIEN[c1] and Chi-Sung LAIH[d]

[a] Department of Information Management, ChaoYang University of Technology, Taiwan, R.O.C.
[b] Department of Information Engineering and Computer Science, Feng Chia University
[c*] Department of Information Management, National Chi-Nan University, Taiwan, R.O.C.
[d] Department of Electrical Engineering at National Cheng Kung University, R.O.C.

Abstract. A Radio Frequency Identification (RFID) yoking proof protocol allows a verifier to collect the evidence that two tags are simultaneously present. Yoking proof protocol has been applied in several potential applications like shipping record checking and medicine dispensation checking, etc. This paper, based on error correction codes (ECC), designs a novel yoking proof protocol, which not only protects tag's anonymity but also requires only simple operations that can be easily supported on low-cost tags. Compared to its counterparts, our scheme provides several practical merits: (1) It protects tag's anonymity using much easier approach, (2) it requires only simple operations on tags, and (3) the computational overhead on the server is much lower.

Keywords: security, authentication, Radio Frequency Identification, yoking proof, error correction codes.

1. Introduction

In 2004, Juels introduced an emerging and interesting RFID application- the yoking proof [7], in which a verifier would like to collect the evidence of simultaneous presence of two tags in the communication range of a specified reader. Juels introduced several applications of yoking proof protocols [7]. Here, we take one example. A pharmacist would like to ensure whether a specific medicine (say A) is dispensed with another medicine (say B) for patient safety. So, the pharmacist can label each medicine A and each medicine B with distinct tags, and then applies the yoking protocol to collect the evidence of simultaneous presence of the tagged medicines before the dispensation. Of course, the pharmacist should learn the identities of the tags (and the identities of the corresponding medicines). But, he would not let adversaries (or any outsiders) learn the identities of medicines to protect patient's privacy.

While Juels called the protocol the yoking protocol, Saito and Sakurai [12] called it the grouping proof protocol and Lopes et al. [11] called it the clumping protocol. In this paper, we refer to them all as yoking protocols (or proofs). Following Juels's work, Saito and Sakurai [12] applied the timestamp to improve Juels's scheme to resist replay attacks; however, Piramuthu [13] showed that Saito-Sakurai's scheme failed to resist the replay attack, and proposed an enhanced scheme using random number challenges.

[1] Corresponding Author. This work was supported in part by National Science Council under the grants NSC 97 - 2221 - E - 260 - 008 - MY2.

Lopes et al. [11] showed the weakness of Piramuthu's scheme that un-correlated random numbers in the scheme could be exploited to forge valid transcripts. Despite of the security weaknesses reported, the previous yoking protocols like [2,7,8,11–13] all require hashing function on tags, and the server overhead to identify tags is too costly (exhaustive searching) for those anonymous versions in their schemes. Up to now, hashing function is still too costly to low-cost tags. Contrary to RFID authentication protocols like [1,3–6,9,10,14–17,28] where the verifier is on-line, we note that the verifier in yoking protocol is usually off-line or in on-line batch mode.

This paper, based on error correction codes (ECC), shall propose a novel RFID yoking proof protocol. Our contributions are two fold: (1) it is the first ECC-based yoking scheme, (2) easily supports tag's anonymity, and (3) it requires only simple operations like Pseudo Random Number Generation (PRNG) and simple bit-wise operations (XOR and AND) on tags. The rest of this paper is organized as follows. Section 2 introduces the security model, and Section 3 proposes our scheme. Section 4 analyzes security and evaluates its performance. Finally, Section 5 states our conclusions.

2. The Model of Secure Yoking Proof Protocols

Regarding RFID security, the previous works like [6,10] have set up the models and defined the security for RFID authentication protocols, where the verifier validates the authenticity of one tag on-line. On the contrary, the verifier in yoking proofs is required to validate both the authenticity and the time-binding of two (or more) related tags off-line or in on-line batch mode.

An RFID yoking proof protocol involves three kinds of entities- tags, the reader and the verifier. We follow the assumptions of [2] that RFID tags do not own clocks or maintain time; however, the technique of measuring the discharge rate of capacitors can be used to limit the time span of a specific activity. The verifier is trusted, and the channel between the readers and the verifier is secure. The verifier is in off-line mode or in on-line batch mode; that is, the verifier would not directly interact with a specified tag on-line.

2.1 Security Requirements

The security threats to yoking protocols include the following. (1) **Replay attacks**: An attacker eavesdrops on the communications and replays them later to violate the correctness of the protocols. (2) **Tag impersonation**: Here, we do not consider the hardware cloning or reverse engineering, but consider an attacker exploits the weaknesses of the protocols to impersonate tags. (3) **Denial-of-service attacks**: An attacker causes tags to assume a state from which they can no longer function properly. The attacker might plot his attack by de-synchronizing the states between tags or between the verifier and the tags. (4) **Privacy, Tracking**: For those applications where anonymity is concerned, an attacker might use the transmissions to identify or track a specific tag.

Security requirements: Therefore, the security requirements of secure yoking protocols include authentication of tags, resistance to DOS attack, secure binding of yoked tags to a specific time span, and desirable anonymity.

2.2 The Model and the Security Definition

Here we define the privacy (anonymity) requirement as the un-traceability (UNT) property. Our security model for UNT is modified from Avoine's RFID privacy model [6] (which concerned the privacy in terms of un-traceability (UNT) during authentication processes). The differences of our model and Avoine's model include: (1) Avoine's model is targeted for RFID authentication protocol, whereas we concern the yoking protocols for the simultaneous presence of two matched tags; (2) Avione's notations of Existential-UNT-QSE restricts Adversary's capacity to respective only one Q (Query), S (Send), E (Execute) query, but our notation of Existential-UNT-QSE allows many queries to Q, S, E oracles; (3) Avione's model restricts to 3-move authentication, but our model is more flexible: our oracle query Q allows the adversary to adaptively adjust its next move, based on the responses from T or from R.

An RFID yoking proof system consists of a single reader $\Re$ and a group of related tags $\Im = \{T_1,...,T_{n1}\}$. The adversary is denoted as *Adver*, and the *i*-th instance of tag *T* is denoted as π_T^i. The capacities of the adversary are defined through the following oracle queries.

– Query(π_T^i, *m1*): this query models *Adver* sending *m1* to *T* and receiving the response from *T*.
– Send(π_R^i, *m2*): this models *Adver* sending *m2* to $\Re$, and receiving the response.
– Execute($\pi_{T_1}^i$, $\pi_{\Re}^k$, $\pi_{T_2}^j$): this models *Adver* executing protocol instance between $\pi_{T_1}^i$, $\pi_{T_2}^j$ and $\pi_{\Re}^k$, and obtaining all the messages exchanged among them.

A protocol is said to be resistant to attacks *A-O* or it is *A-O* if it is resistant to an attack *A* when the adversary has access to the oracles of $O \subset \{Q,S,E\}^*$, where Q, S, and E represent the oracles queries Query, Send, and Execute respectively. An interaction $\omega_i(T)$ denotes the result of the application of an oracle Q, E or S on *T*, and $\Omega_I(T) = \{\omega_i(T) \mid i \in I\} \cup \{Send(\pi_{\Re}^j,*) \mid j \in J\}$, where $J \subset N$. The notation of un-traceability (UNT) is defined as, after having interactions with the target *T*, its partnered tags $\in \Im$ and the reader, and thus obtaining an interaction $\Omega_I(T)$, whose length is less than a pre-defined parameter l_{ref}, an adversary *Adver* needs to find her target among two tags T_1 and $T_2 \in \Im$ (one of them is the target) that are given to her. *Adver* can query both T_1 and T_2, and obtains two interactions $\Omega_{I_1}(T_1)$ and $\Omega_{I_2}(T_2)$ whose lengths are less than a given length l_{chal}. If there exists I_1 and I_2 such that *A* is able to succeed then it is existential traceability [6].

Existential Un-Traceability

Parameters: l_{ref}, l_{chal}, O, $\Im$

1. *Adver* requests the *challenger* and receives her target $T \in \Im$.
2. *Adver* chooses *I* and calls *Oracles*(*T*, $\Im$, *I*, *O*) where $|I| \leqq l_{ref}$ then receives $\Omega_I(T, \Im)$.
3. *Adver* requests the *challenger* and receives her challenge T_1 and T_2 (T_1, $T_2 \in \Im$, and one of them is *T*).

4. *Adver* chooses I_1 and I_2, calls *Oracles*(T_1, $\Im$, I_1, O) and *Oracles*(T_2, $\Im$, I_2, O), and receives $\Omega_{I_1}(T_1, \Im)$ and $\Omega_{I_2}(T_2, \Im)$, where $|I_1|$, $|I_2| \leqq l_{chal}$ and $(I_1 \cup I_2) \cap I = \phi$.
5. *Adver* guesses which one of T_1 and T_2 is T, and outputs her guess T'.

The advantage of *Adver* for a given protocol P is defined as $ADV_P^{UNT}(Adver) = 2\Pr[T' = T] - 1$, where the probability space is over all the random tags $\in \Im$. If *Adver*'s advantage is negligible with parameters l_{ref}, l_{chal} and O, then P is said to be UNT l_{ref}, l_{chal}-O secure.

3. The Proposed ECC-Based Yoking Scheme

Our scheme is based on ECC. We will give some preliminaries about ECC in Section 3.1, and then propose our scheme in Section 3.2.

3.1 Preliminaries – Error Correction Codes

ECC [26] has been applied in the design of public-key cryptosystems like [21,22], in symmetric encryption [23–25], in authentication [18,20], and in secret sharing [19]. The McEliece-like public key cryptosystems [21,22] have been studied to ensure it is indistinguishable against adaptively chosen cipher text attacks (IND-CCA2) [26], and the ECC-based symmetric encryption schemes are designed to ensure it is indistinguishable against chosen plaintext attacks (IND-CPA) [26]. Despite of distinct security strengths, the common feature of the previous ECC-based schemes is that they all require either matrix computations or combinatorial functions which are too costly to low-cost tags. In this paper, we will solve this problem when we design our yoking scheme. In the following, we give some preliminaries of ECC.

Error correction codes and anonymity

A linear error correction code of length n, dimension k, and minimum distance d is denoted by $C(n, k, d)$, and the codes can be defined by its k-by-n generator matrix $\boldsymbol{G}$. As several previous works [19,20], this study artificially adds error vectors to the transmitted data such that both the transmitted data, and the artificial errors can only be recovered by the designated receivers. However, in the previous ECC-based schemes, it is required to compute matrix computations or combinatorial functions, which are too costly for low-cost RFID tags. The challenge is to design an ECC-based RFID yoking scheme where the tag only performs simple operations. A simple scenario is used to describe our idea of achieving anonymity as follows. Let S be the server and $\{T_i\}$ be a set of tags. Let $l=|\{T_i\}|$ be the number of tags. Initially, S randomly chooses a secret linear code $C(n, k, d)$ over GF(2), as specified by its generator matrix G, which is a k by n matrix. Here, we can assume $l|k$ without loss of its generality, and we denote $s=k/l$ as the number of row vectors assigned to a tag. Here the notation $l|k$ means l can divide k, and k / l means the quotient for k being divided by l. S assigns $G[j]$s to T_i for $j=(i-1)*s+1, \ldots, i*s$, where $G[j]$ denotes the j-th row of G. T_i later applies linear combinations of rows $G[j]_{j=(i-1)*s+1 \sim i*s}$ to generate a new codeword c_i. Then, c_i would equal $m_i * G$ for some k-bit vector m_i with the j-th component of m_i being zero for

$j \notin [(i-1)*s+1, i*s]$. T_i generates c_i as above, and adds an error vector e with weight $t = \lfloor (d-1)/2 \rfloor$ to derive $\hat{c}_i$. That is, $\hat{c}_i = c_i + e$. T_i sends $\hat{c}_i$ to S, which then uses the decoding algorithm [27] to derive (m_i, e) to identify the tag. The above idea can achieve anonymity through the identification process, since only the designated receiver S, which owns the private generator matrix and private parity check matrix, can recover (m_i, e) to identify the tag. And the transmission in each session looks random to an eavesdropper. However, the above idea is vulnerable to various attacks, such as replay attacks and message-resend attacks. In the following, we extend the above idea to propose an ECC-based RFID yoking scheme with anonymity.

3.2 The Proposed Scheme

The scheme involves three entities: a backend server (the verifier in the yoking scenarios), a reader, and a set of tags. The channel between the server and the reader is assumed secure; however, the wireless channel between readers and tags is vulnerable to various attacks. We follow the assumptions of [2] that RFID tags do not own clocks or maintain time; however, the technique of measuring the discharge rate of capacitors can be used to limit the time span of a specific activity. The notation is summarized in Table 1. The scheme consists of two phases - the initial phase and the yoking phase.

Table 1. Notations

G, $G[i]$, t: G is the server's secret generator matrix for random chosen linear codes $C(n, k, d)$, and $G[i]$ denotes the i-th row of G. $t = \lfloor (d-1)/2 \rfloor$.
T_i, K_i, l, s: T_i denotes the identity of a tag. K_i is the secret key shared between T_i and the server S, and $\|K_i\| = l_g$. $l = \|\{T_i\}\|$ denotes the number of tags. We assume $l \| k$, and let $s = k/l$ denotes the number of rows of G assigned to each tag.
G_A, G_B: G_A is defined to be the set of row vectors of G assigned to T_A, and G_B is defined to be the set for T_B.
$g()$: a public cryptographic pseudo random number generator- $g(): \{0,1\}^{l_g} \rightarrow_R \{0,1\}^{l_g}$. In this paper, we do not consider forward secrecy and backward secrecy, and a standard cryptographic PRNG introduced in [28,29] that satisfies pre-image resistant is suitable.
S: denotes the backend server who owns the secret parameters $C(n, k, d)$, and maintains a secret database, which contains the information of each tag; for example, the identity T_i, the key K_i, the indices of the assigned rows of G to this tag.
R, N_R: R denotes the reader, and N_R denotes the random numbers generated by the reader. $\| N_R \| = \| K_i \| = l_g$.
$ch = E_{k_V}(timestamp)$: In order to ensure the authenticity of each challenge from the verifier, we define the challenge as $ch = E_{k_V}(timestamp)$, which is an encryption of the verifier's timestamp using the key k_V (k_V is the secret key owned by the verifier). *The proofs corresponding to a specific ch should be returned within a reasonable time span. The idea of using encrypted timestamp in Yoking schemes was first proposed by [11].*

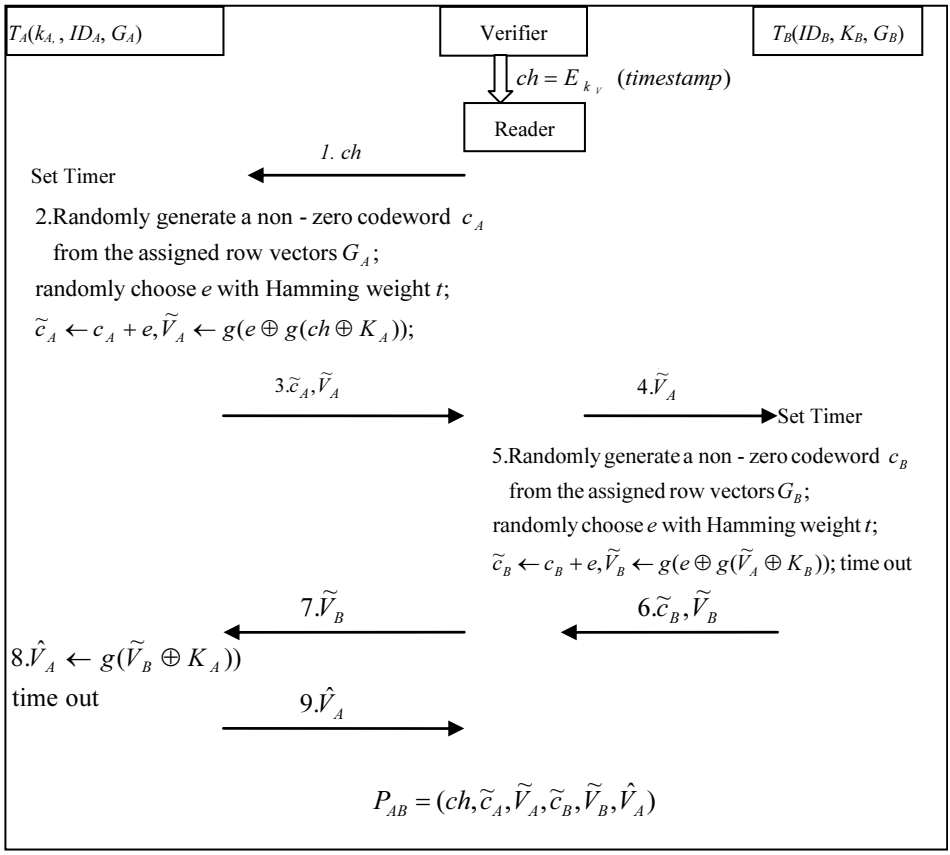


Figure 1. Anonymous yoking proof.

Initialization

Initially, *S* publishes a cryptographic PRNG *g*(), randomly chooses a secret linear code *C*(*n, k, d*) over GF(2) as specified by its generator matrix *G*, and assigns row vectors *G*[*j*]s, for *j*=(*i*-1)**s*+1, …,*i***s,* to T_i. *S* maintains the information of each tag in its database, which contains the tag's identity T_i, the secret key K_i, and the indices of the assigned rows of *G* to each tag. To each tag T_i, *S* writes the identity T_i, the key K_i, *g*(), and the row vectors *G*[*j*]s, for *j* = (*i*-1)**s* + 1, …,*i***s*, into T_i's memory. For tag T_A, the set of assigned row vectors is denoted as G_A.

The yoking phase

The yoking phase of the protocol is described as follows and in Fig. 1. Here, we assume T_A be the initiator tag, and T_B be the responder tag. An initiator tag would respond to reader's yoking proof request first, and its role can be pre-defined or any tag can be an initiator tag. We would not discuss the implementation detail here.

Step 1. The reader first acquires an encrypted challenge $ch = E_{k_V}$ *(timestamp)* from the verifier, and then broadcasts *ch* to its nearby tags. This arrangement makes adversaries can forge a valid challenge with negligible probability.

Step 2. After receiving the challenge *ch*, T_A starts the timer and generates a non-zero codeword c_A via a random linear combination of row vectors from G_A. T_A randomly chooses error vector *e* with Hamming weight *t*, and computes $\widetilde{c}_A = c_A + e$. It computes $\widetilde{V}_A = g(e \oplus g(ch \oplus K_A))$. Here we abuse the notation $g(e \oplus g(ch \oplus K_A))$, even if the length of *e* is distinct from the output length of *g*(), and a necessary string expansion or shrinking is applied when the lengths of two operands are different. Codeword c_A can be viewed as $c_A = m_A * G$, where m_A is a *k*-length vector with the *j*-th component of m_A being zero for those vectors not belonging to G_A. A potential mechanism for implementing timer on low-cost tags has been discussed in [31,32].

Step 3. The tag sends ($\widetilde{c}_A$, $\widetilde{V}_A$) to the reader.

Step 4. The reader forwards $\widetilde{V}_A$ to the responder tag.

Step 5. After receiving $\widetilde{V}_A$, T_B starts the timer and then generates a non-zero codeword c_B via a random linear combination of row vectors from G_B. T_B randomly chooses error vector *e* with Hamming weight *t*, and computes $\widetilde{c}_B = c_B + e$. It computes $\widetilde{V}_B = g(e \oplus g(\widetilde{V}_A \oplus K_B))$. Codeword c_B can be viewed as $c_B = m_B * G$ for some *k*-length vector m_B with the *j*-th component of m_B being zero for those vectors not belonging to G_B.

Step 6. T_B returns ($\widetilde{c}_B, \widetilde{V}_B$) to the reader.

Step 7. The reader forwards $\widetilde{V}_B$ to T_A.

Step 8. T_A computes $\hat{V}_A = g(\widetilde{V}_B \oplus K_A)$.

Step 9. T_A returns $\hat{V}_A$ to the reader.

The final yoking proof is the set $P_{AB} = (ch, \widetilde{c}_A, \widetilde{V}_A, \widetilde{c}_B, \widetilde{V}_B, \hat{V}_A)$. When the verifier receives the proof P_{AB}, it will check the following conditions to verify this proof. If all the following conditions hold, the proof is valid.

(1) Decrypt $ch = E_{k_V}(timestamp)$ to verify the validity of the timestamp.

(2) Decode $\widetilde{c}_A$ to get c_A and the corresponding *e*, and verify whether the equation- $\widetilde{V}_A \stackrel{?}{=} g(e \oplus g(ch \oplus K_A))$ - holds.

(3) Decode $\widetilde{c}_B$ to get c_B and the corresponding *e*, and verify whether the equation- $\widetilde{V}_B \stackrel{?}{=} g(e \oplus g(\widetilde{V}_A \oplus K_B))$ - holds.

(4) Verify whether the equation- $\hat{V}_A \stackrel{?}{=} g(\widetilde{V}_B \oplus K_A)$ - holds.

4. Security Analysis and Performance Evaluation

4.1 Security and Proof

The anonymity (Existential-UNT-QSE) of the proposed ECC-based yoking scheme protocol is proved in the following theorem.

Theorem 1. The proposed ECC-based yoking scheme is Existential-UNT-QSE, if we take the cryptographic PRNG as a random oracle.

Proof: (1) Existential-UNT-QSE: Without the secret parameters- the generator matrix, the parity matrix, tag's secret key, tag's row vectors and tag's identity, an adversary cannot derive any tag's information from the transmissions (which consists of error-vector-added random codeword and cryptographic PRNG output on secret key and random vector). Since the data of each interaction between the tags and reader would depend on the random error vector, the secret key, these data would seem random and independent to an adversary. The adversary has no way to identify the tag or to trace a tag. □

In addition to the above proof of Existential-UNT-QSE, we also analyze the other security properties of the proposed protocols as follows. Our scheme can be viewed as a secret-key-based authentication scheme where the secret keys includes the assigned row vectors and the secret key assigned to each tag, and the attacker can only observes the transmissions or modifies the transmissions to launch its attacks. We, therefore, exclude those chosen ciphertext attacks (like [22]) on McEliece-like public key cryptosystems (like [21]) and the chosen plaintext attacks (like [23,25]) on ECC-based symmetric encryption schemes [24] in the following analysis. But, we consider message-resend attacks and related-message attacks similar to [33], because either genuine readers or attackers may probe the same tag many times. Now we examine the security and attacks as follows.

Authentication of tags: The verifier can authenticate the yoked tags, because the authenticity of tags depends on successful verification of the PRNG function applied on the secret key, the random numbers challenge $ch = E_{k_V}(timestamp)$, and the artificial error vector. Because only the genuine tag has the key, the authentication is ensured. The proposed protocol requires the tags to generate authenticated commitments, $(\widetilde{V}_A, \widetilde{V}_B, \hat{V}_A)$, based on the verifier's challenge $ch = E_{k_V}(timestamp)$ and their secret keys. Each challenge is random and independent; therefore, only genuine tags can generate correct commitments. This authenticates the tags.

Anonymity and un-traceability: An attacker may try to trace a tag or even derive the secret parameters via the observations of eavesdropped sessions. Such attacks are similar to the message-send attacks or the related-message attacks on the previous ECC-based public key encryption schemes or ECC-based symmetric encryption schemes. For example, an attacker may eavesdrop two or more sessions to have several $(c^1 + e^1, \ ..., \ c^i + e^i, \ ...)$. In the previous ECC-based symmetric encryption schemes like [24], $(c^i + e^i) + (c^j + e^j)$ would be $e^i + e^j < 2t$ if they are from the same source, because $c^i = c^j$ for the same source. Therefore, the condition $(c^i + e^i) + (c^j + e^j) < 2t$ could be used to trace an entity. However, in our yoking scheme, the code words c^i, c^j are non-zero code words generated by random linear combinations of the assigned row vectors. Therefore, the condition $(c^i + e^i) + (c^j + e^j) < 2t$ does not always hold even if they are from the same tag. For each tag with s row vectors, it can generate $2^s - 1$ distinct non-zero code words, and there are totally $(2^s - 1) \cdot l$ distinct code words generated by all the tags. The distance between two distinct code words either from the same tag or not is at least d. Therefore, the distance between two transmissions results from the same tag could be $d + 2t$. However, if the same code word of the same tag is used twice (or more), then the distance of using the same code would be $\leq 2t$, and an

Table 2. A concise summary of related schemes

	Security	Tag overhead	Sever overhead
Bolotnyy-Robins [8]	Vulnerable to replay attack	hashing	Exhaustive search to identify tag
Saito et al. [12]	Not consider anonymity, Vulnerable to replay attack	Hashing,	If anonymity is considered by not sending identity, Exhaustive search to identify tag
Peris-Lopes et al. [11]	Security of genetic programming is not verified	Hashing, genetic programming	Exhaustive search to identify tag
Burmester et al. [2]	The anonymous version is vulnerable denial-of-service attack [34]	hashing	Exhaustive search to identify tag
Our scheme	Anonymity, authentication	PRNG	One Decoding operation

adversary could trace a tag in such conditions. However, the adversary should collect yoking sessions as many as he could, instead of probing a tag and telling whether this tag being the target tag, to trace a tag. Therefore, the probability of collecting two sessions from the same tag which uses the same codeword is very low and the practical damage is, therefore, very limited.

Secure binding to specific time span: Because the challenge *ch* is random and independent, an attacker can generate valid challenges in the right time with negligible probability. Both T_A and T_B would generate their final commitments ($\widetilde{V}_B$ and $\hat{V}_A$ respectively) only when they have successfully verified their designated partner's commitments ($\widetilde{V}_A, \widetilde{V}_B$), which involve the encrypted timestamp, the random error vector, the secret key and the chaining commitment. Each activity is limited to the valid time span. Therefore, the final proof securely binds the data to the specific time span.

4.2 Performance Evaluation

It is noted that, in our scheme, only the servers are required to be equipped with the decoding algorithms (which involve matrix operations). The functions required on the tag are cryptographic PRNG function and simple bit operations of XOR and AND. Compared to the previous schemes (like [2,8,11,12]) where tags are supposed to equipped with hash functions, the computations on the tags in our scheme are very simple and efficient. To achieve anonymity, schemes like [2,11] require server perform exhaustive searching to identify tags. On contrary, the server of our scheme only performs one decoding algorithm to identify tags. This ECC-based scheme achieves excellent performance in terms of security, efficiency, server's maintenance, cost, and robustness. A brief comparison of related schemes is summarized in Table 2. Under the setting of code $C(n, k, d)$ and the number of row vectors per tag being l, the space requirement per tag is $l \cdot n + |K_i|$ bits, and the number of supported number of tags is $\lfloor k/l \rfloor$.

5. Conclusions

In this paper, we have proposed a novel ECC-based yoking proof protocol. The contributions of this paper are three-fold: (1) it is the first ECC-based RFID yoking scheme, (2) it easily protects tag's anonymity, and (3) it requires only simple operations on tags.

References

[1] G. Avoine, E. Dysli, and P. Oechslin: Reducing time complexity in RFID systems, The 12th Annual Workshop on Selected Areas in Cryptography (SAC), 2005.
[2] M. Burmester, B. de Medeiros, and R. Motta: Provably secure grouping-proof for RFID tags, IACR Eprint, October 2007, http://eprint.iacr.org/2007/407.pdf.
[3] H. Y. Chien: SASI: A New Ultra-Lightweight RFID Authentication Protocol Providing Strong Authentication and Strong Integrity, IEEE Transactions on Dependable and Secure Computing 4(4), pp. 337–340, October, 2007.
[4] H. Y. Chien, C.-H. Chen: Mutual Authentication Protocol for RFID Conforming to EPC Class 1 Generation 2 Standards, Computers Standards & Interfaces 29(2) (2007), 254–259.
[5] D. N. Duc, J. Park, H. Lee and K. Kim: Enhancing Security of EPCglobal Gen-2 RFID Tag against Traceability and Cloning, The 2006 Symposium on Cryptography and Information Security, 2006.
[6] G. Avoine: Radio frequency identification: adversary model and attacks on existing protocols, Technical Report LASEC-REPORT-2005-001, September 2005.
[7] A. Juels: Yoking proofs for RFID tag, In Proc. of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshops, pp. 138-142, DC, USA, 2004.
[8] L. Bolotnyy, G. Robins: Generalized "yoking-proofs" for a group of RFID tags, in MOBIQUITOUS 2006.
[9] M. Ohkubo, K. Suzki and S. Kinoshita: Cryptographic Approach to 'Privacy-Friendly' Tags, in RFID Privacy Workshop, 2003.
[10] A. Juels and S. Weis: Defining strong privacy for RFID, Cryptology ePrint Archive, Report 2006/137, http://eprint.iacr.org/.
[11] P. Peris-Lopes, J. Hernandez-Castro, J.M. Estevez-Tapiador, and A. Ribagorda: Solving the Simultaneous Scanning problem Anonymously: Clumping proofs for RFID Tags, Unpublished Manuscript, Carlos III University of Madrid, 2007.
[12] J. Saito and K. Sakurai: Grouping proof for RFID tags, in 19th International Conference on Advanced Information Networking and Applications (AINA) March2005, vol. 2, 621–624.
[13] S. Piramuthu: On existence proofs for multiple RFID tags, in IEEE International Conference on Pervasive Services, Workshop on Security, Privacy and Trust in Pervasive and Ubiquitous Computing SecPerU 2006, Lyon, France, June 2006. IEEE, IEEE Computer Society Press.
[14] S. Piramuthu: Protocols for RFID tag/reader authentication, Decision Support Systems 43 (3) (2007), 897-914.
[15] S. A. Weis: Security and Privacy in Radio-Frequency Identification Devices, Masters Thesis MIT, 2003.
[16] C.-L. Lin and L.-C. Chang: An Efficient AES-Based RFID Authentication Protocol, The 3rd Joint Workshop on Information Security (JWIS 2008), July 10-11, 2008, Hanyang University, Seoul, Korea.
[17] N.-W. Lo, K.-H. Yeh, C.-Y. Yeun: New mutual agreement protocol to secure mobile RFID-enabled devices, Information Security Technical Report (2008), 151–157.
[18] R.S. Safavi-Naini, and J.R. Seberry: Error-correcting codes for authentication and subliminal channels, IEEE Transactions on Information Theory, 37(1) (1991), 13–17.
[19] H. Y. Chien, J. K. Jan, and Y. M. Tseng: An unified approach to secret sharing schemes with low distribution cost, Journal of the Chinese Institute of Engineers, 25(6) (2002), 723–733.
[20] C. S. Park: Authentication protocol providing user anonymity and untraceability in wireless mobile communication systems, Computer Networks 44(2), 267–273.
[21] R. J. McEliece: A public-key cryptosystem based on algebraic coding theory, 1978. Jet Propulsion Laboratory DSN Progress Report 42-44. URL: http://ipnpr.jpl.nasa.gov/progressreport2/42-44/44N.PDF.
[22] D. J. Bernstein, T. Lange, and C. Peters, Attacking and defending the McEliece cryptosystem, Cryptology ePrint Archive: Report 2008/318.

[23] R. Struik, J. Tilburg: The Rao–Nam scheme is insecure against a chosen-plaintext attack, in: Advances in Cryptology –CRYPTO 87, Springer, Berlin, 1988, 445–457.
[24] R.M. Campello de Souza, J. Campello de Souza: Array codes for private-key encryption, Electronics Letters 30 (17) (1994), 1394–1396.
[25] A. Al Jabri: Security of private-key encryption based on array codes, Electronics Letters 32 (24) (1996) 2226–2227.
[26] M. Bellare, A. Desai, D. Pointcheval, and P. Rogaway: Relations among notations of security for public key Encryption schemes, Proceedings of CRYPTO'98, LNCS 1462, 26–45. Springer-Verlag, 1998.
[27] S. Lin, and D.J. Costello, Jr. Error Control Coding: Fundamentals and Applications, Prentice-Hall, Englewood Cliffs, NJ, 1983.
[28] H.-Y. Chien, C.-S. Laih: ECC-Based Lightweight Authentication Protocol with Un-traceability for Low-Cost RFIDs, Journal of Parallel and Distributed Computing 2009, http://dx.doi.org/10.1016/j.jpdc.2009.07.007.
[29] A.J. Menezes, P.C. van Oorschot and S.A. Vanstone, Handbook of Applied Cryptography, CRC Press, BocaRaton, 1996.
[30] D.R. Stinson, Cryptography: Theory and Practice, 3rd Edition, Chapman & Hall/CRC, Boca Raton, 2006.
[31] G. Hancke and M. Kuhn: An RFID distance bounding protocol, in Proc. of the IEEE, SecureComm 2005, September 2005.
[32] J. Reid, J.M. Gonzlez, T. Tang, B. Senadji: Detecting Relay Attacks with Timing-Based Protocols. http://eprints.qut.edu.au/archive/00003264/.
[33] T.A. Berson: Failure of the McEliece public-key cryptosystem under message-resend and related-message attack, in: Advances in Cryptology—CRYPTO 97, Springer, Berlin, 1997, pp. 213–220.
[34] H.-Y. Chien: Matched Yoking Protocol with Forward Secrecy, *submitted to Journal of Internet Technology.*

This page intentionally left blank

Radio Frequency Identification System Security
Y. Li and J. Zhou (Eds.)
IOS Press, 2010

Subject Index

This page intentionally left blank

Radio Frequency Identification System Security
Y. Li and J. Zhou (Eds.)
IOS Press, 2010

Author Index

This page intentionally left blank