JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 26, 1323-1344 (2010)

Active RFID System with Cryptography and Authentication Mechanisms*

HSI-WEN WANG, REN-GUEY LEE, CHUN-CHIEH HSIAO+ AND GUAN-YU HSIEH
Department of Electronic Engineering
National Taipei University of Technology
Taipei, 106 Taiwan
+*Department of Electrical Engineering*
National Taiwan University
Taipei, 106 Taiwan
+*Department of Computer Information and Network Engineering*
Lunghwa University of Science and Technology
Taoyuan, 333 Taiwan

Radio frequency identification (RFID) systems have recently been used in a large number of applications. Security and privacy issues have also imposed significant challenges on these systems. Cryptography and authentication protocols have been utilized to effectively solve the security and privacy problems in RFID systems.

In this paper, we integrate public key encryption, embedded computation, and wireless communication technologies into the active RFID system that we proposed with cryptography and authentication mechanisms. In our proposed active RFID system, a secure RFID Tag intermittently transmits cipher text to a RFID Reader which then transmits in multi-hop relaying to a back-end platform to perform data comparison for authentication. In addition, the digital signature scheme – Tame Transformation Signatures (TTS) has the advantages of high security, high-speed key generation, signature, and suitability to embedded systems and is thus suitable to be used in our authentication system. It is used in our proposed system to protect the plain text. The TTS algorithm is from the family of asymmetric public key systems and thus has superiority such as better security, fast key generation, complex algorithm, and low signature delay. The TTS algorithm can thus encrypt data and perform authentication more effectively in active RFID systems.

We have three major contributions in this paper. The first is to fully design and implement an active RFID system which includes active Tags and Readers. In our system, the Tag can stand by and keep working in long term after getting started. The second is to successfully implement a proposed active RFID system with TTS cryptography and authentication mechanisms to protect the content in tags to ensure the security in multi-hop transmission. The last is to adopt multi-hop relays to extend distance between Readers.

Keywords: RFID, TTS, cipher text, cryptography, authentication

1. INTRODUCTION

Radio Frequency Identification (RFID) technology has been used for more than half a century. The earliest applications are for military purpose. As the technologies of antennas and element processing are developed rapidly, RFID has been applied in numerous applications such as automatic fee collecting systems, animal recognition and tracking,

Received May 30, 2008; revised March 26 & May 25, 2009; accepted July 10, 2009.
Communicated by Tzong-Chen Wu.
* This study is conducted under the Service Ecosystem Research Value Engineering (SERVE) Program of the Institute for Information Industry which is subsidized by the Ministry of Economy Affairs of R.O.C.

and factory automation. While in various RFID application systems, the problems of information security and privacy have become the focus of recent research.

For security and privacy in RFID, increasingly more literatures [1-6] have investigated the problem of protection of tag data. In other words, the goal is to ensure that when the data in a tag is transmitted through the air it will not be stolen away within the reception area of RFID reader. There are two main approaches to provide the security of data in RFID tags. One approach is to encrypt the data and store the encrypted data in the RFID tag. The main purpose is to transform the data (plain text) into some encoded format (cipher text) so that no unauthenticated RFID readers can decode the cipher text. The other approach is to perform the authentications of the reader and the tag. The main purpose is to mutually authenticate the reader and the tag before the data transfer. These two approaches are the two most commonly used methods in the literature.

Both passive and active RFID tags have become much smaller than before. It has thus become challenging to realize the security authentication of data in the miniaturized RFID tags. In the literature, most approaches use various data security algorithms to implement the data encryption or use hardware to accelerate the encryption speed for data in RFID tags [7, 8].

The purpose of this paper is to construct a complete RFID authentication system. This system consists of active RFID tags and multiple RFID readers to perform mutual authentication and data transmission. The data transmitted from tags are encrypted using multivariate public key cryptosystem [8-12] to effectively increase the security and reliability of the data. The signals transmitted by every single active RFID tag contain encrypted cipher text. Each RFID reader receives the cipher text from the tags that are within the reception area of the reader. The cipher text is then transmitted from back end to front end via multi-hop relaying mechanism between readers. The front-end computer is integrated with key center and database to retrieve the original data to ensure the data security.

To enable smooth decoding of cipher text in authentication center, the public key algorithm has played an important role in the process. This algorithm has to be with high security, with high encryption and decryption speed, and suitable for embedded systems to let the authentication center perform data authentication for various number of RFID tags [8].

The organization of the rest of this paper is as follows. We first survey the related works of our proposed system in section 2. The overall system architecture of our RFID authentication system is then demonstrated in section 3. In section 4, the public key certificate (PKC) Tame Transformation Signatures (TTS) algorithm used in our system is described. Section 5 illustrates the design of our RFID system including tags and readers. The implementation and evaluation of our proposed system is then given in section 6. Finally, section 7 concludes this paper and gives some future development directions.

2. PROS AND CONS OF PREVIOUS WORK

RFID security system has recently become a popular research topic. In current commercial applications, passive RFID tags have been frequently embedded into products since the passive tags are light and small such that they can be embedded easily into prod-

ucts. However the passive tag could be vulnerable to several restrictions such as the inability to use flexible security algorithm and authentication system. Even under such restricted environment many encryption and authentication protocols have been developed for passive RFID tags [5, 13]. On the other hand, the active tag consists of a microprocessor so it can much more flexibly adapt to various data security algorithms and authentication systems. However the active tag also possesses disadvantages such as the need of attached battery and larger form factor.

To integrate RFID systems with authentication technologies and cryptography algorithms is the current hot topic for researchers [4]. Researches and discussions of attacks to these technologies and algorithms have also been carried out to increase the practicability of RFID technologies. Beside the problems of security and authorization, research issues also include the trackability and privacy protection of RFID tags [6]. The use of symmetric or asymmetric algorithms together with hash function to encrypt data [8, 12, 14-18] depends on the complexity of the algorithms and the computational ability of the microprocessors in tags. It is also possible to use hardware to provide protection mechanisms such as to perform transmission interference to the unauthenticated RFID readers. The research teams in the literature have proposed related methods as follows [7, 17].

2.1 Kill Command Idea

In commercial applications, when products are purchased, ID center will transmit a special command "Kill command" to the Tag attached to the product. After the tag receives the special command, it will delete the content information in it to prevent the intruder from reading the related production information by using illegal readers.

2.2 Faraday Cage Approach

In this approach, to prevent illegal reader from reading tag information the tag is placed in a magnetically shielded container. The drawback of this approach is that it is easy for the thief to utilize the shielded container to avoid the detection of the legal reader so as to steal the content together with its shielded container.

2.3 The Active Jamming Approach

This approach produces magnetic waves that can interfere with illegal readers so as to destruct the correctness of the data read by the illegal readers. However if the broadcasting power is too high, it can even interfere the legal RFID system. If this approach is applied in the hospital or in the mass transportation system, more severe damage can be incurred.

2.4 The Blocker Tag Approach

In this approach, every tag possesses reader's public key and its own private key. The reader and the tag have to perform mutual authentication via these two keys. It is hard for the illegal user to acquire the keys in a short time so as to steal the data. In the public key algorithm, the tag should have high computation power to deal with the heavy com-

putation load for complex algorithms. The tag can be increased in size due to the addition of hardware to accelerate the algorithm computation and thus can not be embedded into certain products.

3. SYSTEM ARCHITECTURE

The system architecture of our proposed system is shown in Fig. 1. The whole system consists of three main parts, namely, active RFID tag, RFID reader, and back-end processing platform. The back-end processing platform includes two parts: key center and database system. Both the RFID tags and RFID readers are composed of power-saving microprocessor MSP430 and low-power RF module Nordic. The RFID reader is equipped with power management function so that it can be powered by AC power as well as alkaline batteries to provide more flexibility to our proposed system.

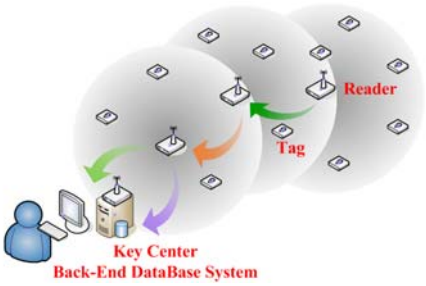

Fig. 1. RFID system architecture.

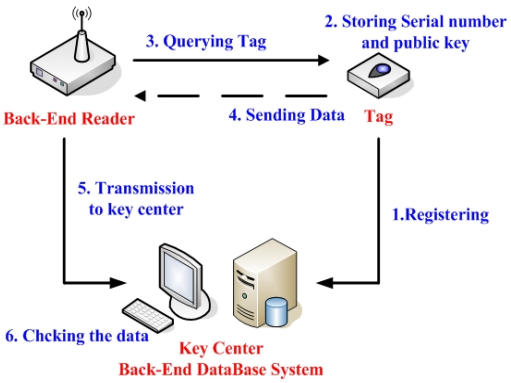

Fig. 2. The setup flow diagram.

The RFID reader plays an important role in our proposed RFID system. In the wireless transmission network, the RFID reader not only can receive the data in the tag within transmission range but also can relay the received and collected data to the next reader closer to the back-end processing platform. In this multi-hop relaying mechanism, the data can eventually be transmitted to the back-end processing platform. The last reader that is connected to the back-end platform is responsible to convert the received data packets to RS-232C frames and to send the data to the key center and the database system for further processing. The operation of the wireless transmission network will be furthered described in detail in section 5.3.

The complete encryption and authentication system has to be implemented by combining Figs. 2 and 3. The authentication protocol of our system is shown in Fig. 3. We have modified the portion of authentication code in O-TRAP protocol [19] that varies synchronously between database and tags since the synchronization scheme can be easily attacked and subsequently cause the loss of connection between system and tags [6]. We have also added a random code in the tag side to prevent tracking by attackers. Before being distributed to any area, the tags in Fig. 1 have to go through the following steps.

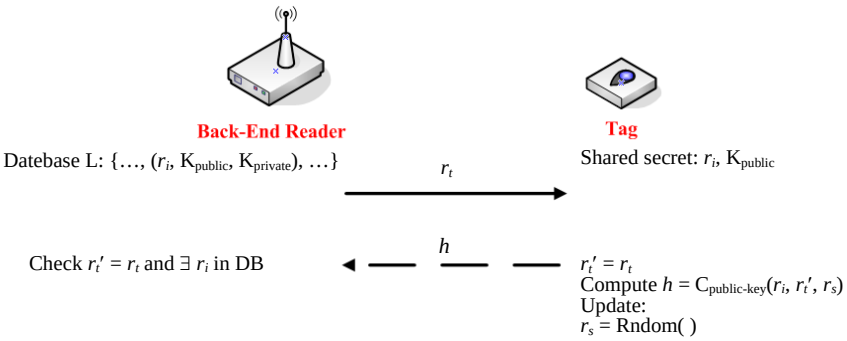

Fig. 3. Our authentication protocol which is modified based on the O-TRAP protocol.

(1) The tag has to be registered to the back-end processing platform before it is distributed. The key center will record the information of the tag r_i, the public key K_{public} and private key $K_{private}$ in the database in the back-end processing platform.
(2) After the registration the tag will store the registered sequence number r_i and the public key K_{public} in its memory that will be used to encrypt the plain text message sent back to the reader.
(3) The reader will include in the challenge command a random number r_t that will be used to authenticate the tag. The tag will include this number in its plain text that will be sent back after encryption.
(4) After receiving the challenge command from the reader the tag extract the random number r_t in it. The r_t will then be combined with the registered sequence number r_i and a random number r_s that is generated by the tag itself as the content of plain text. The plain text will then be encrypted using the public key via $h = C_{public\text{-}key}(r_i, r_t, r_s)$. The cipher text h will then be transmitted to the reader. The random number r_s is included in the plain text so as to produce different encrypted packets even if the tag receives the same packets every time to prevent tracking from attackers using forged reader packets.
(5) After receiving the response packets from the tag the reader will transmit the packets to the back-end authentication server via multi-hop relaying mechanism.
(6) After receiving the cipher text h transmitted from the reader the back-end server will use the private key $K_{private}$ to decipher it and to verify whether the registered sequence number r_i and the random number r_t in the plain text are the same as the one in database and the one transmitted by the reader respectively. If they are indeed the same then it can be concluded that the data is really transmitted by a legal tag and subsequently terminate the whole authentication process.

For the implementation part of our system in this paper, the back-end platform computation program is written with Borland C++ Windows Program, while the RFID tag and readers are implemented by using IAR Embedded Workbench development tools. The system implemented in this paper mainly consists of five parts: public key algorithm, RFID reader, RFID tag, back-end processing program, and wireless transmission network design. The details of these five parts will be depicted in the following sections.

4. PUBLIC KEY CERTIFICATE (PKC) TAME TRANSFORMATION SIGNATURES (TTS) ALGORITHM

In the literature there have been many encryption algorithms and hash functions applied to RFID system. For public key systems, RSA is the most popular and the most adopted algorithm. It has the advantage of high security and the disadvantage of high computational complexity. Due to the need of high computational power, RSA is thus not suitable for applications in embedded systems. To provide adequate security and fast encryption and decryption, we propose a Tame Transformation Signatures (TTS) algorithm for a multivariate public key cryptosystem. This TTS algorithm has the advantages of high security, high-speed key generation, signature, and suitability to embedded systems and is thus suitable to be used in our authentication system. The detailed computational core will be discussed in the following subsection.

4.1 Tame-like PKC

TTS is a very efficient encryption algorithm based on Tame Transformation Method (TTM) [20, 21]. TTS mainly utilizes a set of multi-variable multi-order simultaneous equations formed by specific variables to construct a key tame-like function using the correlated terms in the simultaneous equations. This key tame-like function is the central map or kernel in the multivariate system [21]. The central map or kernel determines the security for the whole algorithm and is also the most critical part that is vulnerable to the attacks of invaders.

The basic computation of public keys is based on K finite field where K is defined as the limit of values in the computation process. A pubic map function mainly consists of three parts. The surjective function is defined as $V = \phi_3 \circ \phi_2 \circ \phi_1$ where ϕ_1: $w \mapsto x = M_1 w + c_1$ and ϕ_3: $y \mapsto z = M_3 y + c_3$ are two affine and invertible functions while ϕ_2: $K^n \to K^m$ is a tame map which includes a set of multi-variable multi-order simultaneous equations formed with specific parameters. The parameters M_1, M_3, c_1 and c_3 in functions φ_1 and φ_2 are matrices with randomly generated binary elements. The foundation of the security of this mechanism is based on an NP-hard algorithm [24]. This algorithm solves the problem of partitioning V composite function by using element ϕ_2 in a set of complex second-order simultaneous equations. The alteration of n and m dimensions can change both the speed of generating keys and the speed of partitioning ϕ_2 in this algorithm.

We now use a simple example to demonstrate the operation flow of the whole TTS algorithm. In this example we use TTS(3, 5) as the security level.

4.2 TTS(3, 5) Dimension Example

The public map we use is $V = \phi_3 \circ \phi_2 \circ \phi_1$: GF($2^5$) $\to$ GF(2^3) where ϕ_3, ϕ_2, and ϕ_1, are as follows.

$$\phi_3 : \begin{bmatrix} z_0 \\ z_1 \\ z_2 \end{bmatrix} = M_3 \begin{bmatrix} y_2 \\ y_3 \\ y_4 \end{bmatrix} + c_3 \tag{1}$$

$$\phi_2 : \begin{aligned} y_2 &= x_2 + a_2\ x_0\ x_1 \\ y_3 &= x_3 + a_3\ x_1\ x_2 \\ y_4 &= x_4 + a_4\ x_2\ x_3 \end{aligned} \tag{2}$$

$$\phi_1 : \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = M_1 \begin{bmatrix} w_0 \\ w_1 \\ w_2 \\ w_3 \\ w_4 \end{bmatrix} + c_1 \tag{3}$$

In this example, the computation is based on GF(2) finite field in which logic 1 and logic 0 is used as the unit for computational process. Assign arbitrary value to c_1 and choose specific matrices M_1 and M_3. Use LU factorization method to compute the inverse matrices M_1^{-1} and M_1^{-3}. Through the relationship between c_1 and M_3, we can compute a specific value for c_3. We can thus acquire the conditions to produce the private key.

Assume the test matrices are as follows.

$$c_1 = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}; M_1 = \begin{bmatrix} 10011 \\ 11010 \\ 10100 \\ 11111 \\ 01010 \end{bmatrix}; M_3 = \begin{bmatrix} 111 \\ 101 \\ 110 \end{bmatrix}$$

We can combine c_1, M_1 and M_3 to produce $c_3 = (0, 1, 0)$ and $z = V(w)$. Note that $w_i^2 = w_i$ is in the finite field GF(2). From Eq. (3), the known conditions – M_1 and c_1 matrices, and the unknown cipher text matrix w, x matrix can be obtained as Eq. (4).

$$\begin{aligned} x_0 &= w_0 + w_3 + w_4 + 1 \\ x_1 &= w_0 + w_1 + w_3 + 1 \\ x_2 &= w_0 + w_2 \\ x_3 &= w_0 + w_1 + w_2 + w_4 + 1 \\ x_4 &= w_1 + w_3 \end{aligned} \tag{4}$$

Where x matrix consists of some cipher text variables. By substituting these variables into ϕ_2 composite function, we can get Eq. (5).

$$\begin{aligned} y_2 &= (w_0 + w_2) + (w_0 + w_3 + w_4 + 1)(w_0 + w_1 + w_3 + 1) \\ &= w_1 + w_2 + w_3 + w_4 + w_0 w_1 + w_0 w_4 + w_1 w_3 + w_1 w_4 + w_3 w_4 \\ y_3 &= (w_0 + w_2 + w_2 + w_4 + 1) + (w_0 + w_1 + w_3 + 1)(w_0 + w_2) \\ &= w_0 + w_1 + w_4 + w_0 w_2 + w_0 w_1 + w_0 w_3 + w_1 w_2 + w_2 w_3 \\ y_4 &= (w_1 + w_3) + (w_0 + w_2)(w_0 + w_1 + w_2 + w_4 + 1) \\ &= w_1 + w_3 + w_0 w_1 + w_0 w_4 + w_1 w_2 + w_2 w_4 \end{aligned} \tag{5}$$

As shown in the ϕ_2 polynomials in Eq. (2), the main principle is to as much as possible let every variable appear in the each quadratic equation to assure the difficulty of

Gaussian Decomposition. This is also why tame map is hard to crack. There are special methods to process the variables that do not appear in the quadratic equations [24].

Now we look at the ϕ_3 composite function. Since the M_3 and c_3 matrices are already known, Eq. (6) can be derived from Eq. (1) as follows. As shown in Eq. (6) every z variable can be produced as the combination of many cipher text variables w's. This is also the way to produce public keys.

$$\begin{aligned} z_0 &= w_0 + w_1 + w_2 + w_3 + w_0w_1 + w_0w_2 + w_1w_3 + w_1w_4 + w_2w_4 + w_3w_4 \\ z_1 &= w_2 + w_4 + w_0w_3 + w_1w_2 + w_1w_3 + w_1w_4 + w_2w_3 + w_2w_4 + w_3w_4 \\ z_2 &= w_0 + w_2 + w_0w_2 + w_0w_3 + w_0w_4 + w_1w_2 + w_1w_3 + w_1w_4 + w_2w_3 + w_3w_4 \end{aligned} \tag{6}$$

The public key is produced by using the combination of w variables in Eq. (6), while the private key is produced by using c_1 and c_3 variables and M^{-1}_1 and M^{-1}_3 inverse functions. Assume z = (1, 1, 0). We now calculate the set of w, w = (w_0, w_1, w_2, w_3, w_4) to satisfy Eq. (6). Finally we use the following three steps to solve the third-order polynomial to produce signature $S(z) = \phi_1^{-1}(\phi_2^{-1}(\phi_3^{-1}(z)))$, $y = M_3^{-1}(z - c_3) = (1, 1, 1)$.

It is obvious that the values of x are not unique. We thus have to assume the values of x_0 and x_1, then the four possible values of x would be x: (0, 0, 1, 1, 0), (0, 1, 1, 0, 1), (1, 0, 1, 1, 0), (1, 1, 0, 1, 1).

Since we can get $w = M_1^{-1}(x - c_1)$ from Eqs. (3)-(7), the four possible values for w would be (1, 1, 0, 1, 1), (1, 0, 0, 1, 1), (1, 0, 0, 0, 1), (1, 1, 1, 0, 1). The four possible results of signatures can all match the original information. The signature verifier has to verify all these four results. The hash values also have to satisfy $z = V(w)$.

In this simple example, the encryption and decryption do not seem to need complex computation; however it is not the case. In fact, V can not be easily partitioned by using initial parameters. It is also difficult to calculate w from real-size parameters. It is thus not easy to forge a secure signature.

4.3 Design of TTS in Authentication System

In the implementation of this paper, we use security level of TTS(8, 10) dimension. There are several reasons for using low dimension security level. The first reason is that since the size of a frame transmitted by the RF module has a 25-byte limit, only the cipher text produced by using low security level such as TTS(8, 10) to encrypt the data can satisfy this limit. The second reason is that if over-size dimension is used, when surrounding tags continuously transmit data, the RFID reader might not have the capability to receive cipher text with larger data volume and the success rate of the back-end platform to verify the signature can thus be decreased. The last reason is that by extension to the second reason, computation of higher dimension can increase the load of signature verification to the back-end platform; it is thus in question whether the back-end platform is fast enough to verify the signature for every cipher text in time.

By the above mentioned analysis and considerations we choose dimension of TTS(8, 10). Its central map is ϕ_2: $x = (x_0, x_1, \ldots, x_9) \mapsto y = (y_2, y_9, \ldots, y_9)$. The polynomial representation of this central map is shown in Eq. (7).

$$y_i = x_i + p_i x_{i-1} x_{i-2}, \text{ for } i = 2 \text{ and } 9 \tag{7}$$

This central map uses 8-byte hash code and 10-byte signature which is much simpler than the polynomial proposed in [25]. With this security level, it is secure enough to resist the stealing of malicious intruders.

The TTS public key encryption algorithm has been internationally certified as one of the high security-level encryption algorithms. Its advantages of high-speed key production, high security level, and suitability for embedded systems will definitely result in the increasing number of applications of it to embedded systems in the current society which highly emphasizes the security of data.

Besides, as compared to traditional asymmetric key RSA algorithm, the speed of various processes such as key production and signature verification of TTS is better. The main reason is that the security of RSA highly relies on the difficulty of prime number factorization of a large integer (ex. 1024-bit). Both public key and private key are functions of large prime numbers (more than 100 decimal digits). The difficulty of deriving plain text from a key and cipher text is equal to that of factorizing two large prime numbers from a product. The disadvantages of RSA are thus as follows. It is troublesome to produce keys. With the restriction of prime number generation technology, it is thus hard to encrypt for every message. The length is too long. To assure security, the product of two prime numbers should be at least 600-bit long. This will highly increase the computational complexity and thus decrease the processing speed which could be several order magnitude slower than the symmetric encryption algorithms. TTS adopts function combination of polynomials to increase the security level. Higher dimension of TTS has more complex combinations of polynomials and it is thus harder to derive TTS's central map. TTS is thus with better mathematical and computational characteristics than RSA.

Table 1. TTS(*m*, *n*) = hash and signature sizes comparison table for TTS(*m*, *n*) = hash and signature sizes.

TTS(*m*, *n*)	Rank	RSA (bits)	ECC (bits)
16, 22	2^{87}	512	112
20, 26	2^{88}	768	128
20, 28	2^{120}	1024	144
24, 32	2^{121}	1536	160
24, 34	2^{153}	2048	176
28, 38	2^{154}	2560	192
28, 40	2^{186}	3072	208
32, 44	2^{187}	4096	224
32, 46	2^{218}	5120	240
36, 50	2^{219}	6122	256

Table 1 shows the security levels of TTS in specific dimensions corresponds to the other algorithms. Take TTS(20, 28) as an example. Its security level equals to the RSA 1024-bit encryption level that is used by current credit card encryption. To classify the security levels, specific attacks are used in the literature to calculate the time needed to crack the algorithms. We can then classify the algorithms with the same time into the same type. We can thus conclude that TTS algorithms are as secure as RSA algorithms. The superior security and processing speed are the main reasons that we utilize TTS in our

proposed system. Bo-Yin Yang *et al*. [8] has pointed out that the computation time of enTTS(20, 28) is 0.044 seconds under 100kHz working frequency utilizing TSMC 0.25 μm process.

5. DESIGN OF RFID SYSTEM, READERS AND TAGS

5.1 Design of RFID Tags

The 16-bit RISC MSP430F1121A from Texas Instruments is used for the main control chip in our RFID tag to be in charge of data computation and transmission operation. The ISM band RF transmission module nRF2402 from Nordic is used for wireless transmission in physical layer. For long-time operation of RFID tags, Hall-effect switch is used in the RFID tag. Before registration to authentication system, the RF transmission module is in hibernation mode. After the registration, a magnet is used to input trigger signal to initiate the RF transmission module. Besides, MSP430 is set to work in LPM3 power-saving mode (MHz 0.9 μA) and the internal clock interrupt in the main control chip is used to transmit data frames only at predetermined time to get the best power-saving effect. For special purposes, the design of external button switch can be used to send special signals to RFID reader for extra processing. Since the miniaturized RFID tags have been used for many applications, button cells with very low ripple noise in voltage are used to provide a stable working voltage of 3V for every block. The block diagram of our prototype circuit is shown in Fig. 4.

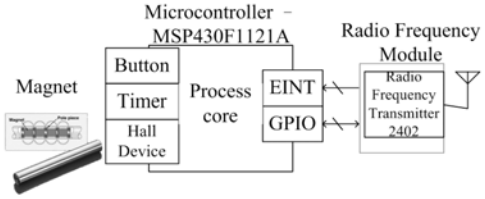

Fig. 4. Block diagram of our prototype RFID tag circuit.

5.2 Design of RFID Readers

A 16-bit RSIC MSP430F149 chip from Texas Instruments is used as main control chip in our RFID reader to perform communication protocols for data acquisition and network transmission. An ISM band RF transmission module nRF2401A from Nordic is also used to realize the wireless transmission in physical layer. RFID readers can be deployed at fix positions or can be used as mobile devices. When deployed at fixed positions, RFID readers are powered by indoor transformers (1A, 5V). While in applications that have to be powered by batteries, 4 AA batteries can be used for power source thanks to the low power consumption characteristics of and the usage of power-saving modes of MSP430. To provide a stable working voltage of 3.3 V for each subsystem, a low noise low dropout (LDO) linear regulator IC is used to ultimately reduce the power ripple noise.

The Microcontroller MSP430 basically uses GPIO protocol to control a set of

nRF2401A to utilize two channels to receive tag signals. One channel is reserved for receiving the frames from tag within the receiving range. The other channel is used to receive the frames from the front-end reader. As for nRF2402, MSP430 uses clock interrupts to send data frames at fixed time to the next reader. The sent data frames will include the tag information received by the reader itself and the data frames sent by the previous readers to achieve the purpose of multi-hop transmission. The block diagram of our prototype RFID readers is shown in Fig. 5.

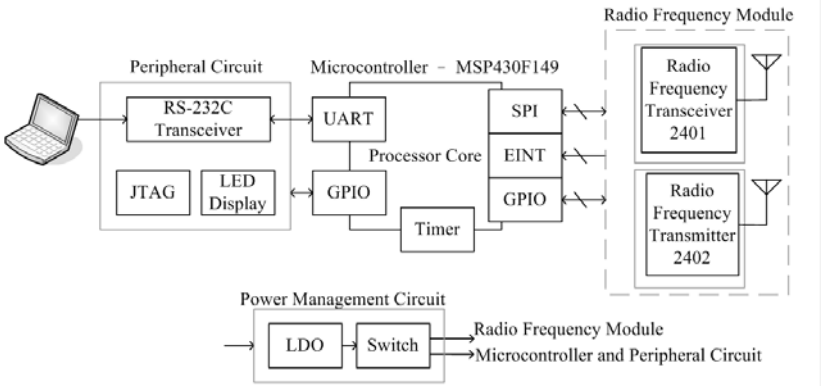

Fig. 5. Block diagram of our prototyped RFID readers.

Fig. 6. Demonstration of multi-hop relaying.

5.2 Design of Wireless Transmission Network

We will discuss the planning for the wireless network for RFID readers in this subsection. The data frame transmission is performed by multi-hop relaying between RFID readers. To assure good communication between tags and readers and between readers and readers, we have designed and implemented transmission protocols used in the network. The multi-hop relaying used in our proposed RFID network is shown in Fig. 6.

Every reader is equipped with two RF transmission modules, namely, nRF2401 and nRF2402 transmission modules, with the exception that the terminal reader has only one nRF2401 module. Though the nRF2401 module has both capabilities of transmission and reception, one nRF2402 module is also equipped to perform multi-hip relaying of the tag data since in our system the data transmission and reception is performed without time synchronization between every reader. The main consideration is to provide the reader

with high reception performance. The RFID reader will switch on the transmission module according to clock interrupts. It will adversely affect the transmission efficiency if the switching of reception or transmission is performed in the same module. That's why nRF2401 only deals with the reception of data frames while nRF2402 which is only capable of transmitting data performs the transmission via multi-hop relaying.

In this paper, we use the channel 2 in nRF2401 module to receive the data transmitted by surrounding tags. The data from the tags will be stored in the software stack in the readers. Before the data reception the software stack will first have to be initialized. The data received by the reader will be sequentially stored in the buffer of the software stack. The software stack pointer will point to the location of the last stored data and thus can be used to indicate the number of data that have been stored in the software stack. Only when the software stack pointer points to the upper limit location of the software stack, the nRF2402 module will be used to transmit the data in the software stack via multi-hop relaying. The already transmitted data will be moved out of the software stack and the software stack pointer will be updated accordingly for the next reception of tag data.

When the channel 2 in nRF2401 module is receiving data from surrounding tags, the channel 10 in nRF2401 can receive information from the previous reader which might contain the data received by channel 2 from the nRF2401 in the previous reader and also contain the data from even further readers. Due to the limitation of the size of packets that can be transmitted by RF modules, the nRF2402 modules in the readers will transmit two sets of data to the next reader. One is the tag data received by the nRF2401 module in the same reader while the other is the data randomly selected from the data transmitted from the previous readers.

5.4 Design of Key Center and Back-End Platform

In this paper, we use Wintel as the execution platform and use Borland C++ Builder 6.0 as integrated development tool to develop our programs. Fig. 7 exhibits the workflow of personal ID encryption in key center. At first, system utilize M_1^{-1}, M_2^{-1}, c_1 and c_3 to generate both private and public keys in the key center, then use these two keys to calculate cipher text from plain text, and finally store the private ~~key~~ and public keys, plain text and cipher text in the key center. Before deployment the tags have to register in the key center and save the encryption ID. Once the cipher text is received from the reader, key center begins to calculate the personal ID information and check the random number in the cipher text. As shown on the right side in Fig. 7, the digital signature is obtained by three sequential steps; first computing $y = \phi_3^{-2}(z) = M_3^{-1}(z - c_3)$, then finding the inverse map $x = \phi_2^{-1}(y)$, and finally computing the digital signature $w = \phi_1^{-1}(x) = M_1^{-1}(x - c_1)$.

6. SYSTEM IMPLEMENTATION AND PERFORMANCE EVALUATION

In this section we demonstrate the implementation results of our active RFID and authentication system. We also perform measurement and analysis for various parameters that are related to the system performance and further discuss the practicability of our system.

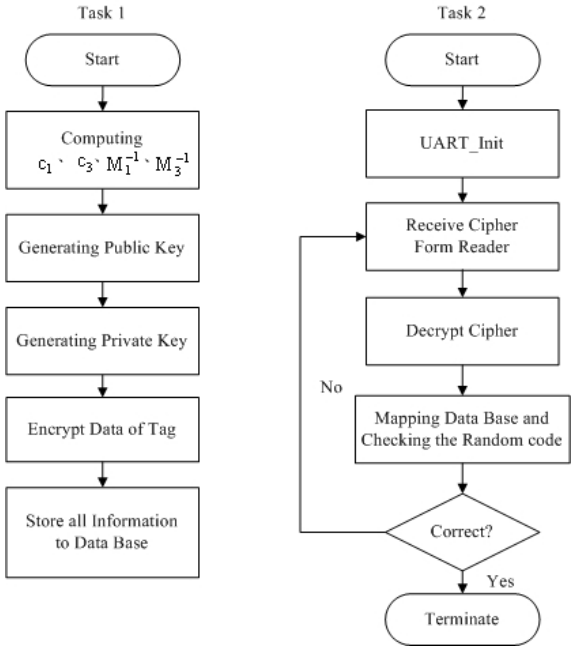

Fig. 7. Workflow of the ID encryption in key center.

6.1 Implementation of Authentication System

In this subsection we will illustrate the function design of key center back-end platform. The main functions of key center back-end platform are shown in Fig. 8 and are as follows.

(1) Selection of Communication Ports. The back-end platform is equipped with a reader to receive data. The key center provides the choices of serial ports for more flexibility and adaptability.
(2) Test of Performance Computation. This provides the time computation test for various public keys, private keys, signatures, and signature verifications.
(3) Storage of Database. The basic data of RFID tags and the produced corresponding public keys and private keys are stored together in the database system.
(4) Selection of Security Levels. Appropriate security levels can be selected for various applications. The time assumed for producing public keys, private keys, signatures, and signature verifications in selected security levels can be computed by function 2.
(5) Real-time Window Display. Windows for receiving signal and message display can be set. The reception window will show the tag information received by back-end reader, while display window will show the prompt messages for all the operations in the key center to allow the users to understand the current operation flow.

The complete authentication system flow is accomplished by combining the above mentioned operations in the key center with our active RFID system.

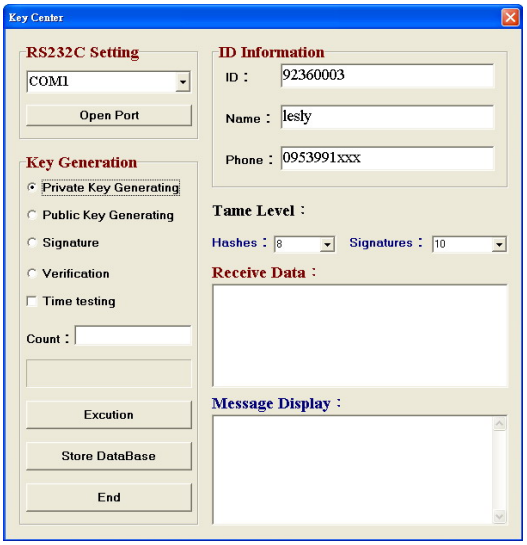

Fig. 8. Key center back-end platform.

6.2 Security-Level Evaluation of Key Production, Signature, and Authentication Subsystems

In this subsection we will evaluate the time consumption for key production, signature, and signature verification of various TTS dimensions, namely, TTS(8, 10), TTS(12, 16), TTS(16, 22), TTS(20, 28), TTS(24, 34) and TTS(28, 40). The resulting data from the evaluation can be used to choose appropriate security levels for specific applications. The following test platform for data analysis is the same as the one described in section 5.

Time Evaluation for Private Key Production: Since the choice of different compilers can affect the performance of mathematical calculation, we take the average value of key production time for ten thousand times to assure the correctness of our results. Fig. 9 shows the time needed for private key production for various security dimensions. As can be observed in Fig. 9, the time needed for producing private keys for low dimension security level is less than that for high dimension security level.

Time Evaluation for Public Key Production: The public keys and private keys are interdependent. The computation of public keys is more complex than that of private keys and thus will consume more time. For accuracy, the time evaluation of public key production is similar to that of private key production. The private keys are produced ten thousand times and the average value is taken also for correctness. Fig. 10 shows the time needed for public key production for various security dimensions.

Time Evaluation for Signatures: After the pubic keys and private keys are produced, the keys can be used to encrypt the data. The security of data will be better if the security dimensions are higher. The encryption time needed will however be longer too. The time needed for signatures in various security dimensions is shown in Fig. 11.

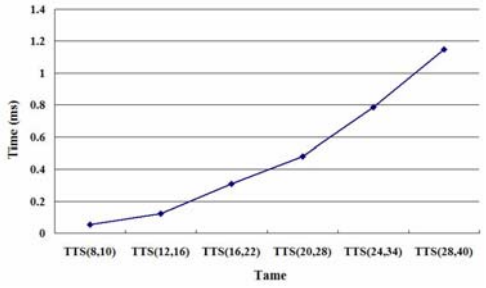

Fig. 9. Time needed to produce private keys for various dimension security levels.

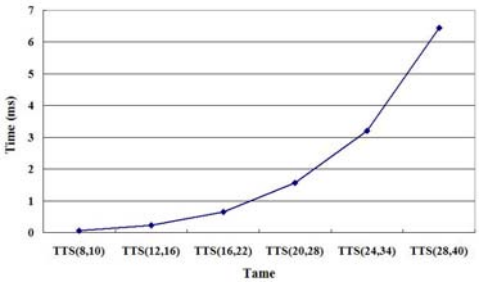

Fig. 10. Time needed to produce key production for various security dimensions.

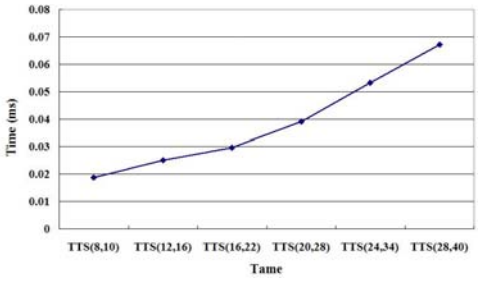

Fig. 11. Time needed for signatures for various security dimensions.

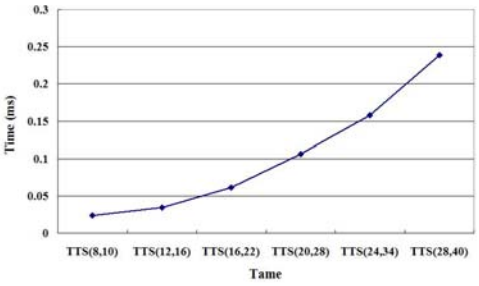

Fig. 12. Time needed for signature verification for various security dimensions.

Time Evaluation for Signature Verification: This time evaluation part is for signature verification. The time assumed in this part will affect the signature verification speed for back-end readers to receive the RFID tag data and then compare the data with the data in the database. The faster the signature verification speed is, the more RFID tag data can be received. Fig. 12 shows the time needed for signature verification for various security dimensions.

6.3 Implementation of RFID Tags

In this paper, we integrate microprocessor, wireless RF module, Hall component, and external event pushbutton into an active RFID tag. The microprocessor can control peripheral components to let them enter power-saving mode and can also use GPIO to transmit data via the wireless RF module. The wireless RF module has the characteristics of high transmission efficiency (1M bps), specific packet frames, multi-channel frequency-hopping ability, and good immunity to noise in the air. The design of Hall component is to effectively control the peripheral circuits to enter low-current working mode from sleeping mode to effectively save the power consumption. The external event pushbutton allows external trigger to start special commands to RFID reader to proceed with exception judgment and thus allows more flexible usage of RFID tags. Fig. 13 shows our implemented active RFID tag and the peripheral operation components.

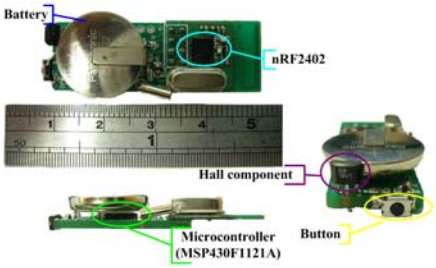

Fig. 13. Appearance and dimension of our active RFID tag.

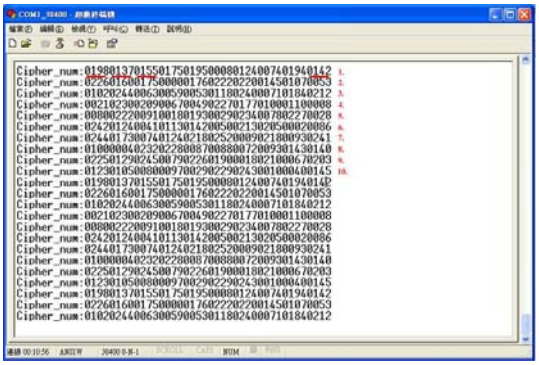

Fig. 14. Cipher texts transmitted by active RFID tags.

The RFID tag that has been registered by the authentication system will transmit cipher text encrypted by using the keys produced by the key center instead of the original ID number. The content of the cipher texts are all random codes in GF(28) finite field. Fig. 14 shows the cipher texts transmitted by active RFID tags. Limited by the message display window size of the key center we currently show 10 sets of cipher texts that are transmitted by the active RFID tags.

In the environment of TTS(8, 10), each set of cipher text is hashed by using the 10 elements in the GF(28) finite field. As can be seen in Fig. 14 the 4 underlined digits represent an element. In this example, 10 tags are used and thus 10 sets of hashed code are shown.

6.4 Evaluation of Power Consumption for RFID Tags

To prolong the usage time of our RFID tags, low power consumption MSP430 is chosen as the microprocessor for processing core. Low power consumption wireless RF module is also used for integration. The wireless RF module transmits a set of cipher text every 0.5 seconds.

In our implementation we use ShockBurst transmission mode which can provide transmission speed up to 1Mbps for power consumption analysis. The total power consumption is shown as follows.

$$E_{total} = V \times I_{active} \times T_{active} + V \times I_{stand\text{-}by} \times T_{stand\text{-}by} \tag{8}$$

The final power consumption of our RFID tag is 0.0354 mW. If our RFID tag is powered by button cell CR-1632 (3V, 125mAh), the duration can be up to 440 days. Active RFID tags with such low power consumption can thus be used in many more applications.

6.5 Implementation of RFID Readers

In our implementation, the active RFID reader mainly consists of a microprocessor, two sets of wireless RF modules (one with reception and transmission capabilities while

the other with only transmission capability), power management IC and RS-232C. There are basically two types of active RFID readers in this paper. One is the back-end reader that connects to the back-end computer. The other is the ordinary reader that relays and collects the data from surrounding RFID tags. The back-end reader does not need the nRF2402 RF module used in the ordinary readers. The back-end reader is also equipped with a RS-232C D-type connector to connect to the back-end platform. The other parts of the back-end reader are the same as the ordinary readers. Fig. 15 shows the appearance and the measurement of the back-end reader, while Fig. 16 shows the 3 RFID readers implemented in this paper.

Fig. 15. Appearance and dimension of our back-end reader.

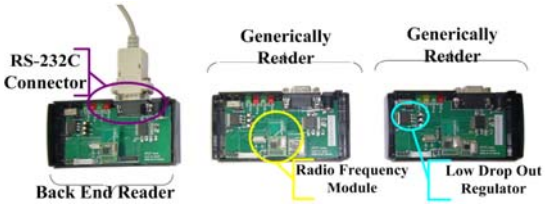

Fig. 16. Deployment of 6 RFID readers.

6.6 Analysis of Successful Read Rates

In this subsection we will evaluate the successful read rates of the back-end reader to receive the cipher texts. We also compare the content of the software stack in the back-end reader with the content in the back-end database. The content of the software stack is the received cipher texts from the front-end readers and thus are cipher texts with random distribution. The data in the software stack has to be transmitted to back end via rapid transmission of RS-232C for data comparison so that the space in the software stack can be saved for the next data reception. In this experiment, 10 sets of tags are used to continuously transmit cipher texts every 0.5 seconds. The successful read rates are calculated for the back-end reader after 1000 receptions of the cipher texts. The successful read rates of the back-end reader under baud rate of 38400 bps and 19200 bps are shown in Fig. 17 respectively.

As can be observed in Fig. 17, the successful read rates are about 96%-98% under the baud rate of 38400 bps. However in Fig. 17, the successful read rates are all 100% under the baud rate of 19200 bps. We can thus conclude that when very high transmission speed is not necessary, lowering the transmission speed can yield very good successful read rates. The reason why the successful read rates under the baud rate of 38400 bps is lower than that of 19200 bps is as follows. The CPUs in tags and readers have to perform CRC and checksum computation beside data transmission. When the baud rate is raised to 38400, occasionally the computation power of the CPU is not sufficient to complete the whole data reception process and cause errors when the packets are received or transmitted.

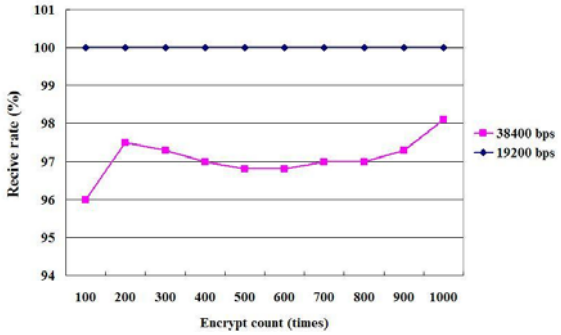

Fig. 17. Successful read rates for the back-end reader.

The packets are thus discarded and the successful read rates are subsequently decreased. However if the data volume is large, the choice of high transmission speed is inevitable to receive and transmit the data to the back-end platform for further processing. There is thus a compromise between high transmission speed and high successful read rate.

6.7 Evaluation of Power Consumption for RFID Readers

In this paper, to accommodate our RFID readers to various environments, the RFID readers can be switched to be powered by utility power or by ordinary batteries. MSP430 with low power consumption is chosen as the microprocessor for processing core. The low power-consumption wireless RF module is also chosen for long-term operation without the consideration of time synchronization.

Both the two wireless RF modules used by our RFID readers adopt ShockBurst transmission modes at a transmission speed of 1Mbps for analysis of power consumption. The total power consumption can be derived as follows.

$$E_{total} = V \times ((I_{RF2041\text{-}active} + I_{RF2402\text{-}active}) + I_{MSP430\text{-}active}) \times T_{active} + V \times ((I_{RF2401\text{-}stand\text{-}by} + I_{RF2402\text{-}stand\text{-}by}) + I_{MSP430\text{-}stand\text{-}by}) \times T_{stand\text{-}by} \quad (9)$$

The final power consumption of a RFID reader is 58.485 mW. If the RFID reader is continuously powered by 4 AAA batteries (1.5V, 2700mAh each), the duration can be up to 12 days. RFID readers with such low power consumption can be used in environments without utility power and can be used in more extended application areas.

6.8 Security and Privacy

In this subsection, we will discuss the security and privacy issues when the proposed system is under three different kinds of attacks, namely, sniffing, spoofing, and tracking.

- Sniffing: If an eavesdropper intercepts the packets transmitted between readers and tags, unless he can acquire both the private and public keys, he would not be able to access the content of the tag data since the tag data are all encrypted via TTS(8,10).
- Spoofing: If an attacker overhears the communication of a read and a tag and records the packet data, the recorded tag packet can then be used in a try to deceive the server

when the reader transmits a challenge request. The server can however utilize the scheme of comparing the random numbers in the recorded ciphered packets with the random numbers generated by the reader to discover the fact that the packets are actually sent by the attacker.

- Tracking: When an attacker tries to track a tag by using forged reader communication packets, since a tag includes a random number in the transmitted plain text and the random number will be changed immediately after the transmission of cipher text, the tag will transmit different packets every time such that the attacker cannot track the tag using the forged reader challenge packet.

7. CONCLUSIONS AND DISCUSSIONS

In this paper we successfully implemented a complete authentication system for active RFID systems. A high performance asymmetric key TTS algorithm has been adopted as the core of our encryption technology and subsequently solves the problems faced by traditional asymmetric key algorithm in the performance of key production, signature, and signature verification. Besides we also designed and implemented RFID readers and active RFID tags. In our system, unauthenticated readers can not directly read the cipher text transmitted by RFID tags. The security and privacy of the data can thus be protected. RFID readers transmit data by using multi-hop relaying in which each RFID reader effectively relays the collected tag information to the back-end platform for further analysis.

Both RFID readers and RFID tags in our system fully utilize the power-saving schemes in high-performance, low-power-consumption microprocessor MSP430 for better power performance. The power switch of the RFID reader can let the reader be free from the limitation of power deployment and thus can provide more flexibility to the reader's applications. The power management control of the RFID tag can effectively prolong the duration of operation period to more than one year. All these design considerations make our system more practical to wider application areas.

Currently RFID technologies have been applied to various applications. The privacy and security of data have also become major issues in these RFID applications. Simple hash functions, symmetric or asymmetric encryption algorithms, or even hardware have been used to solve the security problem. To reduce the cost, researchers have hoped to use software to replace hardware. The focus has thus been on the development of security algorithms. In the literature most researches discussed the mechanism to use security algorithms to solve the RFID security issues, yet very few researches implemented the algorithms in a complete RFID system. In this paper, we thus proposed and implemented an RFID system with security authentication mechanisms to achieve both the goals of data security and data collection via long-distance multi-hop relaying. We can also set up our RFID system in various environments. With the introduction of authentication and security technologies, we do not have to worry any more about the stealing of information from illegal readers or from stealers. The multi-hop mechanism can transmit data collected by the remote readers via relaying so that we do not have to set up a computer for each single reader. This can substantially reduce the total system cost. With the deployment of such a RFID system, we can further develop and verify the usefulness of various software security algorithms in a real RFID system.

REFERENCES

1. P. Pedro, C. H. Julio, E. T. Juan, and R. Arturo, "RFID systems: A survey on security threats and proposed solutions," in *Proceedings of the 11th IFIP International Conference on Personal Wireless Communications*, Vol. 4217, 2006, pp. 159-170.
2. I. Damgard and M. Ostergaard, "RFID security: Tradeoffs between security and efficiency," *Cryptology ePrint Archive*, Report 2006/234, 2006.
3. M. Rieback, G. Gaydadjiev, B. Crispo, R. Hofman, and A. Tanenbaum, "A platform for RFID security and privacy administration," *USENIX/SAGE Large Installation System Administration Conference*, 2006, pp. 89-102.
4. A. Juels, "RFID security and privacy: a research survey," *IEEE Journal of Selected Areas in Communications*, Vol. 24, 2006, pp. 381-394.
5. S. Piramuthu, "Lightweight cryptographic authentication in passive RFID-tagged systems," *IEEE Transactions on Systems, Man, and Cybernetics – Part C: Applications and Reviews*, Vol. 38, 2008, pp. 360-376.
6. K. Ouafi and R. C. W. Phan, "Privacy of recent RFID authentication protocols," in *Proceedings of Information Security Practice and Experience*, LNCS 4997, 2008, pp. 263-277.
7. K. Wong, P. Hui, and A. Chan, "Cryptography and authentication on RFID passive tags for apparel products," *Journal Computers in Industry*, Vol. 57, 2006, pp. 342-349.
8. B. Y. Yang, C. M. Cheng, B. R. Chen, and J. M. Chen, "Implementing minimized multivariate PKC on low-resource embedded systems," in *Proceedings of Security in Pervasive Computing*, LNCS 3934, 2006, pp. 73-88.
9. S. Vaudenay, "RFID privacy based on public-key cryptography," in *Proceedings of International Conference on Information Security and Cryptology*, LNCS 4296, 2006, pp. 1-6.
10. L. Batina, J. Guajardo, T. Kerins, N. Mentens, P. Tuyls, and I. Verbauwhede, "Public key cryptography for RFID-tags," in *Proceedings of the 5th IEEE International Conference on Pervasive Computing and Communications*, 2006, pp. 217-212.
11. C. Castelluccia and G. Avoine, "A pretty good key exchange protocol for RFID tags," in *Proceedings of International Conference on Smart Card Research and Advanced Applications*, Vol. 392, 2006, pp. 289-299.
12. K. Nohl and D. Evans, "Quantifying information leakage in tree-based hash protocols," in *Proceedings of the Conference on Information and Communications Security*, Vol. 430, 2006, pp. 228-237.
13. S. Sadanandan and R. Mahalingam, "Light weight cryptography and applications," *Novel Algorithms and Techniques in Telecommunications, Automation and Industrial Electronics*, 2008, pp. 484-488.
14. M. Feldhofer and C. Rechberger, "A case against currently used hash functions in RFID protocols," in *Proceedings of Workshop on RFID Security*, 2006, pp. 109-122.
15. G. Avoine and P. Oechslin, "A scalable and provably secure hash-based RFID protocol," in *Proceedings of the 3rd IEEE International Conference on Pervasive Computing and Communications*, 2005, pp. 110-114.
16. M. Ohkubo, K. Suzuki, and S. Kinoshita, "Efficient hash-chain based RFID privacy protection scheme," in *Proceedings of the International Conference on Ubiquitous*

Computing, 2004.
17. J. Ayoade, "Security implications in RFID and authentication processing framework," *Computers and Security*, Vol. 25, 2006, pp. 207-212.
18. M. Feldhofer, S. Dominikus, and J. Wolkerstorfer, "Strong authentication for RFID systems using the AES algorithm," in *Proceedings of Workshop on Cryptographic Hardware and Embedded Systems*, Vol. 3156, 2004, pp. 257-370.
19. M. Burmester, T. van Le, and B. de Medeiros, "Provably secure ubiquitous systems: Universally composable RFID authentication protocols," in *Proceedings of the 2nd IEEE/CreateNet International Conference on Security and Privacy in Communication Networks*, 2006, pp. 1-9.
20. A. Satoh and K. Takano, "A scalable dual-field elliptic curve cryptographic processor," *IEEE Transactions on Computers*, Vol. 52, 2003, pp. 449-460.
21. H. Fell and W. Diffie, "Analysis of a public key approach based on polynomial substitution," *Advances in Cryptology – CRYPTO*, LNCS 218, 1985, pp. 340-349.
22. T. Moh, "A fast public key system with signature and master key functions," in *Proceedings of the International Workshop on Cryptographic Techniques and Ecommerce*, Vol. 27, 1999, pp. 2207-2222.
23. J. M. Chen and B. Y. Yang, "A more secure and efficacious TTS scheme," in *Proceedings of the 6th International Conference on Information Security and Cryptology*, LNCS 2971, 2004, pp. 320-338.
24. B. Y. Yang, J. M. Chen, and Y. H. Chen, "TTS: High-speed signatures from low-end smartcards," *Lecture Notes in Computer Science*, Vol. 3156, 2004, pp. 371-385.
25. B. Y. Yang and J. M. Chen, "Rank attacks and defence in tame-like multivariate PKC's, http://eprint.iacr.org/2004/061.

Hsi-Wen Wang (王錫文) was born in Taoyuan, Taiwan, in 1978. He received the B.S. degree in Electrical Engineering from Lung-Hwa University o Science and Technology (LHU), Taoyuan, Taiwan, in 2000 and the M.S. degree in Engineering Science from Lung-Hwa University of Science and Technology (LHU), Taoyuan, Taiwan, in 2005. He is currently working towards the Ph.D. degree in Department of Electronic Engineering and Graduate Institute of Computer and Communication Engineering at National Taipei University of Technology (NTUT), Taipei, Taiwan. His research interests include electronic system design for portable applications, digital video broadcasting-handheld (DVB-H), telecare and mobile-care system design, and wireless sensor network for medical care applications.

Ren-Guey Lee (李仁貴) received the M.S. degree from Department of Electrical Engineering, National Chen Kung University (NCKU), Tainan, Taiwan, in 1989, and the Ph.D. degree from Department of Electrical Engineering, National Taiwan University (NTU), Taipei, Taiwan, in 2000. Since 2002, he has been with the Department of Electronic Engineering and Graduate Institute of Computer and Communication Engineering, National Taipei University of Technology (NTUT), Taipei, Taiwan, where he is currently a Professor. His research interests include electronic system design for portable applications, medical informatics, telecare and mobile care system design, and wireless sensor network for biomedical applications.

Chun-Chieh Hsiao (蕭俊杰) received the M.S. degree in electrical and computer engineering from the State University of New York, Buffalo, in 1990. He is currently a Ph.D. candidate in the Department of Electrical Engineering, National Taiwan University (NTU), Taipei, Taiwan, R.O.C. During 1991-1993, he was with the Computer and Communication Laboratory (CCL), Industrial Technology Research Institute (ITRI) of Taiwan, where he focused on research and development of advanced computer graphics systems. From 1993 to 2003, he was with the Department of Electrical Engineering, Lunghwa University of Science and Technology (LHU), Taoyuan, Taiwan, R.O.C. Since 2003, he has been with the Department of Computer Information and Network Engineering, LHU. His current research interests include telecare and mobile care systems, wireless sensor networks for biomedical applications, and advanced computer systems.

Guan-Yu Hsieh (謝冠宇) was born in 1983. He received the B.S. degree in the Graduate Institute of Computer and Communication Engineering, National Taipei University of Technology, in 2007. His research interests in security for biomedical and RFID applications.