IEICE TRANS. INF. & SYST., VOL.E93–D, NO.7 JULY 2010

PAPER

Efficient Method of Achieving Agreements between Individuals and Organizations about RFID Privacy

Shi-Cho CHA[†a)], ***Member***

SUMMARY This work presents novel technical and legal approaches that address privacy concerns for personal data in RFID systems. In recent years, to minimize the conflict between convenience and the privacy risk of RFID systems, organizations have been requested to disclose their policies regarding RFID activities, obtain customer consent, and adopt appropriate mechanisms to enforce these policies. However, current research on RFID typically focuses on enforcement mechanisms to protect personal data stored in RFID tags and prevent organizations from tracking user activity through information emitted by specific RFID tags. A missing piece is how organizations can obtain customers' consent efficiently and flexibly. This study recommends that organizations obtain licenses automatically or semi-automatically before collecting personal data via RFID technologies rather than deal with written consents. Such digitalized and standard licenses can be checked automatically to ensure that collection and use of personal data is based on user consent. While individuals can easily control who has licenses and license content, the proposed framework provides an efficient and flexible way to overcome the deficiencies in current privacy protection technologies for RFID systems.

key words: *RFID privacy, privacy enhancing technology, RFID*

1. Introduction

Radio Frequency Identification (RFID) technology has been extensively adopted to augment different applications in recent years. As information is transferred via radio waves, objects can be recognized at a distance without optical or visual contact. However, because radio waves are invisible, RFID technologies can threaten privacy [1]–[3]. For example, via remote keyless systems, people can drive their cars without using their keys. However, the people's locations can be determined by tracking the keys.

Therefore, several groups, such as Consumers Against Supermarket Privacy Invasion and Numbering (CASPIAN), Electronic Privacy Information Center (EPIC), and the American Civil Liberties Union (ACLU) have raised concerns about user privacy and requested that organizations withdraw the RFID applications that can invade consumer privacy. For instance, Gillette was requested to terminate its pilot project to photograph customers as they remove razor blades from a shelf in Tesco [4].

To provide basic rules for organizations deploying RFID systems, several customer protection groups issued the following Principles of Fair Information Practice for RFID technology based on the OECD Guidelines for Protection and Privacy of Transborder Flows of Personal Data in 2003 [5]:

- Openness. The development of RFID systems, practices and policies governing the use of RFID systems must be open.
- Purpose specification. The purposes for which RFID tags and readers are used must be specified and disclosed.
- Collection limitation. The subsequent use of collected data should fulfill the purposes disclosed during data collection.
- Security safeguards. Personal data should be protected by reasonable security safeguards from modification, disclosure, and unauthorized access.
- Accountability. Organizations using RFID systems should be accountable to individuals and comply with the principles outlined here.

In simple terms, these principles require organizations to disclose their policies regarding RFID activities and obtain customer consent before collecting and using customer data via RFID technologies. Additionally, organizations must adopt appropriate mechanisms to enforce these policies. However, current RFID technologies that enhance privacy typically focus on how to help users hide their personal information. For example, several technologies have been developed to protect personal data stored in RFID tags [6]–[10] or to prevent organizations from tracking a person through information emitted by RFID tags [10]–[14]. Although these technologies can be used to ensure that only organizations that have obtained a person's consent collect personal data, an efficient way for customers to achieve agreements with organizations is lacking. For instance, among different types of consent, such as oral/verbal, written, proxy, passive,and so forth, written consent provides the strongest evidence, especially when disputes occur. While organizations are usually requested to obtain "written consent" from individuals before using their data, it is very inefficient and inconvenient for customers and organizations to proceed with written consent to use RFID systems. Furthermore, people may favor flexible control of an organization's RFID activities. For example, shoppers may only allow a mall store to track RFID tags they brought when they are shopping in the store. Nearby stores in the same mall cannot track this person unless the person walks into those stores.

To provide an efficient and flexible method of achiev-

Manuscript received July 16, 2009.
Manuscript revised December 16, 2009.

[†]The author is with the Information Management Department, National Taiwan University of Science and Technology, Taipei, 106, Taiwan (R.O.C).
a) E-mail: csc@cs.ntust.edu.tw

DOI: 10.1587/transinf.E93.D.1866

ing agreements between individuals and organizations about RFID privacy, this study proposes to protect privacy in RFID systems with digitalized licenses. In the proposed framework, organizations must obtain licenses before legally collecting personal data via RFID systems. The licenses can be viewed as a type of proxy consent. As current countries usually have electronic signature acts, the digital licenses, which are signed with digital signature, can provide the same power of evidence as written consents. Moreover, licenses can be checked automatically to ensure that collection and use of personal data is based on a person's consent. The proposed method allows individuals to easily control who has a license to collect and use their data and license content, and is an efficient and flexible method that overcomes deficiencies in current privacy protection technologies for RFID systems.

The remainder of this paper is organized as follows. Section 2 introduces preliminary knowledge and related work. Section 3 discusses the design requirements. Section 4 is an overview of the proposed framework. Section 5 illustrates how the framework applies in different scenarios. Section 6 discusses the main components of the proposed framework. Section 7 discusses compliance, security, and scalability issues of the proposed framework. Finally, Sect. 8 draws conclusions along with recommendations for future research.

2. Related Work

2.1 Platform for Privacy Preferences (P3P)

Accompanied by the hope of industry and individuals, the first formal specifications of Platform for Privacy Preferences (P3P) was proposed by the World Wide Web Consortium (W3C) in April 2002 [15] for privacy protection on the Web. The original concept can be described as follows: P3P defines a vocabulary and specification for a Web site to declare its privacy practices. The privacy practices are represented as machine readable "proposals" to describe what personal data will be collected by the site, for what purposes, other recipients of the data, and the destruction timetable. When a user requests a Web page (with which the user has not yet entered into a privacy agreement) from the site, a set of proposals is sent to the user. The user's agent can then choose one proposal that matches the user's preferences, and sends an agreement ID back to the site to express acceptance of the proposal. After receiving the agreement, the site will transfer the requested page to the user. If none of the proposals is accepted by the user, the site can send another set of proposals for further negotiation.

In the above process, the Web site may also request the user's data. This feature originated in the Open Profiling Standard (OPS) [16]. OPS was intended to provide privacy protection for personal profile information exchange over the Web, and was folded into the early P3P. If the user accepts a proposal, the requested data along with the agreement ID are transmitted to the site (in the HTTP request header [17]). The automatic transfer of personal data raises some controversies, however. So the P3P Specification Working Group later decided to remove this function [18]. The negotiation module was also simplified due to the complexity of the original process. Subsequent to these two modifications, officials then established the prototype of the current P3P.

Generally speaking, P3P provides a standard protocol for a Web site to express its privacy practices, and for a user's browsing agent (e.g., Internet Explorer) to determine whether or not the practices match the user's privacy preferences. P3P, however, lacks a mechanism for users or third-party organizations to verify if web sites have faithfully executed the practices [19], and for applications to check if the use of personal data collected by the sites has indeed been authorized by individuals and are used in a way that is in accordance with what the individuals have agreed when releasing their personal data. Therefore, the authors of this study previously proposed the Online Personal Data Licensing (OPDL) framework [20], [21] to overcome deficiencies of P3P by concretizing the agreements between individuals and Web sites into licenses, which can then be used to legally resolve privacy disputes between individuals and web sites, and can also be used by applications to prevent abuse of personal information.

Because P3P and OPDL frameworks focus on the collection and processing of personal data on the Internet, this current study enhances the framework for RFID systems.

2.2 RFID Privacy Enhancing Technologies

Several studies, such as [1], [22]–[24], have reviewed issues related to RFID privacy and solutions. Generally, solutions can be classified as follows: (1) those that detect unauthorized RFID activity; (2) those that protect personal data; and, (3) those that prevent a person from being tracked unknowingly.

First, individuals or customers protection groups need a tool that detects whether an organization is secretly using an RFID system. For example, an RFID Guardian checks RFID scans to find unknown tags in the vicinity [25]. In general, this approach works well when organizations use standard RFID systems; however, its effectiveness may be limited when non-standard or proprietary technologies are used.

Second, personal data in RFID systems can be protected as follows:

- Cryptography. Data in a tag can be encrypted such that only authorized parties can obtain the data [22]. Because of the very tightly constrained environments in which RFID tags are used, several researchers have proposed lightweight versions of the symmetric key [6] and public key [26] cryptography schemes.
- Reader authentication. Tags can only output their IDs to specified readers. For example, a tag can output its ID only when it receives a fixed or dynamically gener-

ated key sent by a reader [10]. The ID can further be hashed such that only an authorized reader can obtain the original ID of the tag by looking up a table or a database containing (tag ID, hashed value) tuples [7].

- Privacy labels. Privacy preference information can be embedded into tags such that readers and back-end information systems can deal with related information based on this information. For instance, Kim et al. [9] proposed a scheme that has five different privacy levels that can be embedded in a tag. Consequently, information systems that contain personal data can determine the amount of associated information to be provided based on the privacy level in a tag. Similarly, Juels and Brainard developed the soft-blocking approach [8]. In the soft-blocking approach, a tag can be classified as a "blocker," "public," or "private" based on its ID. Readers must implement tag privacy agents (TaPAs) to filter out IDs of private tags when blocker tags are present.

The primary vulnerability of these approaches is associated with eavesdropping. First, in reader-authentication schemes, tags do not respond to unauthorized readers. Thus, adversaries may eavesdrop on communication between tags and authorized readers. Furthermore, although adversaries cannot obtain information related to a tag, they can still trace the appearance of a tag based on its ID regardless of whether the ID is encrypted.

Therefore, supplementary technologies can be employed to prevent identification of the location of a specified tag by unauthorized parties. Customers can deactivate or discard RFID tags such that their location cannot be tracked through tags. For example, customers can remove attached tags when products are purchased. Current conventional tags usually support a kill function that permanently deactivates tags automatically via kill commands, including associated passwords [27]. If customers want tags to remain operative while in their possession, they can use technologies such as blocker tags [12] to deactivate tags temporarily.

Instead of rendering tags completely silent, customers may only want to prevent unauthorized readers from obtaining information from their tags. Thus, several researchers have proposed solutions that render tag responses indistinguishable from random numbers for unauthorized readers or adversaries [28], [29]:

- Hash-chain: Tags can change their identities based on a hash function. For instance, [14] proposed a method that allows a tag to modify its IDs through a hash function (along with a specified key) each time it is queried by readers. If hash function H is used, a tag will renew its ID from ID_i to $ID_{i+1} = H(ID_i)$ upon request. Therefore, authorized parties that know H can collect a chain of IDs and identify which tag can refresh its ID as the chain. Obviously, the scalability of this approach is problematic because we may need to compare IDs in each chain to identify a tag. Several studies have addressed this problem [28], [30]. These studies are not discussed in detail here.
- Key search. A tag with identity ID_i can share its secret key K_i with authorized readers. A tag may generate a random nonce R and emit $(H(k_i||R), R)$ with a hash function H each time it receives a request [10]. A reader can then compute $H(k||R)$ for all the keys it has until it identifies k_i for tag identification. Furthermore, multiple keys can be used simultaneously to improve security and search efficiency [31], [32].
- Minimalist cryptography. Instead of incorporating a hash function into a tag, each tag can contain a set of pseudonyms and release different pseudonyms rotationally in response to each reader query [13], [33]. Readers without knowledge of tag pseudonyms cannot correlate different appearances of the tag.
- Re-encryption. An identity ID_o can be encrypted with a random nonce R_j into a ciphertext $E_k(ID_o, R_j)$ for use as a temporary tag identity. The original identity ID_o is used to re-encrypt the temporary identity with a new random nonce R_k periodically to generate a new tag ID $E_k(ID_o, R_k)$. Consequently, only parties that have a particular key can obtain the original identity ID_o. For example, a law enforcement agency may use a public and private key pair to encrypt banknotes [34]. The RFID tag embedded in a banknote stores a ciphertext, C, based on the serial number S of the banknote along with a random nonce R_i. Additionally, authorized readers in shops or banks use the public key from the law enforcement agency to encrypt S and another random nonce R_j into a new ciphertext C'. Consequently, only the law enforcement agency can obtain S using its private key. However, this approach has a limitation in that each reader must know the public key for re-encryption. Thus, [11] used ElGamal cryptosystem to allow readers to re-encrypt a ciphertext without knowledge of the corresponding public key.

Generally, these approaches are limited in determining whether a reader can obtain information from a tag dynamically. For instance, a person may only allow an organization to identify the person's tag on a weekday during the daytime. Key management and access control mechanisms may be necessary in this case. However, the cost would be unacceptable if the mechanisms were implemented in RFID tags. To address the issue without increasing the costs of RFID tags, several studies have proposed using external devices to enhance privacy of RFID applications based on tags that may not have authentication and access control functions. For example, a person may bring a mobile device with an RFID Enhancer Proxy (REP) [35] to inactivate specified nearby RFID tags. The REP can then simulate the tags to nearby RFID readers. While mobile devices typically have higher computational power than RFID tags, REPs can enforce sophisticated privacy protection policies, e.g. deciding whether a tag can be identified based on time and location. Another mechanism called RFID Guardians [25] can authenticate nearby RFID readers and exchange keys to obtain information of selected tags. Then, RFID Guardians

can protect the tags from unauthorized access based on access control policies set by the person.

REPs and RFID Guardians do not satisfy the Principles of Fair Information Practice for RFID technology because REPs and RFID Guardians only provide mechanisms for a person to set access control policies to determine whether an RFID reader of an organization can obtain data from a tag. The person cannot know how the organization uses the data. From this perspective, the proposed licensing mechanism complements current RFID technologies and enhances privacy by allowing users to achieve agreements with organizations about RFID privacy policies to comply with current RFID privacy protection regulations and guidelines.

3. Design Requirements

Suppose a person (U_x) has a set of RFID tags ($\{T_i\}$). Each tag (T_i) has a unique original tag ID (RID_{T_i})[†]. On the other hand, an organization (O_y) may build up its RFID systems to track tags of the person and obtain data stored in the tags. The organization has privacy policies ($\{P_{O_y,j}\}$) about its RFID activities based on P3P for different people or situations. The above scenario suggests the following main requirements:

- Compliance. The proposed approach should provide an efficient way for an organization to obey the Principles of Fair Information Practice for RFID technology.
- Security. The proposed approach should consider the following common threats:
  - Masquerade. A user records and subsequently replays data emitted from another user's tag pretending that he or she is the other user.
  - Manipulation. An unauthorized user should not be able to modify tag data.
  - Data Interception. An unauthorized third user should not be able to observe the other user's data during communication between the other user's tag and an organization.
- Scalability. The proposed approach should be scalable in terms of participating tag owners and RFID systems.

4. System Overview

The proposed framework allows people to issue licenses for readers to collect or use personal data from specified tags. The components of the proposed framework are described as follows (Fig. 1).

First, organizations can utilize *Back-End Servers* to collect personal data from RFID tags. The *Reader Controller* collects data from nearby tags. The *ID Translator* then tries to restore the data to its original state based on the licenses managed by the *License Manager*. If the ID Translator cannot recognize the data, it asks the Reader Controller to obtain a corresponding license. After receiving this message, the Reader Controller asks the *Proposal Generator* to translate its organization's policies into a licensing proposal and sends this proposal to the *Licensing Service*.

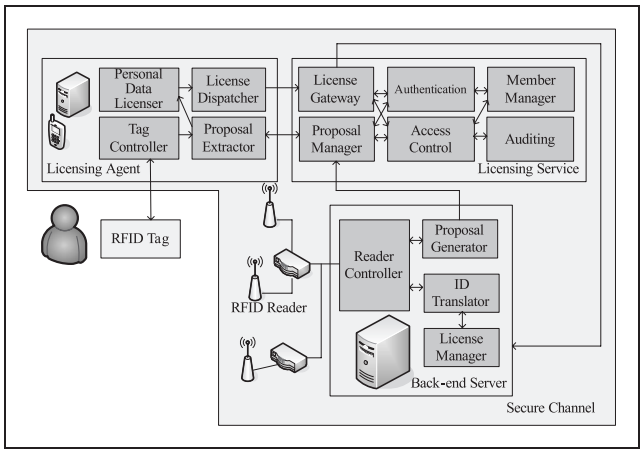

Fig. 1 Architecture of our proposed framework.

A licensing service can be operated by a trustworthy third party or a legal agency. Generally, a Licensing Service can use two operational models when dealing with requests from Back-End Servers. In the *direct negotiation model*, the Proposal Manager in the Licensing Service forwards requests to associated *Licensing Agents*.

The kernel of a Licensing Agent is the *Personal Data Licenser*. A user's Personal Data Licenser decides whether licensing proposals can be allowed based on the user's privacy preferences and issues licenses on behalf of that user. The detailed components of a Personal Data Licenser are outside the scope of this work-those interested can refer to [20], [21]. The direct negotiation model is so called because a user's Personal Data Licenser deals with requests for personal data directly. If a license proposal is allowed, a license is generated. The *License Dispatcher* then sends the issued license to the Licensing Service.

Upon receiving a license, the *License Gateway* in the Licensing Service forwards the license to the corresponding Back-End Server. Consequently, the Back-End Server can access associated data based on the license.

Instead of sending licensing proposals to the Licensing Agents of users, the Licensing Service retains the proposals and gives each proposal an identity for further identification in the *indirect processing* model. When a Back-End Server cannot identify original tag data, the *Reader Controller* in the Back-End Server writes the identity of its licensing proposal to the tag. The *Tag Controller* in user Licensing Agent monitors tags periodically. When the Tag Controller finds a new identity of licensing proposal in a tag, the *Proposal Extractor* requests a licensing proposal from the Licensing Service based on the proposal identity. The Personal Data Licenser then deals with the proposal in the similar manner in the direct negotiation model.

To prevent a tag from being tracked, the Tag Controller

[†]In addition to identity information, a tag may store other data. This article leaves that situation to our future work.

updates the identities of tags or tag data periodically. Its implementation is not compelled to any form; for example, it can be implemented as a software agent in a person's mobile phone or a component embedded in an RFID tag when the tag have sufficient computing power.

Finally, security mechanisms must be employed to protect personal data. Each user registers for membership with the licensing service. The *Member Manager* maintains personal data about the user for authentication and communication along with information about tags carried by the user. At the same time, an organization that wishes to identify tags of users and collect information from the tags should have an account for the licensing service. The Member Manager also manages the authentication information of the organization. Secure channels, such as SSL, are established to protect communications between Back-End Servers and the Licensing Service and between Licensing Agents of users and the Licensing Service. The *Authentication* mechanism can authenticate member users and member organizations based on the information provided by the Member Manager. For example, if SSL is used for secure communication, X.509 certificates can be exchanged enabling both parties to authenticate each other. The *Access Control* mechanism checks if requesters have the authorization to perform the requested operations. The *Auditng* mechanism logs the operation requests for future tracking.

5. Example Scenarios

This section presents scenarios for the two operational models.

5.1 An Example Scenario for the Indirect Processing Model

In the indirect processing model, people must use a mobile device, such as cellular phones, and PDAs, to monitor their tags. A typical scenario for the model has the following steps (Fig. 2).

- Step 1: When an RFID reader of an organization detects a new tag, it writes the identity of the organization's licensing proposal to the tag.
- Step 2: The Licensing Agent discovers the event and obtains the tag identity.
- Step 3: The Licensing Agent uses the identity to request the associated licensing proposal from the Licensing Service.
- Step 4: If the proposal matches user preferences, a license (along with temporary keys to obtain the tag ID) is generated; otherwise, the proposal may be rejected directly. When necessary, a notification form is generated asking the user to decide whether to accept the proposal.
- Step 5: The Licensing Agent sends the license to the Licensing Service over a secure channel.
- Step 6: After receiving the license, the Licensing Service authenticates and forwards the license to the organization's Back-End Server with secure communication technologies. The Back-End Server then knows how to restore the tag ID from a series of pseudonyms.

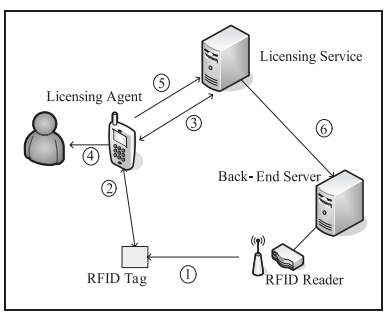

Fig. 2 A typical scenario for the indirect processing model.

5.2 An Example Scenario for the Direct Negotiation Model

Figure 3 presents a representative scenario of the direct negotiation model.

- Step 0: Before a tag can be identified, it must be initialized by its owner's Licensing Agent. Therefore, the tag knows which identity it should have at a specified time.
- Step 1: When the tag is detected by an RFID reader, the Back-End Server of the reader checks whether the tag can be identified.
- Step 2: If the server cannot identify the tag, it sends a request to the Licensing Service.
- Step 3: After authenticating the server, the Licensing Service finds the associated Licensing Agent and forwards the licensing proposal regarding the server to the agent securely. For example, the Licensing Service may send a GSM short message to the Licensing Agent. The Licensing Agent then obtains the proposal through SSL.
- Step 4: Similar to step 4 in the above scenario of the indirect processing model, the Licensing Agent decides whether to accept the proposals.
- Step 5: If a proposal is accepted, a license is generated

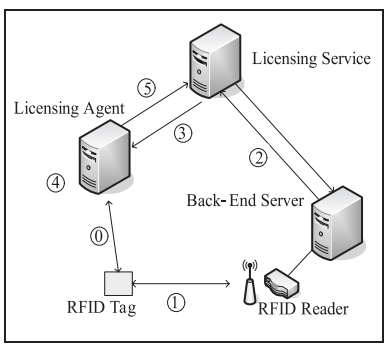

Fig. 3 A typical scenario for the direct negotiating model.

and transferred to the Licensing Service; otherwise, the Licensing Agent sends a reject message back to the Licensing Service.
- Step 6: The Licensing Service forwards the license to the Back-End Server allowing the server to identify the tag. Note that the license is transferred over a secure channel in the whole process to protect the license.

6. Main Components

This section describes the principal components of the proposed framework. The following sub-sections discuss in detail licensing proposals and licenses, the licensing service, and tag identification.

6.1 Licensing Proposals and Licenses

Licensing proposals are based on P3P privacy policies [21], [36]. Typically, a privacy policy has the following definition:

***Definition* 1** (Privacy Policy): A privacy policy $P_{O_y,j}$ of an organization O_y is represented by a 4-ary tuple: (I,SP,DI, $\{ST_k\}$). I is basic information about the privacy policy, such as the name of the organization and the URL of another version of a human readable proposal. SP is the URL of the organization's security policy†. DI shows information about how to solve disputes between the organizations and people who accept the proposal. $\{ST_k\}$ is a set of statements about the privacy practices of the organization. Note that the proposal is signed by the organization to thwart masquerading.

***Definition* 2** (Privacy Statement): A privacy statement ST_k is represented by a set of 4-ary tuple: $\{(\{D_m\},\{PU_l\},CO,RT)\}$. $\{D_m\}$ describes what data are going to be collected or used. $\{PU_l\}$ represents a set of purposes for using the data. RT indicates the retention policies of the organization for collected data. CO provides a further explanation about the statement.

Figure 4 shows an example of a P3P-based licensing proposal. In this proposal, a supermarket (examplemart) wants to collect and use personal data from its customers. The supermarket has another version of a human-readable proposal disclosed at http://examplemart/humanreadableproposal.html. The proposal is signed by the supermarket to thwart masquerading. In this example, the URLs of the supermarket's security policy, risk assessments, and controls for risks are offered in the SECURITY-POLICY element so customers can evaluate the supermarket's data security. In the DISPUTES element, the supermarket states that individuals can send their complaints to an independent organization, "certification.example.org," by setting the resolution-type attribute value to "independent" and the service attribute to "certification.example.org". Individuals can verify the truth of the dispute solution by setting requests to the URL stated in the verification attribute. Moreover, the REMEDIES element

```
<POLICY name="example proposal"
 discuri="http://exampleshop/humanreadableproposal.html"
sigalgorithm="DSA" signature=….. date=…… ID=……>
<SECURITY-POLICY discuri="http://exampleshop/securitypolicy.html"
 risksriskmanage="http://exampleshop/riskmanage.html" />
<DISPUTES-GROUP>
 <DISPUTES resolution-type="independent"
  service="http://www.certification.example.org"
  verification="http://www.certification.example.org/verify?
  proposalid=…… short-description="certification.example.org">
  <REMEDIES><correct/></REMEDIES>
</DISPUTES></DISPUTES-GROUP>
<STATEMENT><CONSEQUENCE> .....
 </CONSEQUENCE>
 <PURPOSE><develop/><tailoring/><individual-analysis/>
  <individual-decision></PURPOSE>
 <DATA-GROUP><DATA ref="#user.tagid"/> </DATA-GROUP >
<RETENTION>2y</RETENTION>
</STATEMENT>
</POLICY>
```

Fig. 4 An example of a licensing proposal.

indicates that if a policy breach occurs, the supermarket has implemented a policy to rectify the error with the ⟨correct/⟩ element.

The other information in the example (Fig. 4) outlines the supermarket's practices regarding personal data. The practices are put in STATEMENT elements. A statement element describes what data need to be collected or used in a DATA-GROUP element. Each DATA element represents a requested data item and uses a "ref" attribute to specify the name of the data item. This work uses "#user.tagid" to represent data for tag ID. The supermarket can describe its purposes to collect or use a person's data in the PURPOSE element. P3P classify purposes into 12 different classes. In this example, the supermarket traces the identities of RFID tags to determine a person's habits, interests, or other personal characteristics for the purposes of research, analysis, reporting, generating recommendations, and providing tailored services by describing purposes with ⟨develop /⟩, ⟨tailoring /⟩, ⟨individual-analysis /⟩, and ⟨individual-decision /⟩ tags in the PURPOSE element. The mart has a retention policy of retaining collected data for 2 years at most and expresses the policy in the RETENTION element.

After receiving the licensing proposal, the Personal Data Licenser of a person's Licensing Agent informs the person to determine whether to accept the proposal. Note that a person may set rules based on APPEL [37], which is designed for users to express their privacy preferences based on privacy policies from P3P enabled Web sites, to enable the person's Licensing Agent to reject or accept a proposal directly. The details of APPEL rules and associated proposal processing processes are not discussed here.

After receiving a user's confirmation, the Personal Data Licenser generates a license and sends it to the requester.

†Recall that the security safeguard principle requires that organizations adopt reasonable security controls to protect the collected personal data. Generally, P3P does not support this principle; thus, we extend P3P by requesting an organization to disclose its security policies.

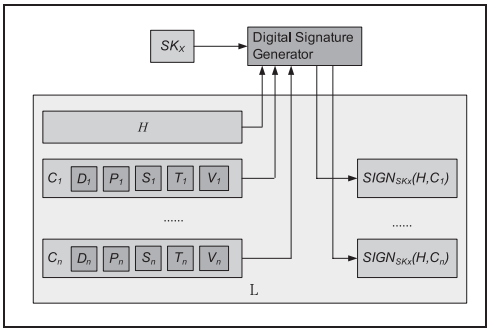

Fig. 5 A license contains a header, a set of clauses, and signatures for clauses.

```
<DATA-GROUP><DATA ref="#user.tagid"/><RID>.....</RID >
<KEY from="2009/7/1 14:00:00"validate="300">
.....</KEY>
<KEY from="2009/7/1 14:05:00"validate="300">
.....</KEY>
</DATA-GROUP>
```

Fig. 6 Part of an example license.

This work adopts the license format defined in OPDL [20], [21]. Figure 5 shows a logic-based view of a license. Suppose a license is issued by person U_X (with private key SK_{U_X}) to service provider O_Y. The main components of the license include a header H and a number of clauses $C_1 \ldots\ldots C_n$. The header contains general information about the license (e.g., the licenser, licensee, the time the license was issued, and security level claimed by the licensee). Each clause C_i contains the allowed privacy practices/purposes P_i for using data, with whom (S_i) information may be shared, and the time T_i at which the data must be destroyed (or a validation period for the clause) for a set D_i of data items. Additionally, a person may also assign a set of values, V_i to D_i, or keys, $K_{i,p}$, to access D_i in period p[†].

Figure 6 shows part of an example license. If on 2009/7/1, a person allows an organization to trace one of his tags from 14:00 to 14:10, information to identify the tag is put in a DATA element and embedded in a license. This article shows the details about how to identify a tag based on associate licenses in Sect. 6.2. Note that the current framework uses key search schemes. Therefore, a DATA element has an RID element that represents the original ID of the associated tag and a series of keys for identifying the tag during specified time periods (with KEY elements and "from" and "valid" attributes for the elements). Extensibility to other schemes, such as minimalist cryptography and hash-chain, is left to future work.

6.2 Tag Identification

This work adopts the key search approach to ensure that only licensed parties can identify RFID tags brought by a user. Each tag has a root key that produces the tentative tag ID. Suppose the original ID of a tag T_i is RID_{T_i} (Fig. 7). The root key of the tag is then RK_{T_i}. In a specified time period

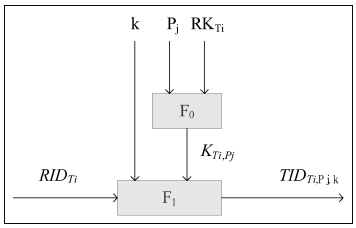

Fig. 7 Generation of a tentative ID of tag Ti in the k-th time slice of P_j.

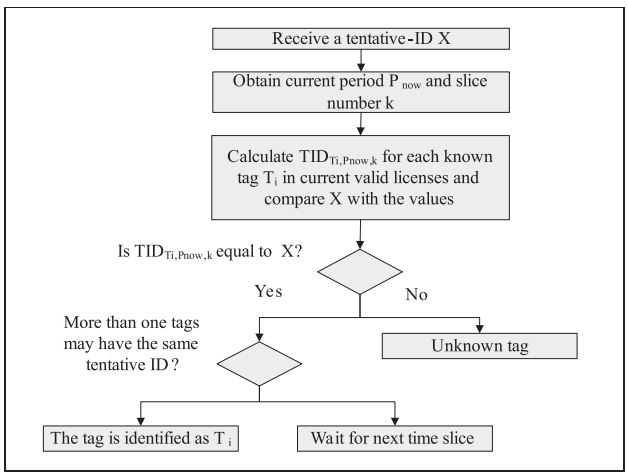

Fig. 8 Flowchart for tag identification in Back-End Servers.

P_j, the Licensing Agent or the tag itself generates a temporary key K_{T_i,P_j} with a function F_0 based on the root key and the start time of the period. A time period can further be divided into n slices. In the k-th time slice of P_j , the Licensing Agent or the tag uses the original ID of the tag RID_{T_i}, the temporary key K_{T_i,P_j}, and the slice number k to generates a tentative ID $TID_{T_i,P_j,k}$ through a function F_1. Tag Controller of T_i then updates the tentative identity of T_i to the value.

Figure 8 shows how a Back-End Server identifies a tag. When a Back-End Server receives a tentative identity X emitted from T_i, it first obtains current period number P_{now} and slice number k. Suppose that the Back-End Server has a set of valid tags T. For each tag T_i in T, the Back-End Server knows the key $K_{T_i,P_{now}}$ needed to identify the tag at P_{now}. The Back-End Server uses the information to calculate tentative ID $TID_{T_i,P_{now},k}$ for each tag. The Back-End Server then compares X with the value.

If no tentative ID equals X, the Back-End Server requests licenses as mentioned; otherwise, the Back-End Server checks whether two tags have the same tentative ID at that moment. If two different tags generate the same tentative ID, it is called an ID collision situation. The size of an ID (or RID) indicates an ID collision situation in the following experiment. As Table 1 illustrates, this work chooses

[†]To prove that a license is issued by a person and that clauses are unaltered, each C_i together with the license header H is signed with the licenser's private key. This design can verify clauses separately. The details are not discussed here.

Table 1 ID collision rate by ID size.

size (bits)	24	32	40	48	96
Avg. Collision Number	59466	240	1	0	0
Avg. Collision Rate	5.95%	0.02%	0.00%	0.00%	0.00%

five ID sizes. For each ID size, 1 million distinct IDs and associated 5-byte temporary keys are generated randomly. For each ID, ten slice numbers (from 0 to 9) are chosen. This method can then generate ten tentative IDs by using RC4 to encrypt the ID with keys composed of the slice numbers and the ID's temporary key. Table 1 shows that the average ID collision rate drops dramatically as the size of an ID increases. While current RFID standards, such as EPC Global C1 G2, usually use a 96 bit ID size, the collision situation could be ignored.

Assume that the Licensing Service can find out when a tag has the same tentative ID as other tags. The Back-End Server can find out if more than one tag has a tentative ID equal to X from the Licensing Service. If there is only one tag T_j having tentative ID X, the Back-End Server identifies the original tag ID as RID_{T_j}. Otherwise, the Back-End Server waits to identify the tag during the next time slice. Intuitively, the proposed approach ensures that only licensed Back-End Servers can use the keys stored in licenses to identify specific RFID tags. Additionally, a Back-End Server cannot trace a tag when the tag's Licensing Agent does not issue licenses to the Back-End Server.

6.3 Licensing Service

The Licensing Service plays the role of a bridge and trusted third party between Licensing Agents and Back-End Servers. First, the Licensing Service provides the following interface for exchanging proposals in the indirect processing model.

- registerProposal(P,I_O): An organization O can register its licensing proposal P with the Licensing Service. Information of the organization I_O is also sent. If a person accepts the proposal, the associated license can be sent to the organization based on I_O. If successful, an identity for the proposal is returned.
- withdrawProposal(P): To withdraw a licensing proposal.
- queryProposal(P_{ID}): A Licensing Agent requests that the LicensingService obtain a licensing proposal based on its identity P_{ID}.

In the direct negotiation model, tag owners register their Licensing Agents with the Licensing Service to enable the Licensing Service to forward requests to tag owners. An organization sends requests to the LicensingService to obtain licenses with the following interface:

- requestLicense(TID,P,p): An organization requests the LicensingService to forward its licensing proposal P to the tag owner with tentative ID TID in time period p.

On the other hand, a Licensing Agent can send a license or an update request to a Back-End Server through the Licensing Service. This request may be one of the following types.

- acceptProposal(P,L): After a person accepts a licensing proposal P in the indirect processing model, the person uses the interface to transmit a license L for the proposal to the Licensing Service.
- declineProposal(P): A person rejects a licensing proposal P.
- withdrawLicense (L): This is used to withdraw a license.

This paper omits a detailed description of the P3P vocabulary for updates requests since the vocabulary is the same vocabulary used for licensing proposals described in Sect. 6.1. Finally, the Licensing Service keeps logs of its services for solving disputes between individuals and data collectors regarding the genuineness of update requests.

7. Evaluation

7.1 Compliance

This section demonstrates how the proposed framework complies with the Principles of Fair Information Practices for RFID technology. First, the proposed framework requires organizations to obtain licenses from individuals before collecting and using their data. By requiring organizations to outline their purposes and information related to RFID activities, and by allowing individuals to determine whether to accept a proposal, the proposed framework clearly conforms to openness and purpose specification principles.

Second, security safeguard principles can be satisfied by requiring organizations to state their security policies, risk assessment and controls against risks in their proposals. This allows individuals to be aware of the security controls used to protect collected data.

Finally, the accountability and collection limitation principles can be satisfied in the following two ways. First, a tag owner can decide not to issue a new license to an organization when the organization does not obey its disclosed policies. Because the "old" information of a tag's location becomes useless when the tag moves to a new location, licensing mechanisms can ensure that abusive organizations cannot obtain the "most recent" tag information. Alternatively, organizations can be verified and audited by a third party certification organization to prevent data abuses. An automatic procedure can also be created that checks licenses for personal data stored by an organization. The proposed procedure can be integrated into an internal audit system that ensures enforcement of disclosed policies of the organization.

7.2 Security Evaluation

This section discusses how the proposed approach addresses several potential attacks.

7.2.1 Data Interception

A malicious person may try to intercept communication between Licensing Agents and Licensing Services or between Licensing Services and Back-End Servers to obtain keys embedded in licenses. The person could then track associated tags based on the keys. To prevent the attack, secure channels can be established to protect communications. For example, Back-End Servers and Licensing Agents can build SSL communications with Licensing Services.

At the same time, as described before, the proposed approach now uses a key search scheme to generate tentative IDs of tags. A malicious person cannot obtain the raw ID of a tag without the associated keys.

7.2.2 Manipulation

The proposed approach addresses the manipulation threat in the following two ways:

First, as described in Sect. 6.1, the scheme uses digital signature technologies. Therefore, if a malicious person modifies a proposal or a license, people who receive the proposal can discover that the proposal or the license has been modified based on the associated signatures .

Second, current RFID tags usually have access control functions to prevent unauthorized people from modifying data (including identities) stored in the tags. For example, if we implemented the proposed approach with standard EPC Global C1 G2 tags, we can decide that only the controller of a person's Licensing Agent can update the ID of the person's tag based on the standard "access" and "lock" commands.

7.2.3 Masquerading

The proposed approach requests a tag to change its tentative ID every time slice. Therefore, even if a malicious person obtains the tentative ID of a tag, that tentative ID becomes useless after one time slice. If the method uses a small time slice, the probability of masquerading becomes low. However, a malicious person may still record a tag's tentative ID in a time slice and replay the tentative ID for masquerading during that time slice. Suppose that there is a set of Back-End Servers $S = \{s_i\}$ receiving the emitted tentative ID of a tag in a time slice. A malicious person that records the tentative ID may replay the ID to a Back-End Server $s_j \in S$ or a Back-End Server $s_j \notin S$.

In the former case, the Back-End Server that receives the replayed ID can recognize that it has received the same ID more than once in the same time slice. In some applications, the Back-End Server does not need to do anything. For example, the Back-End Server may just need to trace the appearance of a tag. In other security-sensitive applications, the owner of the Back-End Server may hire security guards, use video surveillance systems, or adopt other appropriate countermeasures to deal with the situation.

The current proposed approach can reduce the probability of the occurrence of the latter case by decreasing the size of time slices. In addition, both wireless and wired network security mechanisms can be used to prevent recorded tentative IDs from being transmitted to another place and adopt intrusion detection systems to avoid tentative IDs from being recorded by malicious people. Furthermore, if the proposed scheme modifies the communication protocol between readers and tags, a Back-End Server may be requested to generate a random number and send it to a tag when the server intends to obtain information about the tag. Upon receiving the request, the tag generates tentative IDs based on the approach in this article and emits the result of the tentative ID XOR the random number into the air. Therefore, the threat of masquerading can further be reduced. This study leaves the modification issue to future work.

7.3 Scalability Evaluation

This section discusses the scalability issue in tag identification. As described in Sect. 6.2, when a Back-End Server received an ID emitted from a tag in a time slice, the Back-End Server's tag identification process is composed of two major operations: (1) to obtain tentative IDs of tags in the time slice; (2) to find out what tag has the same tentative ID with the received ID. Two experiments will evaluate the performance of the above operations. Table 2 and Table 3 illustrate the results of the experiments that are executed on a personal computer with a 3.0 G Core Duo CPU and 4 GB of RAM running Windows XP SP3. Each experiment has four rounds. Each round uses a different tag size – 64 bits, 96 bits, 128 bits, and 160 bits.

No matter what tag size is used, the experiment generates tentative IDs for a tag based on the scheme described in Sect. 6.2. Suppose that the original ID of a tag T_i is RID_{T_i}. Also, the root key of the tag is RK_{T_i}. In k-th time slice of period p, the tag has tentative ID $E_{E_p(RK_{T_i})||k}(RID_{T_i})$ where this experiment uses RC4 as the encryption function E. The experiment is implemented with Java programming language. To reduce the cost of tag data management, data is stored in a SQL2005 database. Therefore, SQL commands can access the data.

The first experiment tests tag tentative ID generation as

Table 2 Average time (in m-seconds) for generating tentative IDs by number of tags and size of an ID.

size (bits)	64	96	128	160
100000	29	29	29	31
150000	42	44	45	44
200000	56	55	56	61
250000	70	71	75	76
300000	82	84	89	88

Table 3 Average time (in m-seconds) for searching a tentative ID by number of tags and size of an ID.

size (bits)	64	96	128	160
100000	21.9	21.8	23.4	25
150000	31.3	32.8	34.4	34.4
200000	40.6	43.7	45.3	46.8
250000	51.6	53.1	57.8	59.4
300000	61	64.1	68.7	71.9

the number of tags increases. Each round randomly generates 300,000 tags and stores the IDs and associated keys of the tags in a database. The experiment measures the time to generate tentative IDs for the tags and to store the tentative IDs in the database. Table 2 shows that the time to obtain tentative IDs of tags in a specified time slice increases linearly as the number of tags increases when the size of a tag ID is 96 bits. Moreover, it needs 82, 84, 89, and 90 seconds to obtain tentative IDs of 300,000 tags when the tag ID size is 64 bits, 96 bits, 128 bits, and 160 bits, respectively.

The second experiment evaluates the performance of finding a specified tentative ID from the database. Table 3 shows the results of the experiment. When the number of tags is 300,000, the process takes 61, 64.1, 68.7, and 71.9 milliseconds to find a specified tentative ID with a tag size of 64 bits, 96 bits, 128 bits, and 160 bits, respectively. Because tentative IDs are not changed in a time slice, a Back-End Server only needs to do the operation once every time slice. Therefore, the proposed approach is suitable for applications that have one million total users and 4000 different user appearances in a time slice when the size of a time slice is 5 minutes. Note that a distributed system architecture can enhance the scalability of the proposed approach. For example, the load of generating tentative IDs can be distributed on a cluster of Back-End Servers. The details are outside the scope of this article.

8. Conclusions and Future Work

This work presented a novel technical and legal approach that responds to concerns regarding the privacy of personal data in RFID systems. Current RFID privacy enhancing technologies generally focus on protecting personal data stored in RFID tags and preventing organizations from tracking a person through the information emitted by RFID tags. A method is still needed to allow customers to enter into agreements with organizations and efficiently control the RFID activities of organizations. In light of these deficiencies, this work extends the P3P and OPDL frameworks and applies the frameworks to the RFID environment . In the proposed approach, organizations must obtain licenses before collecting personal data using RFID technologies. Digitalized and standard licenses can be checked automatically to ensure that collection and use of personal data is based on consent. While individuals can easily control who has a license to collect and use their data and the content of licenses, the proposed approach is an efficient and flexible way to overcome the deficiencies in current privacy protection technologies for RFID systems.

Other than a concrete implementation of the proposed framework, many tasks must be performed. First, the proposed framework only protects RFID tags carried by a person. However, organizations may deploy their own RFID tags that cannot be controlled by a Licensing Agent directly. How to apply the proposed framework to this scenario is an important challenge.

Second, an organization must write the identity of its licensing proposal to a specified memory location of a tag to be accessed by the Licensing Agent of the tag in the indirect processing model. However, a malicious person may overwrite the data written by the organization. Therefore, RFID detection tools, such as RFID Guardian [25], may be required to find the malicious person. Furthermore, as an organization can write information to a tag's specified memory location, the Licensing Agent may need to clear its memory periodically to prevent an organization from tracking the information it wrote.

Third, to ensure that an organization has a valid license for obtaining tag information, the key used for tag identification must be changed periodically. Thus, tags, Licensing Agents of users, and Back-End Servers of organizations should be synchronized to ensure that they use the same keys. If Network Transfer Protocol (NTP) cannot be used to synchronize time, time information may be needed in tags so Back-End Servers can determine which keys are used to generate temporary tag IDs.

Fourth, this study focuses on evaluating performance for a Back-End Server to identify a tag based on the licenses that the Back-End Server already has. Evaluating the waiting time for a Back-End Server to obtain a license from a user is also important. However, the waiting time depends on how long a user takes to issue a license after receiving a licensing proposal. Designing an experiment to estimate the waiting time would be an interesting future work.

Finally, while this work adopts the key search approach to prevent malicious people from identifying unauthorized RFID tags, this work also inherits the limitations of the key search approach. This may require the development of mechanisms, such as special-purpose intrusion detection systems and distributed Licensing Services and Back-End Servers, to compensate for the limitations. Additionally, the authors may consider other approaches, such as minimalist cryptography or re-encryption mentioned in Sect. 2.2. However, using other approaches may bring other deficiencies that will need to be considered in the future.

Acknowledgments

This work was supported in part by the National Science Council of Taiwan (R.O.C.) under grants NSC 96-2221-E-011-058.

References

[1] S.L. Garfinkel, A. Juels, and R. Pappu, "RFID privacy: An overview

of problems and proposed solutions," IEEE Security and Privacy, vol.3, no.3, pp.34–43, 2005.

[2] S. Kinoshita, M. Ohkubo, F. Hoshino, G. Morohashi, O. Shionoiri, and A. Kanai, "Privacy enhanced active RFID tag," Proc. International Workshop on Exploiting Context Histories in Smart Environments – ECHISE'05, Munich, Germany, May 2005.

[3] V. Lockton and R.S. Rosenberg, "RFID: The next serious threat to privacy," Ethics and Inf. Tech., vol.7, no.4, pp.221–231, 2005.

[4] F. Thiesse, "RFID, privacy and the perception of risk: A strategic framework," J. Strateg. Inf. Syst., vol.16, no.2, pp.214–232, 2007.

[5] CASPIAN, Privacy Rights Clearinghouse, ACLU, EFF, EPIC, Junkbusters, Meyda Online, and PrivacyActivism, "RFID position statement of consumer privacy and civil liberties organizations." http://www.privacyrights.org/ar/ RFIDposition.htm, 2003.

[6] M. Feldhofer, S. Dominikus, and J. Wolkerstorfer, "Strong authentication for RFID systems using the AES algorithm," Proc. Workshop on Cryptographic Hardware and Embedded Systems – CHES 2004, ed. M. Joye and J.J. Quisquater, Lecture Notes in Computer Science, vol.3156, pp.357–370, IACR, Boston, Massachusetts, USA, Aug. 2004.

[7] 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," Proc. Conference on E-Commerce Technology for Dynamic E-Business – CEC-East'04, pp.164–168, Beijing, China, Sept. 2005.

[8] A. Juels and J. Brainard, "Soft blocking: Flexible blocker tags on the cheap," WPES '04: Proc. 2004 ACM Workshop on Privacy in the Electronic Society, pp.1–7, ACM, New York, NY, USA, 2004.

[9] I. Kim, B. Lee, and H. Kim, "Privacy protection based on user-defined preferences in RFID system," Proc. International Conference on Advanced Communication Technology – ICACT'06, Phoenix Park, Korea, Feb. 2006.

[10] S. Weis, S. Sarma, R. Rivest, and D. Engels, "Security and privacy aspects of low-cost radio frequency identification systems," Proc. International Conference on Security in Pervasive Computing – SPC 2003, ed. D. Hutter, G. Müller, W. Stephan, and M. Ullmann, Lecture Notes in Computer Science, vol.2802, pp.454–469, Boppard, Germany, March 2003.

[11] P. Golle, M. Jakobsson, A. Juels, and P. Syverson, "Universal re-encryption for mixnets," Proc. The Cryptographers' Track at the RSA Conference – CT-RSA, ed. T. Okamoto, Lecture Notes in Computer Science, vol.2964, pp.163–178, San Francisco, California, USA, Feb. 2004.

[12] A. Juels, R. Rivest, and M. Szydlo, "The blocker tag: Selective blocking of RFID tags for consumer privacy," Proc. Conference on Computer and Communications Security – ACM CCS, ed. V. Atluri, pp.103–111, Washington, DC, USA, Oct. 2003.

[13] A. Juels, "Minimalist cryptography for low-cost RFID tags," Proc. International Conference on Security in Communication Networks – SCN 2004, ed. C. Blundo and S. Cimato, Lecture Notes in Computer Science, vol.3352, pp.149–164, Amalfi, Italia, Sept. 2004.

[14] M. Ohkubo, K. Suzuki, and S. Kinoshita, "Cryptographic approach to "privacy-friendly" tags," Proc. RFID Privacy Workshop, MA, USA, Nov. 2003.

[15] L. Cranor, M. Langheinrich, M. Marchiori, M. Presler-Marshall, and J. Reagle, "Platform for Privacy Preference (P3P)," W3C Recommendations, 2002. Retrieved from http://www.w3c.org/TR/P3P/

[16] P. Hensley, M. Metral, U. Shardanand, D. Converse, and M. Meyers, "Proposal for an open profiling standard," 1997. Retrived from http://www.w3.org/TR/ NOTE-OPS-FrameWork.html

[17] D.M. Kristol, "HTTP Cookies: Standards, privacy, and politics," ACM Trans. Internet Technology (TOIT), vol.1, no.2, pp.151–198, Nov. 2001.

[18] W3C, "Removing data transfer from P3P," 1999. Retrieved from http://www.w3c.org/P3P/data-transfer.html

[19] EPIC and Junkbuster, "Pretty poor privacy: An assessment of P3P and internet privacy," http://www.epic.org/reports/prettypoorprivacy.html, June 2000.

[20] S.C. Cha and Y.J. Joung, "Online personal data licensing," Proc. 3rd International Conference of Law and Technology (LAWTECH2002), pp.28–33, Boston, USA, 2002.

[21] S.C. Cha and Y.J. Joung, "From P3P to OPDL," Proc. 3rd Workshop on Privacy Enhancing Technologies (PET2003), Dresden, Germany, March 2003.

[22] A. Juels, "RFID security and privacy: a research survey," IEEE J. Sel. Areas Commun., vol.24, no.2, pp.381–394, Feb. 2006.

[23] M. Rieback, B. Crispo, and A. Tanenbaum, "The evolution of RFID security," IEEE Pervasive Computing, vol.5, no.1, pp.62–69, Jan.-March 2006.

[24] S. Spiekermann and S. Evdokimov, "Critical rfid privacy-enhancing technologies," Computing in Science and Eng., vol.7, no.2, pp.56–62, 2009.

[25] M. Rieback, G. Gaydadjiev, B. Crispo, R. Hofman, and A. Tanenbaum, "A platform for RFID security and privacy administration," Proc. USENIX/SAGE Large Installation System Administration Conference, pp.89–102, Washington DC, USA, Dec. 2006.

[26] L. Batina, J. Guajardo, T. Kerins, N. Mentens, P. Tuyls, and I. Verbauwhede, "Public-key cryptography for RFID-tags," Proc. International Workshop on Pervasive Computing and Communication Security – PerSec 2007, pp.217–222, New York, USA, March 2007.

[27] EPCGlobal, "EPC radio-frequency identity protocols class-1 generation-2 UHF RFID: Protocols for communications at 860MHz – 960MHz," 2005. EPC Global Specification for RFID Air Interface.

[28] G. Avoine and P. Oechslin, "A scalable and provably secure hash based RFID protocol," Proc. International Workshop on Pervasive Computing and Communication Security – PerSec 2005, pp.110–114, Kauai Island, Hawaii, USA, March 2005.

[29] S.A. Juels and A. Weis, "Defining strong privacy for RFID," Proc. Fifth Annual IEEE International Conference on Pervasive Computing and Communications Workshops – PerCom Workshops' 07, pp.342–347, March 2007.

[30] T. Dimitriou, "A lightweight RFID protocol to protect against traceability and cloning attacks," Proc. Conference on Security and Privacy for Emerging Areas in Communication Networks – SecureComm, Athens, Greece, Sept. 2005.

[31] T. Dimitriou, "A secure and efficient rfid protocol that could make big brother (partially) obsolete," Proc. International Conference on Pervasive Computing and Communications – PerCom 2006, Pisa, Italy, March 2006.

[32] D. Molnar and D. Wagner, "Privacy and security in library RFID: issues, practices, and architectures," CCS '04: Proc. 11th ACM conference on Computer and communications security, pp.210–219, New York, NY, USA, 2004.

[33] R. Langheinrich and M. Marti, "Practical minimalist cryptography for RFID privacy," IEEE Syst. J., vol.1, no.2, pp.115–128, Dec. 2007.

[34] A. Juels and R. Pappu, "Squealing euros: Privacy protection in RFID-enabled banknotes," Proc. Financial Cryptography – FC'03, ed. R.N. Wright, Lecture Notes in Computer Science, vol.2742, pp.103–121, Le Gosier, Guadeloupe, French West Indies, IFCA, Jan. 2003.

[35] A. Juels, P.F. Syverson, and D.V. Bailey, "High-power proxies for enhancing rfid privacy and utility," Privacy Enhancing Technologies, pp.210–226, 2005.

[36] L. Cranor, B. Dobbs, S. Egelman, G. Hogben, J. Humphrey, M. Langheinrich, M. Marchiori, M. Presler-Marshall, J. Reagle, M. Schunter, D.A. Stampley, and R. Wenning, "The platform for privacy preferences 1.1 (P3P1.1) specification," W3C Working Group Note, 2006.

[37] L. Cranor, M. Langheinrich, and E. Zurich, "A P3P preference exchange language 1.0 (APPEL1.0)," W3C Working Draft, 2002. Retrieved from http://www.w3c.org/TR/ P3P-preferences.html

Shi-Cho Cha received his B.S. and Ph.D. in Information Management from the National Taiwan University in 1996 and 2003. He is currently an assistant professor at the Department of Information in the National Taiwan University of Science and Technology, where he has been a faculty member since 2006. He is a certified PMP, CISSP, and CISM. From 2003–2006, he was a manager at Pricewaterhouse-Coopers, Taiwan. His current research interests are in the area information security management, identity management, and RFID privacy.