12-2011

On two RFID privacy notions and their relations

Yingjiu LI
Singapore Management University, yjli@smu.edu.sg

Robert H. DENG
Singapore Management University, robertdeng@smu.edu.sg

Junzuo LAI
Singapore Management University, jzlai@smu.edu.sg

Changshe MA
South China Normal University

DOI: https://doi.org/10.1145/2043628.2043631

On Two RFID Privacy Notions and Their Relations

YINGJIU LI, ROBERT H. DENG, and JUNZUO LAI, Singapore Management University
CHANGSHE MA, South China Normal University

Privacy of RFID systems is receiving increasing attention in the RFID community. Basically, there are two kinds of RFID privacy notions in the literature: one based on the indistinguishability of two tags, denoted as ind-privacy, and the other based on the unpredictability of the output of an RFID protocol, denoted as unp-privacy. In this article, we first revisit the existing unpredictability-based RFID privacy models and point out their limitations. We then propose a new RFID privacy model, denoted as unp*-privacy, based on the indistinguishability of a real tag and a virtual tag. We formally clarify its relationship with the ind-privacy model. It is proven that ind-privacy is weaker than unp*-privacy. Moreover, the minimal (necessary and sufficient) condition on RFID tags to achieve unp*-privacy is determined. It is shown that if an RFID system is unp*-private, then the computational power of an RFID tag can be used to construct a pseudorandom function family provided that the RFID system is complete and sound. On the other hand, if each tag is able to compute a pseudorandom function, then the tags can be used to construct an RFID system with unp*-privacy. In this sense, a pseudorandom function family is the minimal requirement on an RFID tag's computational power for enforcing RFID system privacy. Finally, a new RFID mutual authentication protocol is proposed to satisfy the minimal requirement.

Categories and Subject Descriptors: C.2.0 [**General**]: *Security and protection*; D.4.6 [**Operating Systems**]: Security and protection—*Cryptographic controls*

General Terms: Security, Design

Additional Key Words and Phrases: RFID, privacy, pseudorandom function

ACM Reference Format:
Li, Y., Deng, R. H., Lai, J., and Ma, C. 2011. On two RFID privacy notions and their relations. ACM Trans. Info. Syst. Sec. 14, 4, Article 30 (December 2011), 23 pages.
DOI = 10.1145/2043628.2043631 http://doi.acm.org/10.1145/2043628.2043631

1. INTRODUCTION

Radio Frequency Identification (RFID) has been widely envisioned as an inevitable replacement of barcodes and other consumer labeling techniques for automatic object identification. An RFID system consists of small devices called RFID tags, one or more RFID readers and a back-end database. Unlike barcodes, each RFID tag records a sufficiently long bitstring to uniquely identify the tag or its bearer. Readers communicate with tags using RF signals at a distance from a few inches to several feet. Since RF signals are invisible and penetrating, RFID systems provide a perfect environment for

This work is partly supported by A*Star SERC under grant number 082 101 0022 in Singapore, and also supported in part by the Office of Research at Singapore Management University.
C. Ma's work is partly supported by NSFC under grant number 61070217.
Authors' addresses: Y. Li, R. H. Deng, and J. Lai, School of Information Systems, Singapore Management University, 80 Stamford Road, Singapore 178902; email: yjli@smu.edu.sg; C. Ma, School of Computer, South China Normal University, Guangzhou, China 510631.

attackers. The prevalence of RFID technologies introduces various serious risks and poses unique security concerns [Juels 2006; Peris-Lopez et al. 2006].

Security problems in RFID systems can be classified into two types. The first is concerned with attacks that aim to wipe out the functioning of the system. The second type, the one that interests us here, is related to privacy. In particular, unauthorized tracking of RFID system users and RFID tag bearers via clandestine scanning, where an adversary uses an unauthorized reader collecting RF waves to track the movement of RFID tags, has been recognized as one of the most imperative privacy concerns in the deployments of RFID systems. A privacy-preserving RFID system should therefore provide anonymity (i.e., hiding of a tag's identity) as well as unlinkability of the protocol transcripts of a tag. Much effort [Ateniese et al. 2005; Avoine et al. 2005; Garfinkel et al. 2005; Juels et al. 2008; Juels et al. 2003; Molnar and Wagner 2004; Spiekermann and Evdokimov 2009] has been made to address the privacy issues in RFID systems, mostly focused on two aspects: one is to construct RFID protocols [Ohkubo et al. 2004; Tsudik 2006; Peris-Lopez et al. 2006; Tsudik 2007] that are compatible with the constraints of tags; the other is to formalize privacy models for RFID systems. In the former aspect, dozens of protocols have been proposed in the literature, while many of them are reported to have privacy flaws. In the latter aspect, two major RFID privacy notions have been proposed: one based on the indistinguishability of two tags [Juels and Weis 2007], denoted as ind-privacy, and the other based on the unpredictability of the output of an RFID protocol [Ha et al. 2008], denoted as unp-privacy. In this article, we closely examine the two types of privacy notions, explain why many existing protocols have privacy flaws, and construct an efficient privacy-preserving protocol.

One fundamental problem we investigate is to find a reasonable RFID privacy notion that is easy to work with. In addition, to understand which level of privacy an RFID system provides, it is critical to clarify the relationships among different RFID privacy notions.

The other problem we investigate regards the minimal cryptographic function that needs to be supported in tags in order to guarantee the privacy of RFID systems. A definite answer to this problem will help design low-cost tags for RFID systems with privacy. It will also help explain why many existing RFID protocols that do not support the minimal cryptographic function have privacy flaws.

1.1. Our Contributions

In this article, we address the above two basic problems for RFID privacy and make the following contributions.

(1) We revisit the existing unpredictability-based RFID privacy models, unp-privacy for short, which is the first unpredictability-based RFID privacy model proposed by Ha et al. [2008]. We point out the limitations of the unp-privacy model and propose a new privacy model, denoted as unp*-privacy, based on the indistinguishability of a real tag and a virtual tag.

The underlying intuition of the unp-privacy model is that the output of a tag looks random. Informally, for an unp-private RFID protocol, it requires that every probabilistic polynomial time (PPT) adversary not be able to distinguish the output of a tag from a random value. However, van Deursen and Radomirović [2009] showed that an adversary can trace a tag by observing the output of the reader and break the privacy of a RFID protocol which is proved unp-private.

To rectify the deficiency of the unp-privacy model, Ma et al. [2009] proposed a refined unp-privacy model, denoted as unp$'$-privacy. Compared with the original unp-privacy model, an unp$'$-private RFID protocol requires that every PPT

adversary not be able to distinguish transcripts of a single protocol session between the reader and a tag from random values. In the unp′-privacy definition, an adversary is not allowed to issue multiple oracle queries on the challenge tag. Such a limitation on the adversary is too restrictive in most practical applications. We demonstrate this flaw of the unp′-privacy model with a counterexample 3-round RFID protocol that has clear weakness with respect to privacy but is proven to satisfy unp′-privacy. In the protocol, each tag maintains a state which is either $s = 0$ or $s = 1$. A tag is in state $s = 0$ under normal circumstances and in state $s = 1$ otherwise. An active attacker first sets a tag's state to $s = 1$ by simply modifying a protocol message and then traces the tag in the subsequent protocol session. However, the transcripts of a protocol session between the reader and a tag appear random. In other words, the protocol is unp′-private but suffers from an apparent weakness in privacy.

In our unp⋆-privacy model, an adversary is allowed to issue multiple oracle queries on the challenge tag. The underlying intuition of unp⋆-privacy is that every PPT adversary cannot distinguish the transcripts of multiple protocol sessions between the reader and a real tag from those between the reader and a virtual tag, which are random values.

(2) We prove that unp⋆-privacy implies ind-privacy. Since there is an essential difference between these two notions, we bridge the gap by introducing a restricted ind-privacy model, ind′-privacy, which is proven to be equivalent to ind-privacy. Then, we prove that unp⋆-privacy implies ind′-privacy. Moreover, we show that ind-privacy does not imply unp⋆-privacy by constructing an RFID system which is ind-private but not unp⋆-private.

(3) We determine the minimal condition for RFID tags to achieve unp⋆-privacy in an RFID system. It is shown that if an RFID system is unp⋆-private, then each RFID tag can be used to construct a pseudorandom function (PRF) family or its equivalents provided that the RFID system is complete and sound. On the other hand, if every tag is endowed with the capability to compute a PRF or its equivalents, then an RFID system with unp⋆-privacy can be constructed. The minimal requirement on the computational power for RFID tags shows that unp⋆-privacy cannot be guaranteed without implementing appropriate cryptographic functions. This explains why many lightweight RFID protocols are vulnerable to privacy related attacks.

(4) According to the minimal condition on RFID tags, we design an efficient RFID protocol with unp⋆-privacy and mutual authentication. In the case that a tag has not been desynchronized (e.g., due to attacks) since the last successful read of the tag, our protocol requires the minimal computational cost for identifying the tag (in exact match). In the case that a tag has just been desynchronized, our protocol requires exhaustive search for identifying the tag as in most of the existing protocols.

We emphasize that ind-privacy [Juels and Weis 2007] is the correct notion for RFID privacy and it may not be necessary to have more strengthened notions on RFID privacy in practice. However, this notion is difficult to apply in security analysis in the sense that no RFID protocols have been directly proven to satisfy the ind-privacy definition. The notions of unp-privacy [Ha et al. 2008] and unp′-privacy [Ma et al. 2009] aim to improve workability in security analysis of RFID protocols. Unfortunately, these two notions are not ideal and have deficiencies. The new unp⋆ notion proposed in this paper is based on unp-privacy and unp′-privacy but eliminates their shortcomings. This new notion is formulated to be easier to work with and it implies ind-privacy. Another merit of unp⋆ is that it is equivalent to the existence of PRF. This sheds new light on what

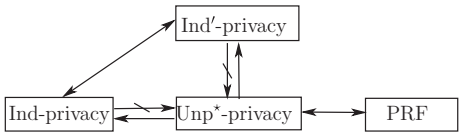

Fig. 1. Relations among privacy notions.

it takes, from a theoretical point of view, to have secure and privacy-preserving RFID systems.

1.2. Related Work

Avoine [2005] first formalized the adversary model in RFID systems and proposed very general and flexible definitions of RFID privacy. Based on the formal adversary model, Juels and Weis [2007] defined the notion of strong privacy. The aim of Avoine [2005] was to capture a range of adversarial abilities, while Juels and Weis [2007] seeked to characterize a very strong adversary with a relatively simple definition. The privacy notion in Juels and Weis [2007] is based on the indistinguishability of two RFID tags, denoted as ind-privacy. However, to our knowledge, there is no RFID protocol that has been *directly* proven to be ind-private; On the other hand, if an RFID protocol is not ind-private, it can be checked against the ind-privacy model easily.

Damgård and Pedersen [2008] considered the completeness and soundness for RFID systems. Vaudenay [2007] considerd side-channel attacks in his RFID privacy model and proposed eight privacy classes, which were later consolidated to three by Ng et al. [2008]. Paise and Vaudenay [2008] extended the definitions in Vaudenay [2007] to address mutual authentication. However, the privacy definitions in Vaudenay [2007], Ng et al. [2008], and Paise and Vaudenay [2008] contradict reader authentication for any privacy notion that allows tag corruption [Paise and Vaudenay 2008].

Ha et al. [2008] proposed a different privacy model based on the unpredictability of tag outputs, denoted as unp-privacy. Unfortunately, this model was later shown to have some deficiencies in its definition [van Deursen and Radomirović 2009]. Recently, Ma et al. [2009] proposed a refined unp-privacy model for RFID systems, which we denote as unp′-privacy for convenience. In this article, we show that the unp′-privacy model has a deficiency when applied to 3-round RFID protocols.

Since it is extremely important to reduce the cost of RFID tags in practice, significant effort has been made to construct lightweight RFID protocols for low-cost tags such as EPC Class-1 Generation-2 tags. Sarma et al. [2003] analyzed the gate complexity of the embedded chip with respect to the cost per tag. However, no research has been conducted on the minimal computation power that should be endowed on tags to ensure privacy.

To provide privacy for RFID systems, typical lightweight RFID protocols [Karthikeyan and Nesterenko 2005; Duc et al. 2006; Chien and Chen 2007; Konidala et al. 2007] exploit simple operations such as XOR, bit inner product, 16-bit pseudo-random number generator (PRNG), and cyclic redundancy checksum (CRC). Most of these protocols, however, have privacy flaws [Peris-Lopez et al. 2008; van Deursen and Radomirovic 2008]. Juels [2004] proposed a pseudonym-throttling scheme without using any cryptographic functions for tags. The privacy of this scheme is guaranteed under the condition that the rate of pseudonym releases is slowed down to a certain level. If this condition does not hold, the privacy of this scheme cannot be ensured. While specific attacks have been discovered to break the privacy for different lightweight protocols, no theoretical model has been provided in the literature to explain why those protocols are vulnerable to privacy attacks. In this paper, we prove that to guarantee the privacy of an RFID system, it is necessary and sufficient to endow each tag with

the ability to compute a pseudorandom function; thus it explains why many existing lightweight protocols have privacy problems. We also provide an example to show how to design an efficient protocol that provides privacy with minimal requirement on RFID tags.

1.3. Organization

The rest of the article is organized as follows. In Section 2, we define the mathematical notations and pseudorandom functions used in this article. In Section 3, we introduce two privacy models, ind-privacy and unp-privacy, for RFID systems. We revisit the unp-privacy model and its recent improvement and clarify their limitations in Section 4. In Section 5, We introduce our privacy model, unp*-privacy, to avoid the limitations of previous unp-privacy models and establish its relation with the ind-privacy model. In Section 6, we show that the minimal requirement to guarantee unp*-privacy is equipping each tag with the ability to compute a pseudorandom function. We also provide an efficient construction of RFID protocol with unp*-privacy according to the minimal requirement on tags. In Section 7, we conclude this article and discuss some open problems.

2. PRELIMINARIES

In this section, we introduce the preliminaries that will be used in this article, including mathematical notations and pseudorandom functions.

2.1. Mathematical Notations

If $A(\cdot,\cdot,\ldots)$ is a randomized algorithm, then $y \leftarrow A(x_1, x_2, \ldots; cn)$ means that y is assigned with the unique output of the algorithm A on inputs $x_1, x_2, \ldots$ and coins cn, while $y \xleftarrow{\$} A(x_1, x_2, \ldots)$ is a shorthand for first picking cn at random and then setting $y \leftarrow A(x_1, x_2, \ldots; cn)$. Let $y \leftarrow A^{O_1,\ldots,O_n}(x_1, x_2, \ldots)$ denote that y is assigned with the output of the algorithm A which takes $x_1, x_2, \ldots$ as inputs and has oracle accesses to $O_1, \ldots, O_n$. If S is a set, then $s \in_R S$ indicates that s is chosen uniformly at random from S. If $x_1, x_2, \ldots$ are strings, then $x_1||x_2||\cdots$ denotes the concatenation of them. If x is a string, then $|x|$ denotes its bit length in binary code. If S is a set, then $|S|$ denotes its cardinality (i.e. the number of elements of S). Let $\Pr[E]$ denote the probability that an event E occurs, $\mathcal{N}$ denote the set of all integers, R denote the set of all real numbers, and ε denote the empty string.

Definition 2.1. A function $f : \mathcal{N} \to R$ is said to be *negligible* if for every $c > 0$ there exits a number $m \in \mathcal{N}$ such that $f(n) < \frac{1}{n^c}$ holds for all $n > m$.

2.2. Pseudorandom Functions

Let $F : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$ be a family of functions, where $\mathcal{K}$ is the set of keys (or indexes) of F, $\mathcal{D}$ is the domain of F, and $\mathcal{R}$ is the range of F. Let $|\mathcal{K}| = \gamma$, $|\mathcal{D}| = m$, and $|\mathcal{R}| = n$. Let $\mathsf{Rand}^{\mathcal{D}\to\mathcal{R}}$ be the family of all functions with domain $\mathcal{D}$ and range $\mathcal{R}$. A *polynomial time predictable test* (*PTPT*) for F is an experiment, where a probabilistic polynomial time algorithm T, given γ, m, n as input and with access to an oracle O_f for a function $f \in_R F$ or $f \in_R \mathsf{Rand}^{\mathcal{D}\to\mathcal{R}}$, outputs either 0 or 1. Figure 2 shows a PTPT for F.

Definition 2.2. An algorithm T passes the *PTPT* for the function family F if it correctly guesses the random bit which is selected by *PTPT* experiment, i.e. $b' = b$. The advantage of algorithm T is defined as

$$\mathrm{Adv}_T(\gamma, m, n) = \left|\Pr[b' = b] - \frac{1}{2}\right|,$$

$\mathrm{Exp}_T^{ptpt}(F, \gamma, m, n)$

1. $b \in_R \{0,1\}$
2. if $b = 1$ then $k \in_R \mathcal{K}$ and set $f = F_k$, otherwise $f \in_R \mathsf{Rand}^{\mathcal{D}\to\mathcal{R}}$
3. $b' \leftarrow T^{O_f}$

Fig. 2. Polynomial time predictable test.

where the probability is taken over the choice of f in F and the coin tosses of algorithm T.

Definition 2.3. A function family $F : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$ is said to be a pseudorandom function family if it has the following properties:

Indexing. Each function in F has a unique γ-bit key (index) associated with it. It is easy to select a function $f \in F$ randomly if γ random bits are available.

Polynomial Time Evaluation. There exists a polynomial time algorithm such that, given input of a key (index) $k \in \mathcal{K}$ and an argument $x \in \mathcal{D}$, it outputs $F(k, x)$.

Pseudorandomness. No probabilistic polynomial time algorithm T can pass the *PTPT* for F with non-negligible advantage.

For convenience, we use $F_k(x)$ and $F(k, x)$ interchangeably for a PRF family F in this paper.

3. IND-PRIVACY AND UNP-PRIVACY OF RFID SYSTEMS

In this section, we give a formal model for RFID system and two formal definitions for RFID system privacy, ind-privacy and unp-privacy.

3.1. Model of RFID System

For simplicity, we consider an RFID system comprising of a single legitimate reader[1] R and a set of ℓ tags $\mathcal{T}_1, \ldots, \mathcal{T}_\ell$. The reader and the tags are probabilistic polynomial time interactive Turing machines. Typically, each tag is a passive transponder identified by a unique ID and has only limited memory which can be used to store only several keys and/or some state information. The reader is composed of one or more transceivers and a backend processing subsystem. In this paper, we assume that the reader is secure, which means that an adversary cannot obtain any information about the RFID system from the legitimate reader except the information obtained from RFID communications and tags (in other words, the legitimate reader is a "black-box" to an adversary).

Canonical RFID Protocol. Every tag exchanges messages with the reader through a protocol π. In the following, we use *canonical protocol*[2] to describe a generic privacy-preserving challenge-response RFID authentication protocol as shown in Figure 3. The protocol π is invoked by the reader R sending a challenge message c to the tag $\mathcal{T}_i$, which upon receiving the challenge message c responds with a message r, where r is computed according to the tag's key $k_{\mathcal{T}_i}$, the challenge message c, its coin toss $cn_{\mathcal{T}_i}$, and its internal state $s_{\mathcal{T}_i}$. We write r as $r = F_{\mathcal{T}_i}(k_{\mathcal{T}_i}, cn_{\mathcal{T}_i}, s_{\mathcal{T}_i}, c)$, where $F_{\mathcal{T}_i}$ is a function computed by the tag. This protocol can be executed in two or three rounds. In the

[1]It's straightforward to extend the model to include multiple legitimate readers. Notice that an adversary can use its own readers to interact with tags.

[2]To the best of our knowledge, our canonical protocol can be used to describe most of existing RFID protocols except some of the HB family protocols [Hopper and Blum 2001; Juels and Weis 2005; Katz and Shin 2006], which require more than three rounds to authenticate each tag in a statistical sense. We consider it an open problem to extend our research to those protocols.

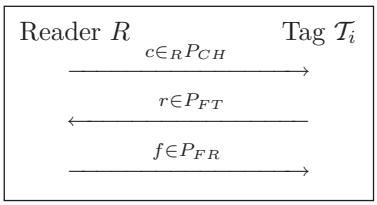

Fig. 3. Canonical RFID protocol.

third round, if exists, the reader sends the tag the final message f, which is computed according to the reader's internal state s_R, it's coin toss cn_R, the challenge message c, and the tag's response r. We write it as $f = F_R(k_R, cn_R, s_R, c, r)$, where F_R is a function computed by the reader based on a key k_R, which may or may not be the same as $k_{\mathcal{T}_i}$. Let P_{CH}, P_{FT}, P_{FR}, P_K, P_{CN}, P_S denote the challenge message space, the range of function $F_{\mathcal{T}}$, the final message space, the key space of the tag, the coin space of the tag, and the state information space of the tag, respectively. The view of an adversary about the protocol π is the set $\{(c, r, f)\}$. Throughout this article, we only consider RFID protocols in this canonical form.

Definition 3.1. An RFID system RS is defined to be a tuple $(R, \mathcal{T}, \mathsf{ReaderSetup}, \mathsf{TagSetup}, \mathsf{ReaderStart}, \mathsf{TagCompute}, \mathsf{ReaderCompute}, \pi)$, where

ReaderSetup(κ). It is a setup procedure which generates the system parameter σ and key k_R (if needed) for the reader R according to the security parameter κ. It also setups a database for the reader R to store necessary information for tag identification.

TagSetup$(\mathcal{T}_i, \kappa)$. It is a setup procedure which generates key $k_{\mathcal{T}_i}$ for a tag $\mathcal{T}_i$ and sets the tag's initial internal state st_0. It also associates the tag $\mathcal{T}_i$ with its unique ID as well as other necessary information such as tag key and/or tag state information as a record in the database of reader R.

ReaderStart. It is an algorithm for reader R to generate a session identifier sid of a new session and a challenge message c_{sid} of the session.

TagCompute$(\mathcal{T}_i, sid, c_{sid})$. It is an algorithm for tag $\mathcal{T}_i$ to compute its response r_{sid}, taking a session identifier sid and challenge message c_{sid} as input.

ReaderCompute(sid, c_{sid}, r_{sid}). It is an algorithm for the reader R to compute the final message f_{sid}, taking a session identifier sid, challenge message c_{sid} and response message r_{sid} as input.

Protocol $\pi(R, \mathcal{T}_i)$. It is a canonical interactive protocol between the reader R and the tag $\mathcal{T}_i$. We associate each session of protocol π with a unique session identifier sid. As an abusing of the notation, let

$$(c_{sid}, r_{sid}, f_{sid}) \leftarrow \pi(R, \mathcal{T}_i, sid)$$

denote the running of protocol π between R and $\mathcal{T}_i$ with challenge message c_{sid} and session identifier sid. The external output of the protocol $\pi(R, \mathcal{T}_i)$ is the tuple $(c_{sid}, r_{sid}, f_{sid})$. A tuple (c, r, f) is said to be a valid set of protocol messages of $\pi(R, \mathcal{T}_i)$ if there exists a session identifier sid such that

$$\pi(R, \mathcal{T}_i, sid) = (c, r, f).$$

A tag $\mathcal{T}_i$ is said to be *accepted* if its corresponding record is identified by the reader R in its database upon performing the protocol $\pi(R, \mathcal{T}_i)$.

Note that, the ReaderStart, TagCompute and ReaderCompute algorithms can be obtained from the protocol π. For convenience, we use $RS = (R, \mathcal{T}, \mathsf{ReaderSetup}, \mathsf{TagSetup}, \pi)$ to denote an RFID system.

Experiment $\mathbf{Exp}_{\mathcal{A}}^{sound}[\kappa, \ell, q, s, v]$
1. setup the reader R and a set of tags $\mathcal{T}$ with $|\mathcal{T}| = \ell$;
2.$\{(c_{sid^*}, r_{sid^*}, f_{sid^*}), \mathcal{T}_j\} \leftarrow \mathcal{A}^{O_1, O_2, O_4}(R, \mathcal{T})$.

Fig. 4. Soundness experiment.

3.2. Description of Adversary

In a nutshell, an adversary $\mathcal{A}$ is a probabilistic polynomial time interactive Turing machine that is allowed to perform oracle queries during attacks. In the following, we specify what kinds of oracles the adversary $\mathcal{A}$ is permitted to query.

InitReader. It invokes the reader R to start a session of protocol π and generate a session identifier sid and challenge message $c_{sid} \in_R P_{CH}$. The reader returns the session identifier sid and the challenge message c_{sid}.

InitTag$(\mathcal{T}_i, sid, c_{sid})$. It invokes tag $\mathcal{T}_i$ to start a session of protocol π with session identifier sid and challenge message $c_{sid} \in P_{CH}$. The tag $\mathcal{T}_i$ responds with the session identifier sid and a message $r_{sid} \in P_{FT}$.

SetTag$(\mathcal{T}_i)$. It updates different key and state information to tag $\mathcal{T}_i$ and returns the tag's current key and internal state information.

SendRes(sid, c, r). It takes the challenge and response messages c, r with session identifier sid as input and (in three-round protocol) returns the reader's final message f_{sid}.

Let O_1, O_2, O_3 and O_4 denote InitReader, InitTag, SetTag and SendRes oracles, respectively.

Remark 1. The four kinds of queries defined above can be used to model most, if not all, of the attacks to RFID communications or tags, including eavesdropping, alteration of communication messages, replay attacks, corruption of tags, and physical or side-channel attacks to tags. For example, eavesdropping can be modeled as follows: first call InitReader() to get (sid, c_{sid}), then call InitTag(sid, c_{sid}) to get (sid, r_{sid}), and finally call SendRes(sid, c_{sid}, r_{sid}) to get f_{sid}. For another example, any tag key compromise due to tag corruption, physical or side-channel attacks can be modeled by sending the SetTag query to the tag.

3.3. Completeness and Soundness of RFID Systems

Here, we review the definitions of completeness and soundness of RFID systems presented in [Damgård and Pedersen 2008]. Informally, completeness means that a legitimate tag will always be accepted by the legitimate reader, and the soundness means that only a legitimate tag will be accepted by the legitimate reader.

Definition 3.2 *Completeness.* Assume that at the end of every session sid the output of that session is the tuple $(c_{sid}, r_{sid}, f_{sid})$, where r_{sid} is correctly generated by a legitimate tag. Completeness means that the reader outputs "accept" with probability 1 for any such session.

Next, consider the soundness experiment $\mathbf{Exp}_{\mathcal{A}}^{sound}[\kappa, \ell, q, s, v]$ as shown in Figure 4, where ℓ, q, s, v are experiment parameters. The adversary $\mathcal{A}$ is given an RFID system RS as input and is allowed to launch O_1, O_2 and O_4 oracle queries without exceeding q, s and v overall calls, respectively. At the end of the experiment, $\mathcal{A}$ outputs a tuple $(c_{sid^*}, r_{sid^*}, f_{sid^*})$ and a tag $\mathcal{T}_j \in \mathcal{T}$. Let E denote the event that r_{sid^*} is not sent by tag

Experiment $\mathbf{Exp}^{ind}_{\mathcal{A}}[\kappa, \ell, q, s, u, v]$
1. setup the reader R and a set of tags $\mathcal{T}$ with $|\mathcal{T}| = \ell$;
2. $\{\mathcal{T}_i, \mathcal{T}_j, st\} \leftarrow \mathcal{A}_1^{O_1,O_2,O_3,O_4}(R, \mathcal{T})$; //*learning stage*
3. set $\mathcal{T}' = \mathcal{T} - \{\mathcal{T}_i, \mathcal{T}_j\}$;
4. $b \in_R \{0, 1\}$;
5. if $b = 0$ then $\mathcal{T}_c = \mathcal{T}_i$, else $\mathcal{T}_c = \mathcal{T}_j$;
6. $b' \leftarrow \mathcal{A}_2^{O_1,O_2,O_3,O_4}(R, \mathcal{T}', st, \mathcal{T}_c)$; //*guess stage*
7. the experiment outputs 1 if $b' = b$, 0 otherwise.

Fig. 5. Ind-privacy experiment.

$\mathcal{T}_j$ in session sid^* while the reader R accepts the tag $\mathcal{T}_j$ in session sid^* with protocol message tuple $(c_{sid^*}, r_{sid^*}, f_{sid^*})$.

Definition 3.3. An adversary $\mathcal{A}$ (ϵ, t, q, s, v)-breaks the soundness of the RFID system RS if the probability that event E occurs is at least ϵ and the running time of $\mathcal{A}$ is at most t.

Definition 3.4 *Soundness*. The RFID system RS provides (ϵ, t, q, s, v)-soundness if there exists no adversary $\mathcal{A}$ which can (ϵ, t, q, s, v)-break the soundness of RS.[3]

3.4. Ind-Privacy

Juels and Weis [Juels and Weis 2007] presented an indistinguishability-based RFID privacy model which is reminiscent of the classic indistinguishability under chosen-plaintext attack (IND-CPA) and under chosen-ciphertext attack (IND-CCA) in cryptosystem security.

Figure 5 illustrates the ind-privacy experiment $\mathbf{Exp}^{ind}_{\mathcal{A}}[\kappa, \ell, q, s, u, v]$ ($\mathbf{Exp}^{ind}_{\mathcal{A}}$, for simplicity), in which an adversary $\mathcal{A}$ is comprised of a pair of algorithms $(\mathcal{A}_1, \mathcal{A}_2)$ and runs in two stages. Throughout the experiment, the adversary $\mathcal{A}$ is allowed to launch O_1, O_2, O_3 and O_4 oracle queries without exceeding q, s, u and v overall calls, respectively. The experiment proceeds as follows. At first, the experiment initializes the RFID system by producing a reader R and a set of tags $\mathcal{T} = \{\mathcal{T}_1, \ldots, \mathcal{T}_\ell\}$ according to the security parameter κ. Then, in the *learning stage*, algorithm $\mathcal{A}_1$ outputs a state information st and a pair of tags $\{\mathcal{T}_i, \mathcal{T}_j\}$ to which it has not sent SetTag queries. Next, the experiment selects a random bit b and sets the challenge tag $\mathcal{T}_c = \mathcal{T}_i$ if $b = 0$, and $\mathcal{T}_c = \mathcal{T}_j$ otherwise. Finally, in the *guess stage*, algorithm $\mathcal{A}_2$ is asked to guess the random bit b by outputting a bit b'. During this stage, algorithm $\mathcal{A}_2$ is allowed to launch O_1, O_2, O_3 and O_4 oracle queries to $\mathcal{T}_c$ and the tag set $\mathcal{T}' = \mathcal{T} - \{\mathcal{T}_i, \mathcal{T}_j\}$ with the restriction that it cannot query SetTag($\mathcal{T}_c$).

Definition 3.5. The advantage of adversary $\mathcal{A}$ in the experiment $\mathbf{Exp}^{ind}_{\mathcal{A}}[\kappa, \ell, q, s, u, v]$ is defined as:

$$\mathrm{Adv}^{ind}_{\mathcal{A}}(\kappa, \ell, q, s, u, v) = \left| \Pr\left[\mathbf{Exp}^{ind}_{\mathcal{A}}[\kappa, \ell, q, s, u, v] = 1\right] - \frac{1}{2} \right|,$$

[3]Our definition of soundness is compatible with the weak soundness introduced in [Damgård and Pedersen 2008], in which strong soundness has also been defined (strong soundness allows an adversary to launch SetTag oracle, or O_3, queries to corrupt any tags except the tag $\mathcal{T}_j$).

Experiment $\mathbf{Exp}_{\mathcal{A}}^{unp}[\kappa, \ell, q, s, u, v]$
1. setup the reader R and a set of tags $\mathcal{T}$ with $|\mathcal{T}| = \ell$;
2. $\{\mathcal{T}_c, st\} \leftarrow \mathcal{A}_1^{O_1,O_2,O_3,O_4}(R, \mathcal{T})$; *//learning stage*
3. $b \in_R \{0, 1\}$;
4. if $b = 0$ then $r^* \in_R P_{RS}$,
   else r^* is taken from $(c^*, r^*, f^*) \leftarrow \pi(R, \mathcal{T}_c, sid)$;
5. $b' \leftarrow \mathcal{A}_2(r^*, st)$; *//guess stage*
6. the experiment outputs 1 if $b' = b$, 0 otherwise.

Fig. 6. Unp-privacy experiment.

where the probability is taken over the choice of tag set $\mathcal{T}$ and the coin tosses of the adversary $\mathcal{A}$.

Definition 3.6. An adversary $\mathcal{A}$ $(\epsilon, t, q, s, u, v)$-breaks the ind-privacy of RFID system RS if the advantage $\mathrm{Adv}_{\mathcal{A}}^{ind}(k, \ell, q, s, u, v)$ of $\mathcal{A}$ in the experiment $\mathbf{Exp}_{\mathcal{A}}^{ind}$ is at least ϵ and the running time of $\mathcal{A}$ is at most t.

Definition 3.7 $(\epsilon, t, q, s, u, v)$-*Ind-Privacy.* An RFID system RS is said to be $(\epsilon, t, q, s, u, v)$-ind-private if there exists no adversary who can $(\epsilon, t, q, s, u, v)$-break the ind-privacy of RS.

Remark 2. The ind-privacy implies that an adversary cannot distinguish between any two tags in the tag set $\mathcal{T}$ which the adversary has not corrupted. This definition can be easily extended to the case where an adversary cannot distinguish between any ι tags in the tag set $\mathcal{T}$ that have not been corrupted. This latter case may be considered as an application of the notion of ι-privacy (or ι-anonymity) [Samarati and Sweeney 1998] in the RFID system we defined.

3.5. Unp-Privacy

The goal of the adversary in the above ind-privacy game is to distinguish two different tags within its computational power and parameters. The idea is intuitively appealing; however, the ind-privacy model is difficult to apply *directly* in proving a given protocol is ind-private. To our knowledge, no mutual authentication RFID protocol has been proven *directly* to be ind-private. To address this concern, Ha et al. [2008] proposed a different privacy model based on the unpredictability of tag outputs, denoted as unp-privacy.

Figure 6 illustrates the unp-privacy experiment $\mathbf{Exp}_{\mathcal{A}}^{unp}[\kappa,\ \ell, q, s, u, v]$ ($\mathbf{Exp}_{\mathcal{A}}^{unp}$, for simplicity), in which an adversary is also comprised of a pair of algorithms $(\mathcal{A}_1, \mathcal{A}_2)$ and runs in two stages. In the *learning stage*, algorithm $\mathcal{A}_1$ is required to select only one challenge tag $\mathcal{T}_c$. It also outputs a state information st which will be transmitted to algorithm $\mathcal{A}_2$. Throughout the experiment, adversary $\mathcal{A}$ is allowed to launch O_1, O_2, O_3 and O_4 oracle queries without exceeding q, s, u and v overall calls respectively under the condition that $\mathcal{A}_1$ cannot query $\mathsf{SetTag}(\mathcal{T}_c)$. Then in the *guess stage*, algorithm $\mathcal{A}_2$ is required to infer whether the challenge message r^* is chosen from the output of running the protocol $\pi(R, \mathcal{T}_c)$. Note that, $\mathcal{A}_2$ is not allowed to query any oracle.

Definition 3.8. The advantage of adversary $\mathcal{A}$ in the experiment $\mathbf{Exp}_{\mathcal{A}}^{unp}$ is defined as:

$$\mathrm{Adv}_{\mathcal{A}}^{unp}(\kappa, \ell, q, s, u, v) = \left| \Pr\left[\mathbf{Exp}_{\mathcal{A}}^{unp}[\kappa, \ell, q, s, u, v] = 1\right] - \frac{1}{2} \right|,$$

Experiment $\mathbf{Exp}_{\mathcal{A}}^{unp'}[\kappa, \ell, q, s, u, v]$
1. setup the reader R and a set of tags $\mathcal{T}$ with $|\mathcal{T}| = \ell$;
2. $\{\mathcal{T}_c, c_0, st\} \leftarrow \mathcal{A}_1^{O_1, O_2, O_3, O_4}(R, \mathcal{T})$; *//learning stage*
3. set $\mathcal{T}' = \mathcal{T} - \{\mathcal{T}_c\}$;
4. $b \in_R \{0, 1\}$;
5. if $b = 0$ then $(r^*, f^*) \in_R P_{RS} \times P_{FR}$,
   else run the protocol with the challenge message c_0;
   get the transcripts of the protocol execution (c_0, r_0, f_0);
   set $(r^*, f^*) = (r_0, f_0)$;
6. $b' \leftarrow \mathcal{A}_2^{O_1, O_2, O_3, O_4}(R, \mathcal{T}', st, r^*, f^*)$; *//guess stage*
7. the experiment outputs 1 if $b' = b$, 0 otherwise.

Fig. 7. Unp′-privacy experiment.

where the probability is taken over the choice of tag set $\mathcal{T}$ and the coin tosses of the adversary $\mathcal{A}$.

Definition 3.9. An adversary $\mathcal{A}$ $(\epsilon, t, q, s, u, v)$-breaks the unp-privacy of RFID system RS if the advantage $\mathrm{Adv}_{\mathcal{A}}^{unp}(\kappa, \ell, q, s, u, v)$ of $\mathcal{A}$ in the experiment $\mathbf{Exp}_{\mathcal{A}}^{unp}$ is at least ϵ and the running time of $\mathcal{A}$ is at most t.

Definition 3.10 $(\epsilon, t, q, s, u, v)$*-Unp-Privacy.* An RFID system RS is said to be $(\epsilon, t, q, s, u, v)$-unp-private if there exists no adversary who can $(\epsilon, t, q, s, , u, v)$-break the unp-privacy of RS.

4. LIMITATIONS OF UNP-PRIVACY MODEL AND RECENT IMPROVEMENT

Note that in the unp-privacy game, the adversary $\mathcal{A}_2$ does not get the full transcript of the RFID protocol execution between the reader and the challenge tag, but only r^* which is either a random message or the message sent by the tag. As a result, an RFID protocol having known weakness in privacy (e.g., protocol messages are traceable by an adversary) can be shown to be unp-private, as confirmed by van Deursen and Radomirović [2009].

At CCS'09, Ma et al. [2009] proposed an improved unp-privacy model, which we denote as unp′-privacy for convenience. In the unp′-privacy model, the adversary is given not only r^*, but also the last message f^* of the protocol. The unp′-privacy model is robust for 2-round RFID protocols, as demonstrated in Ma et al. [2009]; however, we will show in this section that the model has a deficiency when applied to 3-round protocols. In such cases, an active attacker can trace a tag in an unp′-private RFID system.

4.1. Unp′-Privacy Model

Figure 7 illustrates the unp′-privacy experiment $\mathbf{Exp}_{\mathcal{A}}^{unp'}[\kappa,\ \ell, q, s, u, v]$ ($\mathbf{Exp}_{\mathcal{A}}^{unp'}$, for simplicity), in which an adversary is also comprised of a pair of algorithms $(\mathcal{A}_1, \mathcal{A}_2)$ and runs in two stages. In the *learning stage*, algorithm $\mathcal{A}_1$ is required to select only one challenge tag $\mathcal{T}_c$ and a *test* message $c_0 \in P_{CH}$. It also outputs a state information st which will be transmitted to algorithm $\mathcal{A}_2$. Throughout the experiment, adversary $\mathcal{A}$ is allowed to launch O_1, O_2, O_3 and O_4 oracle queries without exceeding q, s, u and v overall calls respectively under the condition that $\mathcal{A}_1$ cannot query $\mathsf{SetTag}(\mathcal{T}_c)$. Then in the *guess stage*, algorithm $\mathcal{A}_2$ has oracle accesses to tags except $\mathcal{T}_c$ and is required to

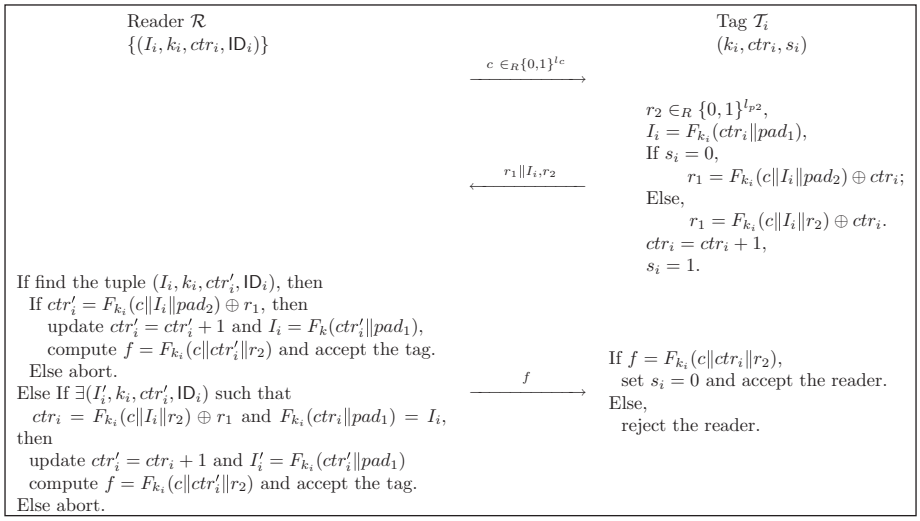

Fig. 8. Counterexample.

infer whether the challenge message pair (r^*, f^*) is chosen from the output of running the protocol $\pi(R, \mathcal{T}_c)$ with *test* message c_0.

Definition 4.1. The advantage of adversary $\mathcal{A}$ in the experiment $\mathbf{Exp}_{\mathcal{A}}^{unp'}$ is defined as:

$$\mathrm{Adv}_{\mathcal{A}}^{unp'}(\kappa, \ell, q, s, u, v) = \left| \Pr\left[\mathbf{Exp}_{\mathcal{A}}^{unp'}[\kappa, \ell, q, s, u, v] = 1\right] - \frac{1}{2} \right|,$$

where the probability is taken over the choice of tag set $\mathcal{T}$ and the coin tosses of the adversary $\mathcal{A}$.

Definition 4.2. An adversary $\mathcal{A}$ $(\epsilon, t, q, s, u, v)$-breaks the unp′-privacy of RFID system RS if the advantage $\mathrm{Adv}_{\mathcal{A}}^{unp'}(\kappa, \ell, q, s, u, v)$ of $\mathcal{A}$ in the experiment $\mathbf{Exp}_{\mathcal{A}}^{unp'}$ is at least ϵ and the running time of $\mathcal{A}$ is at most t.

Definition 4.3 $(\epsilon, t, q, s, u, v)$-*Unp′-Privacy.* An RFID system RS is said to be $(\epsilon, t, q, s, u, v)$-unp′-private if there exists no adversary who can $(\epsilon, t, q, s, , u, v)$-break the unp′-privacy of RS.

4.2. A Counterexample

Ma et al. [2009] introduced an efficient 2-round protocol and proved that it is unp′-private, where the adversary is provided with tag response r^* only in the guess stage. The unp′-privacy model is robust for 2-round RFID protocols, as demonstrated in Ma et al. [2009]; however, we show that the model has a deficiency when applied to 3-round protocols.

We modify the 2-round protocol of Ma et al. [2009] to a 3-round mutual authentication protocol as illustrated in Figure 8 and show that the new protocol has clear weakness with respect to privacy but can be proven to be unp′-private. This example therefore exposes a deficiency of the unp′-privacy model when it is applied to 3-round mutual authentication protocols.

Let $F : \{0,1\}^{l_k} \times \{0,1\}^{l_d} \to \{0,1\}^{l_r}$ be a PRF family. Let $ctr \in \{0,1\}^{l_r}$ be a counter. Let $pad_1 \in \{0,1\}^{l_{p1}}$ and $pad_2 \in \{0,1\}^{l_{p2}}$ be two paddings such that $l_r + l_{p1} = l_d$. The RFID system is constructed as follows.

ReaderSetup(κ). It sets up a reader R with $\sigma = \{F, pad_1, pad_2\}$ according to security parameter κ.

TagSetup$(\mathcal{T}_i, \kappa)$. It sets up a tag $\mathcal{T}_i$ with a key $k_i \in \{0,1\}^{l_k}$, a counter $ctr_i = 1$ and a 1-bit flag $s_i = 0$. It also stores a tuple $(I_i, k_i, ctr_i, \mathsf{ID}_i)$ in the reader's database, where $I_i = F_{k_i}(ctr_i \| pad_1)$ and ID_i is the tag's identity.

Protocol $\pi(R, \mathcal{T}_i)$. $\mathcal{R}$ first sends a challenge $c \in_R \{0,1\}^{l_c}$ to $\mathcal{T}_i$, where $l_c + l_r + l_{p2} = l_d$. Upon receiving c, $\mathcal{T}_i$ executes the following steps:

(1) Randomly choose $r_2 \in \{0,1\}^{l_{p2}}$ and compute $I_i = F_{k_i}(ctr_i \| pad_1)$;
(2) Set $r_1 = F_{k_i}(c \| I_i \| pad_2) \oplus ctr_i$ if $s_i = 0$, else set $r_1 = F_{k_i}(c \| I_i \| r_2) \oplus ctr_i$;
(3) Respond with $(r_1 \| I_i, r_2)$, increment ctr_i by 1 and set $s_i = 1$.

Upon receiving the response $(r_1 \| I_i, r_2)$, $\mathcal{R}$ identifies the tag from its database as follows.

(1) Search for the tuple $(I_i, k_i, ctr_i', \mathsf{ID}_i)$ using I_i as an index. If such a tuple exists, compute $F_{k_i}(c \| I_i \| pad_2)$ and then
   (a) If $ctr_i' = F_{k_i}(c \| I_i \| pad_2) \oplus r_1$, update $ctr_i' = ctr_i' + 1$ and $I_i = F_{k_i}(ctr_i' \| pad_1)$, respond with $f = F_{k_i}(c \| ctr_i' \| r_2)$ and accept the tag;
   (b) Else abort the protocol.
(2) Else look up the database for a tuple $(I_i', k_i, ctr_i', \mathsf{ID}_i)$ in an exhaustive search such that $ctr_i = F_{k_i}(c \| I_i \| r_2) \oplus r_1$ and $F_{k_i}(ctr_i \| pad_1) = I_i$. Then
   (a) If such a tuple exists, update $ctr_i' = ctr_i + 1$ and $I_i' = F_{k_i}(ctr_i' || pad_1)$, respond with $f = F_{k_i}(c \| ctr_i' \| r_2)$ and accept the tag;
   (b) Else abort the protocol.

Upon receiving f, $\mathcal{T}_i$ checks whether $f = F_{k_i}(c \| ctr_i \| r_2)$. If not, $\mathcal{T}_i$ rejects the reader. Else, $\mathcal{T}_i$ sets $s_i = 0$ and accepts the reader.

A flaw of the protocol is that an active attacker can find out whether a tag's state is $s = 0$ or $s = 1$. If a tag is in state $s = 0$, the reader does not verify the integrity of r_2; while if the tag is in state $s = 1$, this verification occurs implicitly. Note that under normal circumstances tags will be in state $s = 0$. Hence, an active attacker can flag a tag by setting its state to $s = 1$ and trace the tag in subsequent protocol sessions. However, the following theorem states that the protocol is unp′-private.

CLAIM 1. *The given mutual authentication RFID protocol is unp′-private, assuming the function family* $F : \{0,1\}^{l_k} \times \{0,1\}^{l_d} \to \{0,1\}^{l_r}$ *is a PRF family.*

PROOF. Assume that the protocol in Figure 8 is not unp′-private. That is, there exists an adversary $\mathcal{A}$ who can $(\epsilon, t, q, s, u, v)$-break the unp′-privacy of the protocol. We construct an algorithm $\mathcal{B}$ that can pass the *PTPT* for the function family F.

On the input of an oracle O_f for a function $f = F_k$ or $f \in_R \mathsf{Rand}^{\mathcal{D}\to\mathcal{R}}$, algorithm $\mathcal{B}$ invokes $\mathcal{A}$ and simulates the unp′-privacy experiment for $\mathcal{A}$ as follows.

Simulate the learning stage. Initially, $\mathcal{B}$ selects an index i between 1 and ℓ randomly and sets the initial state of the tag $\mathcal{T}_i$ as $ctr_i = 1, s_i = 0$. The key of $\mathcal{T}_i$ is implicitly set to be k, which is unknown to $\mathcal{B}$. For $1 \le j \le \ell$ and $j \ne i$, $\mathcal{B}$ selects a random key (index) $k_j \in_R \{0,1\}^{\kappa_1}$, then sets the key and the internal state of the tag $\mathcal{T}_j$ as k_j and $ctr_j = 1, s_j = 0$, respectively.

When adversary $\mathcal{A}$ asks queries about O_1, O_2, O_3 and O_4, algorithm $\mathcal{B}$ uses O_f and the keys $k_1, \ldots, k_{i-1}, k_{i+1}, \ldots, k_\ell$ to respond. Note that, when $\mathcal{A}$ issues O_3 query on tag $\mathcal{T}_i$, $\mathcal{B}$ aborts and randomly outputs a bit.

Experiment $\mathbf{Exp}_{\mathcal{A}}^{unp^*}[\kappa, \ell, q, s, u, v]$
1. setup the reader R and a set of tags $\mathcal{T}$ with $|\mathcal{T}| = \ell$;
2. $\{\mathcal{T}_c, st\} \leftarrow \mathcal{A}_1^{O_1,O_2,O_3,O_4}(R^{\varsigma}\mathcal{T})$; //*learning stage*
3. $b \in_R \{0,1\}$;
4. $b' \leftarrow \mathcal{A}_2^{O_1,O_2,O_4}(R, \mathcal{T}_c, st)$; //*guess stage*
   4.1 when $\mathcal{A}_2$ queries O_1, O_2 and O_4 oracles, if $b = 1$, run the algorithm ReaderStart, TagCompute, ReaderCompute respectively, and return the results;
   4.2 else $b = 0$, return a random element from P_{CH}, P_{FT}, P_{FR}, respectively.
5. the experiment outputs 1 if $b' = b$, 0 otherwise.

Fig. 9. Unp*-privacy experiment.

Simulate the challenge stage. $\mathcal{A}$ submits a test message $c_0 \in \{0,1\}^{l_c}$ and an uncorrupted challenge tag $\mathcal{T}_c$. If $\mathcal{T}_c \neq \mathcal{T}_i$, $\mathcal{B}$ aborts and randomly outputs a bit. Otherwise, $\mathcal{B}$ proceeds as follows.

(1) Randomly choose $r_2 \in \{0,1\}^{l_{p2}}$.
(2) Set $x = ctr_i \| pad_1$, query O_f on x and get the response I_i.
(3) If $s_i = 0$, query O_f on $x = c_0 \| I_i \| pad_2$, get the response y and set $r_1 = y \oplus ctr_i$; else query its oracle on $x = c_0 \| I_i \| r_2$, get the response y and set $r_1 = y \oplus ctr_i$.
(4) Set $r^* = (r_1 \| I_i, r_2)$ and update $ctr_i = ctr_i + 1$ and $s_i = 0$.
(5) Query O_f on $c_0 \| ctr_i \| r_2$, get the response f^*, and send (r^*, f^*) to $\mathcal{A}$.

Simulate the guess stage. Algorithm $\mathcal{B}$ answers adversary $\mathcal{A}$'s queries about O_1, O_2, O_3, O_4 using O_f and the keys $k_1, \ldots, k_{i-1}, k_{i+1}, \ldots, k_\ell$.

Output. Finally, adversary $\mathcal{A}$ outputs a bit b'. $\mathcal{B}$ also takes b' as its output.

If $\mathcal{B}$ does not abort during the simulation, $\mathcal{B}$'s simulation is perfect and is identically distributed as the real one from the construction. It is obvious that the probability that $\mathcal{B}$ does not abort during the simulation is $\frac{1}{\ell}$. Therefore, the advantage of $\mathcal{B}$ is at least $\frac{\epsilon}{\ell}$.

The running time of $\mathcal{B}$ is approximate to that of $\mathcal{A}$. This completes the proof. □

5. A NEW PRIVACY MODEL, UNP*-PRIVACY

The limitation in the definition of unp′-privacy, as shown in the counterexample, is due to the constraint imposed on the adversary $\mathcal{A}_2$, that is, $\mathcal{A}_2$ only has access to r^* and f^* as supplied by the challenger and is not allowed to query oracles on the challenge tag $\mathcal{T}_c$. In this section, we propose a new RFID privacy model, denoted as unp*-privacy, as a remedy to this problem.

The intuition of the unp*-privacy model is that no adversary should be able to distinguish the output of a real tag from that of a virtual tag, given transcripts of multiple procotol sessions of both, where a virtual tag is defined as a tag without any secret information. This implies that no adversary can link a real tag and its behavior without learning its secret key. We emphasize that our unp*-privacy model does not impose any restrictions on the number of oracle queries issued by the adversary on the challenge tag. In what follows, we introduce the unp*-privacy model, investigate the relationship between this new model and the ind-privacy model.

5.1. Unp*-Privacy Model

Figure 9 illustrates the unp*-privacy experiment $\mathbf{Exp}_{\mathcal{A}}^{unp^*}[\kappa, \ell, q, s, u, v]$ ($\mathbf{Exp}_{\mathcal{A}}^{unp^*}$, for simplicity), in which an adversary is also comprised of a pair of algorithms $(\mathcal{A}_1, \mathcal{A}_2)$ and runs in two stages. Throughout the experiment, adversary $\mathcal{A}$ is allowed to launch O_1, O_2, O_3, and O_4 oracle queries without exceeding q, s, u, and v overall calls

respectively. In the *learning stage*, algorithm $\mathcal{A}_1$ issues O_1, O_2, O_3, and O_4 oracle queries, and output an uncorrupted challenge tag $\mathcal{T}_c$. It also outputs a state information st which will be transmitted to algorithm $\mathcal{A}_2$. Next, the experiment selects a random bit b. Algorithm $\mathcal{A}_2$ is allowed to query O_1, O_2, and O_4 oracles on R and $\mathcal{T}_c$. The experiment responds to $\mathcal{A}_2$ queries as follows:

InitReader. If $b = 0$, generate a new session identifier sid, choose $c_{sid} \in_R P_{CH}$ and forward (sid, c_{sid}) to $\mathcal{A}_2$; else, run the algorithm ReaderStart, and forward the result to $\mathcal{A}_2$.

InitTag$(\mathcal{T}_c, sid, c_{sid})$. If $b = 0$, choose $r_{sid} \in P_{FT}$ and forward r_{sid} to $\mathcal{A}_2$; else, run the algorithm TagCompute$(\mathcal{T}_c, sid, c_{sid})$ and forward the result to $\mathcal{A}_2$.

SendRes(sid, c_{sid}, r_{sid}). If $b = 0$, choose $f_{sid} \in P_{FR}$ and forward f_{sid} to $\mathcal{A}_2$; else, run the algorithm ReaderCompute(sid, c_{sid}, r_{sid}) and forward the result to $\mathcal{A}_2$.

Finally, $\mathcal{A}_2$ is asked to guess the value of the random bit.

Definition 5.1. The advantage of adversary $\mathcal{A}$ in the experiment $\mathbf{Exp}_{\mathcal{A}}^{unp^\star}$ is defined as:

$$\mathrm{Adv}_{\mathcal{A}}^{unp^\star}(\kappa, \ell, q, s, u, v) = \left| \Pr\left[\mathbf{Exp}_{\mathcal{A}}^{unp^\star}[\kappa, \ell, q, s, u, v] = 1\right] - \frac{1}{2} \right|,$$

where the probability is taken over the choice of tag set $\mathcal{T}$ and the coin tosses of the adversary $\mathcal{A}$.

Definition 5.2. An adversary $\mathcal{A}$ $(\epsilon, t, q, s, u, v)$-breaks the unp*-privacy of RFID system RS if the advantage $\mathrm{Adv}_{\mathcal{A}}^{unp^\star}(\kappa, \ell, q, s, u, v)$ of $\mathcal{A}$ in the experiment $\mathbf{Exp}_{\mathcal{A}}^{unp^\star}$ is at least ϵ and the running time of $\mathcal{A}$ is at most t.

Definition 5.3. $(\epsilon, t, q, s, u, v)$-*Unp*-Privacy*. An RFID system RS is said to be $(\epsilon, t, q, s, u, v)$-unp*-private if there exists no adversary who can $(\epsilon, t, q, s, , u, v)$-break the unp*-privacy of RS.

Note that the protocol given in Figure 8 does not satisfy our privacy model. In the unp*-privacy experiment, if $b = 0$, the adversary modifies the second message randomly; with overwhelming probability, the third message of the protocol is empty. However, if $b = 1$, the third message is always a random value and not empty. So, with overwhelming probability, the adversary can distinguish the two cases. Hence, the protocol in Figure 8 is not unp*-private.

5.2. Relation between Unp*-Privacy and Ind-Privacy

We investigate the relation between ind-privacy and unp*-privacy. For this purpose, we introduce a restricted ind-privacy model, called ind$'$-privacy, as a "bridge" to show that it is equivalent to ind-privacy and that unp*-privacy implies ind$'$-privacy. Then, we show that ind-privacy does not imply unp*-privacy by constructing an RFID system which is ind-private but not unp*-private. Overall, we prove that unp*-privacy is stronger than ind-privacy.

While the notion of ind-privacy may suffice, it is more difficult to use in proving security for an RFID system than the notion of unp*-privacy. In fact, most of ind-privacy proofs known so far, including the proof in the original paper [Juels and Weis 2007], use unp*-privacy as a bridge, showing that every PPT adversary is not able to distinguish the transcripts of protocol execution from random values. One technical challenge of applying ind-privacy is how to transfer the ability to distinguish between two tags to the ability to break a cryptographic primitive or to solve a hard problem. Another difference is that unp*-privacy restricts the adversary from making queries

Experiment $\mathbf{Exp}_{\mathcal{A}}^{ind'}[\kappa, \ell, q, s, u, v, w]$
1. setup the reader R and a set of tags $\mathcal{T}$ with $|\mathcal{T}| = \ell$;
2. $\{\mathcal{T}_i, \mathcal{T}_j, st\} \leftarrow \mathcal{A}_1^{O_1,O_2,O_3,O_4}(R, \mathcal{T})$; //*learning stage*
3. $b \in_R \{0, 1\}$;
4. if $b = 0$ then $\mathcal{T}_c = \mathcal{T}_i$, else $\mathcal{T}_c = \mathcal{T}_j$;
5. $b' \leftarrow \mathcal{A}_2^{O_1,O_2,O_4}(R, \mathcal{T}_c, st)$; //*guess stage*
6. the experiment outputs 1 if $b' = b$, 0 otherwise.

Fig. 10. Ind′-privacy experiment.

on the set $\mathcal{T}'$ in the second stage of the game. Such restriction can make the security proof more concise. Intuitively, ind-privacy implies that a tag cannot be distinguished from a group of tags in an RFID system, which may demonstrate a unique pattern as a group (e.g., with common prefix in protocol transcripts), while unp*-privacy requires that a tag's interaction with a reader does not have any such group patterns.

5.2.1. Ind′-Privacy. Figure 10 shows the ind′-privacy experiment $\mathbf{Exp}_{\mathcal{A}}^{ind'}[\kappa, \ell, q, s, u, v]$ ($\mathbf{Exp}_{\mathcal{A}}^{ind'}$, for simplicity), which is identical to the ind-privacy experiment given in Figure 5 except that $\mathcal{A}_2$ in $\mathbf{Exp}_{\mathcal{A}}^{ind'}$ is not allowed to query oracles on other tags except for $\mathcal{T}_c$.

Definition 4.1. The advantage of adversary $\mathcal{A}$ in the ind′-privacy experiment $\mathbf{Exp}_{\mathcal{A}}^{ind'}$ is defined as:

$$\mathrm{Adv}_{\mathcal{A}}^{ind'}(\kappa, \ell, q, s, u, v) = \left|\Pr\left[\mathbf{Exp}_{\mathcal{A}}^{ind'} = 1\right] - \frac{1}{2}\right|,$$

where the probability is taken over the choice of tag set $\mathcal{T}$ and the coin tosses of the adversary $\mathcal{A}$.

Definition 4.2. An adversary $\mathcal{A}$ $(\epsilon, t, q, s, u, v)$-breaks the ind′-privacy of RFID system RS if its advantage $\mathrm{Adv}_{\mathcal{A}}^{ind'}(\kappa, \ell, q, s, u, v)$ in the experiment $\mathbf{Exp}_{\mathcal{A}}^{ind'}$ is at least ϵ and its running time is at most t.

Definition 4.3 $(\epsilon, t, q, s, u, v)$-*Ind′-Privacy.* An RFID system RS is said to be $(\epsilon, t, q, s, u, v)$-ind′-private if there exists no adversary $\mathcal{A}$ who can $(\epsilon, t, q, s, , u, v)$-break the ind′-privacy of RS.

5.2.2. Ind-Privacy $\Longleftrightarrow$ Ind′-Privacy. The only difference between ind-privacy and ind′-privacy is that, in ind-privacy the adversary can issue oracle queries on any tag in $\mathcal{T}' \cap \{\mathcal{T}_c\}$ in the guess stage, while in ind′-privacy the adversary can only issue oracle queries on $\mathcal{T}_c$ in the guess stage. In other words, ind′-privacy puts more restrictions on the adversary, so ind-privacy implies ind′-privacy. However, in ind′-privacy, the adversary can issue O_3 queries on all tags in $\mathcal{T}'$ in the learning stage, obtain the secrets and states of all tags in $\mathcal{T}'$ and store them in a list TagKey-List. In the guess stage, when the adversary wants to make O_1, O_2, O_3, O_4 queries on any tag in $\mathcal{T}'$, the adversary can obtain the corresponding query answers itself using the list TagKey-List. As a result, the restriction on the adversary does not weaken its power in the ind′-privacy model, and ind′-privacy implies ind-privacy.

THEOREM 1. *For an RFID system* $RS = (R, \mathcal{T}, \mathsf{ReaderSetup}, \mathsf{TagSetup}, \pi)$*, ind-privacy is equivalent to ind′-privacy.*

PROOF. It is obvious that ind-privacy $\Longrightarrow$ ind′-privacy holds.

Now we formally prove that ind-privacy $\Longleftarrow$ ind′-privacy holds. Assume that RS is not ind-private. That is, there exists an adversary $\mathcal{A}$ such that it $(\epsilon, t, q, s, u_1, v)$-breaks the ind-privacy of RS. We construct an algorithm $\mathcal{B}$ that uses $\mathcal{A}$ as a subroutine and $(\epsilon, t, q, s, u_2, v)$-breaks the ind′-privacy of RS, where $u_2 \leq u_1 + \ell - 2$. The algorithm $\mathcal{B}$ proceeds as follows. On the input of the RFID system RS and the security parameter κ, it invokes adversary $\mathcal{A}$ with input RS and κ and conducts the ind-privacy experiment with $\mathcal{A}$ as follows.

Simulate the learning stage. When adversary $\mathcal{A}$ asks queries about O_1, O_2, O_3 and O_4, algorithm $\mathcal{B}$ also queries them in the ind′-privacy experiment $\mathbf{Exp}_{\mathcal{B}}^{ind'}$ and returns the responses to adversary $\mathcal{A}$ accordingly.

Simulate the challenge stage. When adversary $\mathcal{A}$ outputs two uncorrupted tags $\mathcal{T}_i, \mathcal{T}_j \in \mathcal{T}$, algorithm $\mathcal{B}$ submits the same two tags $\mathcal{T}_i$ and $\mathcal{T}_j$ in the ind′-privacy experiment $\mathbf{Exp}_{\mathcal{B}}^{ind'}$ which responds with a challenge tag $\mathcal{T}_c \in \{\mathcal{T}_i, \mathcal{T}_j\}$. Then, $\mathcal{B}$ *issues* O_3 *queries on the tag set* $\mathcal{T} - \{\mathcal{T}_i, \mathcal{T}_j\}$ *and stores the results in a list* TagKey-List. Finally, $\mathcal{B}$ forwards $\mathcal{T}_c$ to $\mathcal{A}$.

Simulate the guess stage. When adversary $\mathcal{A}$ asks queries about O_1, O_2, O_3 and O_4, algorithm $\mathcal{B}$ uses its oracles O_1, O_2, O_4 and the list TagKey-List to respond.

Output. If $\mathcal{A}$ outputs a bit b', then $\mathcal{B}$ also takes it as its output.

It is obvious that the simulation is perfect. Thus we have shown an adversary $\mathcal{A}$ against the ind-privacy of the RFID system RS with advantage ϵ can be used to construct another adversary $\mathcal{B}$ against the ind′-privacy of the same RFID system with an identical advantage.

The number of times that $\mathcal{B}$ queries the O_3 oracle is at most $u_1 + \ell - 2$. The running time of $\mathcal{B}$ is approximate to that of $\mathcal{A}$. This completes the proof. □

5.2.3. Unp-Privacy $\Longrightarrow$ Ind′-Privacy.* Recall that unp⋆-privacy means that every PPT adversary is not able to distinguish the transcripts of the protocol execution between the reader and a real tag from those of protocol execution between the reader and a virtual tag, which are random values. The underlying intuition of ind′-privacy is that every PPT adversary cannot distinguish the transcripts of the protocol execution between the reader and two distinct tags. It is obvious that a PPT adversary cannot distinguish between one random value and another random value. So, if the transcripts of the protocol execution between the reader and each tag looks random, the adversary cannot distinguish the transcripts of the protocol execution between the reader and two tags. In other words, unp⋆-privacy implies ind′-privacy.

THEOREM 2. *Assume that the RFID system* $RS = (R, \mathcal{T}, \mathsf{ReaderSetup}, \mathsf{TagSetup}, \pi)$ *is* $(\epsilon, t, q, s, u, v)$*-unp*-private, then it is* $(\epsilon, t, q, s, u, v)$*-ind′-private.*

PROOF. Assume that RS is not ind′-private. That is, there exists an adversary $\mathcal{A}$ which can $(\epsilon, t, q, s, u, v)$-break the ind′-privacy of RS. Then, we construct an algorithm $\mathcal{B}$ that runs $\mathcal{A}$ as a subroutine and $(\epsilon, t, q, s, u, v)$-breaks the unp*-privacy of RS.

Given an RFID system RS and the security parameter κ, algorithm $\mathcal{B}$ invokes $\mathcal{A}$ with the same input and simulates the ind′-privacy experiment for $\mathcal{A}$ as follows.

Simulate the learning stage. Algorithm $\mathcal{B}$ answers adversary $\mathcal{A}$'s queries about O_1, O_2, O_3, O_4 by asking them in the unp*-privacy experiment.

Simulate the challenge stage. When adversary $\mathcal{A}$ outputs two uncorrupted tags $\mathcal{T}_i, \mathcal{T}_j \in \mathcal{T}$, algorithm $\mathcal{B}$ selects a random bit $b \in \{0, 1\}$ and sets the challenge tag $\mathcal{T}_c = \mathcal{T}_i$ if $b = 0$ and $\mathcal{T}_c = \mathcal{T}_j$ otherwise. Finally, $\mathcal{B}$ forwards $\mathcal{T}_c$ to $\mathcal{A}$, and also submits $\mathcal{T}_c$ in the unp*-privacy experiment as its own challenge tag.

Simulate the guess stage. Algorithm $\mathcal{B}$ answers adversary $\mathcal{A}$'s queries about O_1, O_2, O_4 by asking them in the unp*-privacy experiment.

Output. Finally, adversary $\mathcal{A}$ outputs a bit b'. If $b' = b$, algorithm $\mathcal{B}$ outputs 1, otherwise it outputs 0.

The simulation of $\mathcal{B}$ is perfect. When the internal random bit selected by the unp*-privacy experiment is equal to 1, the probability of $b' = b$ is equal to $\frac{1}{2} \pm \epsilon$; otherwise, the probability of $b' = b$ is equal to $\frac{1}{2}$, because in this case the challenge tag $\mathcal{T}_c$ is in fact a virtual tag in adversary $\mathcal{A}$'s view. Hence, the advantage of $\mathcal{B}$ is equal to that of $\mathcal{A}$ (i.e., ϵ).

The running time of $\mathcal{B}$ is exactly the same as that of $\mathcal{A}$. This completes the proof. □

5.2.4. Unp-Privacy $\Longrightarrow$ Ind-Privacy.* From Theorem 1 and Theorem 2, one can derive the following:

THEOREM 3. *Assume that the RFID system RS is unp*-private, then it is ind-private.*

5.2.5. Ind-Privacy $\not\Longrightarrow$ Unp-privacy.* An ind-private RFID protocol implies that the distributions of protocol transcripts between reader and any two tags are computationally indistinguishable. Note that, the distribution could be any distribution, not necessarily random distribution. A unp*-privacy RFID protocol requires that the distribution of the protocol transcripts is random. Hence, ind-privacy does not imply unp*-privacy.

Let $RS = \{R, \mathcal{T}, \mathsf{ReaderSetup}, \mathsf{TagSetup}, \pi\}$ be any RFID system. We construct a new RFID system $RS' = \{R, \mathcal{T}, \mathsf{ReaderSetup}, \mathsf{TagSetup}, \pi'\}$ such that for every protocol message $(c, r, f) \leftarrow \pi(R, T_i)$, we have $(c, r||r, f) \leftarrow \pi'(R, T_i)$. Then, we have the following:

THEOREM 4. *If the RFID system RS is ind-private, then the RFID system RS' is also ind-private, but not unp*-private.*

PROOF. It is easy to see that RS' is ind-private if RS is ind-private. We proceed to show that it is not unp*-private. Since every protocol message of π' is in the form $(c, r||r, f) \in P_{CH} \times P_{RS}^2 \times P_{FR}$, the adversary can easily distinguish it from a random tuple $(c', r_1||r_2, f')$ chosen from $P_{CH} \times P_{RS}^2 \times P_{FR}$ by checking whether $r_1 = r_2$. Therefore, RS' is not unp*-private. □

This theorem indicates that ind-privacy does not imply unp*-privacy. In the practical sense, ind-privacy does not necessarily mean that an adversary cannot distinguish a tag (or a group of tags) in an RFID system from a tag (or a group of tags) in another RFID system, while unp*-privacy does if the protocol messages have the same length.

6. UNP*-PRIVACY $\Longleftrightarrow$ PRF

In this section, we investigate the minimal requirement for RFID systems to achieve unp*-privacy. Since an RFID reader is usually equipped with enough computational power, we assume that the reader is not resource-limited and focus on the minimal requirement for RFID tags only. We show that the necessary and sufficient condition for enforcing unp*-privacy in an RFID system is to equip every tag with the power of computing a PRF. Our result provides a theoretical foundation to explain why so many lightweight RFID protocols suffer from privacy vulnerabilities without implementing necessary cryptographic primitives.

6.1. Unp*-Privacy $\Longrightarrow$ PRF

Given an RFID system RS with unp*-privacy, we show that each tag's computation function $F_{T_i}()$ can be used to construct a PRF family.

Without loss of generality, let $P_{CH} = \{0,1\}^{\alpha}$, $P_K = \{0,1\}^{\alpha_1}$, $P_{CN} = \{0,1\}^{\alpha_2}$, $P_S = \{0,1\}^{\alpha_3}$, and $P_{FT} = \{0,1\}^{\alpha_2+\alpha_3}$, where $\alpha, \alpha_1, \alpha_2$ and α_3 are four polynomials of κ. For an index $\lambda \in P_K \times P_{CN} \times P_S$, assume that λ can be uniquely represented by $\lambda_k \| \lambda_{cn} \| \lambda_s$ (i.e. $|\lambda_k| = \alpha_1$, $|\lambda_{cn}| = \alpha_2$ and $|\lambda_s| = \alpha_3$), where $\lambda_k \in P_K$, $\lambda_{cn} \in P_{CH}$ and $\lambda_s \in P_S$.

Given an RFID system $RS = (R, \mathcal{T}, \mathsf{ReaderSetup}, \mathsf{TagSetup}, \pi)$, choose a tag $\mathcal{T}_i \in_R \mathcal{T}$ and define a function family $G : \mathcal{K} \times \mathcal{D} \longrightarrow \mathcal{R}$ as

$$G_\lambda(x) = F_{\mathcal{T}_i}(\lambda_k, \lambda'_{cn}, \lambda'_s, x),$$

where

$$\begin{aligned} \lambda \in \mathcal{K} = P_K \times P_{CN} \times P_S, \mathcal{D} = P_{CH} \text{ and } \mathcal{R} = P_{FT}, \\ \lambda = \lambda_k \| \lambda_{cn} \| \lambda_s, \\ \lambda'_{cn} \| \lambda'_s = F_{\mathcal{T}_i}(\lambda_k, \lambda_{cn}, \lambda_s, x). \end{aligned}$$

We proceed to prove that the function family $G : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$ is a PRF family.

THEOREM 5. *If the RFID system $RS = (R, \mathcal{T}, \mathsf{ReaderSetup}, \mathsf{TagSetup}, \pi)$ is complete, sound, and unp*-private, then the constructed function family $G : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$ is a PRF family.*

PROOF. Since the tag has only limited memory to store tag key and/or state information and since the RFID system RS is complete and sound, the function $F_{\mathcal{T}_i}()$ cannot be an empty function (i.e. $r \neq \varepsilon$) and its output cannot be independent of the challenge messages, or else, one can break the soundness of RS by simply replaying the outputs of tag $\mathcal{T}_i$. Moreover, the function $G_\lambda(x)$ defined above is polynomial-time computable since the protocol $\pi(R, \mathcal{T}_i)$ can be run in polynomial time. Furthermore, it is easy to index a function of family G by uniformly choosing an index from $\mathcal{K}$. Next, we show that the function family G is pseudorandom.

Assume that the function family G is not pseudorandom. That is, there exists an algorithm T which passes the *PTPT* for G with non-negligible advantage. We construct an algorithm $\mathcal{B}$ which runs T as a subroutine and breaks the unp*-privacy of RS with non-negligible advantage.

Algorithm $\mathcal{B}$ proceeds as follows. It first selects a tag $\mathcal{T}_i$ randomly from $\mathcal{T}$ and sets $\mathcal{T}_i$ as the challenge tag for the unp*-privacy experiment. Then, algorithm $\mathcal{B}$ invokes algorithm T. When algorithm T asks queries about O_f, $\mathcal{B}$ uses its O_2 (i.e., $\mathsf{InitTag}$) oracle to respond. When algorithm T outputs a bit b, algorithm $\mathcal{B}$ also outputs the bit b.

Now, we calculate the advantage of $\mathcal{B}$ in the unp*-privacy experiment, which provides a perfect simulation for T. The probability that $\mathcal{B}$ makes a correct guess of the coin toss of the unp*-privacy experiment is no less than the success probability of T. Hence, the advantage of $\mathcal{B}$ is non-negligible. Furthermore, it is obvious that the running time of algorithm $\mathcal{B}$ is the same as that of T. This completes the proof. □

6.2. Unp*-Privacy ⟸ PRF

Now, we construct an RFID system with unp*-privacy by implementing a PRF on each tag. Let $F : \{0,1\}^{l_k} \times \{0,1\}^{l_d} \to \{0,1\}^{l_r}$ be a PRF family, $ctr \in \{0,1\}^{l_r}$ be a counter, and $pad_1 \in \{0,1\}^{l_{p1}}$ be a padding such that $l_r + l_{p1} = l_d$. The RFID system is constructed as follows and the protocol is illustrated in Figure 11.

ReaderSetup(κ). It sets up a reader R with $\sigma = \{F, pad_1, pad_2\}$ according to security parameter κ.

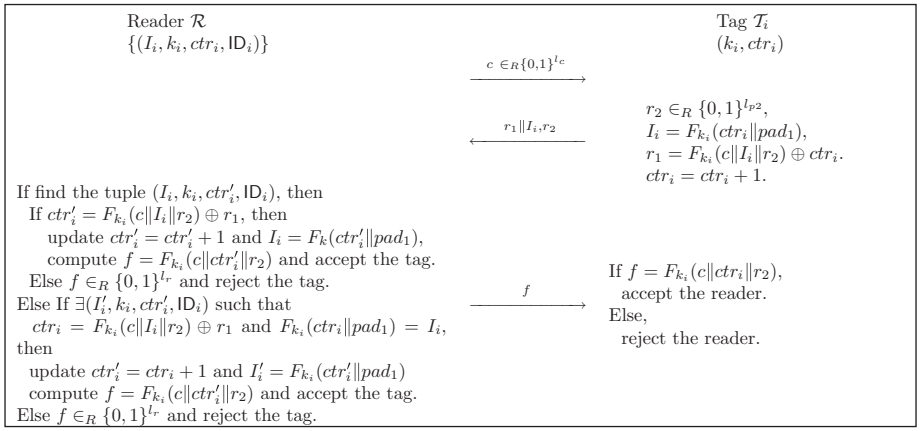

Fig. 11. Mutual authentication protocol with Unp*-privacy.

TagSetup$(\mathcal{T}_i, \kappa)$. It sets up a tag $\mathcal{T}_i$ with a key $k_i \in \{0,1\}^{l_k}$ and a counter $ctr_i = 1$. It also stores a tuple $(I_i, k_i, ctr_i, \mathsf{ID}_i)$ in the reader's database, where $I_i = F_{k_i}(ctr_i \| pad_1)$ and ID_i is the tag's identity.

Protocol $\pi(R, \mathcal{T}_i)$. $\mathcal{R}$ sends a challenge $c \in_R \{0,1\}^{l_c}$ to $\mathcal{T}_i$. Upon receiving c, $\mathcal{T}_i$ executes the following steps:

(1) Randomly choose $r_2 \in \{0,1\}^{l_{p2}}$, where $l_c + l_r + l_{p2} = l_d$.
(2) Compute $I_i = F_{k_i}(ctr_i \| pad_1)$ and $r_1 = F_{k_i}(c \| I_i \| r_2) \oplus ctr_i$.
(3) Respond with $(r_1 \| I_i, r_2)$ and increment ctr_i by 1.

Upon receiving the response $(r_1 \| I_i, r_2)$, $\mathcal{R}$ identifies the tag from its database as follows:

(1) Search for the tuple $(I_i, k_i, ctr'_i, \mathsf{ID}_i)$ using I_i as an index. If such a tuple exists, compute $F_{k_i}(c \| I_i \| r_2)$ and then
   (a) If $ctr'_i = F_{k_i}(c \| I_i \| r_2) \oplus r_1$, update $ctr'_i = ctr'_i + 1$ and $I_i = F_{k_i}(ctr'_i \| pad_1)$, respond with $f = F_{k_i}(c \| ctr'_i \| r_2)$ and accept the tag.
   (b) Else, respond with $f \in_R \{0,1\}^{l_r}$ and reject the tag.
(2) Else look up the database for a tuple $(I'_i, k_i, ctr'_i, \mathsf{ID}_i)$ in an exhaustive search such that $ctr_i = F_{k_i}(c \| I_i \| r_2) \oplus r_1$ and $F_{k_i}(ctr_i \| pad_1) = I_i$. Then
   (a) If such a tuple exists, update $ctr'_i = ctr_i + 1$ and $I'_i = F_{k_i}(ctr'_i || pad_1)$, respond with $f = F_{k_i}(c \| ctr'_i \| r_2)$ and accept the tag.
   (b) Else, respond with $f \in_R \{0,1\}^{l_r}$ and reject the tag.

Upon receiving f, $\mathcal{T}_i$ checks whether $f = F_{k_i}(c \| ctr_i \| r_2)$. If not, $\mathcal{T}_i$ rejects the reader; otherwise, accepts.

Next, we prove that the constructed RFID system is unp*-private.

THEOREM 6. *If the function family $F : \{0,1\}^{l_k} \times \{0,1\}^{l_d} \to \{0,1\}^{l_r}$ is a PRF family, then the RFID system $RS = (R, \mathcal{T}, \textit{ReaderSetup}, \textit{TagSetup}, \pi)$ defined above is unp*-private.*

PROOF. Assume that RS is not unp*-private. That is, there exists an adversary $\mathcal{A}$ which can $(\epsilon, t, q, s, u, v)$-break the unp*-privacy of RS. We construct an algorithm $\mathcal{B}$ that can pass the *PTPT* for the function family F.

On the input of an oracle O_f for a function $f = F_k$ or $f \in_R \mathsf{Rand}^{\mathcal{D}\to\mathcal{R}}$, algorithm $\mathcal{B}$ invokes $\mathcal{A}$ and simulates the unp*-privacy experiment for $\mathcal{A}$ as follows:

Simulate the learning stage. Initially, $\mathcal{B}$ selects an index i between 1 and ℓ randomly and sets the initial state of the tag $\mathcal{T}_i$ as $ctr_i = 1$. The key of $\mathcal{T}_i$ is implicitly set to be k, which is unknown to $\mathcal{B}$. For $1 \le j \le \ell$ and $j \ne i$, $\mathcal{B}$ selects a random key (index) $k_j \in_R \{0,1\}^{\kappa_1}$, then sets the key and the internal state of the tag $\mathcal{T}_j$ as k_j and $ctr_j = 1$, respectively.

When adversary $\mathcal{A}$ asks queries about O_1, O_2, O_3 and O_4, algorithm $\mathcal{B}$ uses O_f and the keys $k_1, \ldots, k_{i-1}, k_{i+1}, \ldots, k_\ell$ to respond. Note that, when $\mathcal{A}$ issues O_3 query on tag $\mathcal{T}_i$, $\mathcal{B}$ aborts and randomly outputs a bit.

Simulate the challenge stage. $\mathcal{A}$ submits an uncorrupted challenge tag $\mathcal{T}_c$. If $\mathcal{T}_c \ne \mathcal{T}_i$, $\mathcal{B}$ aborts and randomly outputs a bit.

Simulate the guess stage. Algorithm $\mathcal{B}$ answers adversary $\mathcal{A}$'s queries about O_1, O_2, O_4 using O_f and the keys $k_1, \ldots, k_{i-1}, k_{i+1}, \ldots, k_\ell$.

Output. Finally, adversary $\mathcal{A}$ outputs a bit b'. $\mathcal{B}$ also takes b' as its output.

If $\mathcal{B}$ does not abort during the simulation, $\mathcal{B}$'s simulation is perfect and is identically distributed as the real one from the construction. It is obvious that the probability that $\mathcal{B}$ does not abort during the simulation is $\frac{1}{\ell}$. Therefore, the advantage of $\mathcal{B}$ is at least $\frac{\epsilon}{\ell}$.

The running time of $\mathcal{B}$ is approximate to that of $\mathcal{A}$. This completes the proof. □

6.3. Minimal Requirement on RFID Tags for Unp*-Privacy

Combining Theorems 5 and 6, one can derive the following:

THEOREM 7. *An RFID system* $RS = (R, \mathcal{T}, \mathsf{ReaderSetup}, \mathsf{TagSetup}, \pi)$ *with unp*-privacy can be constructed if and only if each tag* $\mathcal{T}_i \in \mathcal{T}$ *is empowered to compute a PRF, provided that RS is complete and sound.*

This theorem indicates that to ensure unp*-privacy, the computational power of tags cannot be weaker than that of computing a PRF. In other words, the minimal requirement on tags to achieve unp*-privacy for RFID systems is the ability to compute a PRF or its equivalents such as symmetric block ciphers and cryptographic hash functions [Goldreich et al. 1986].

This minimal requirement highlights why many lightweight RFID protocols [Karthikeyan and Nesterenko 2005; Duc et al. 2006; Chien and Chen 2007; Konidala et al. 2007] have privacy flaws [Peris-Lopez et al. 2008; van Deursen and Radomirovic 2008], as these protocols are constructed based on simple operations such as XOR, bit inner product, 16-bit pseudo-random number generator (PRNG), and cyclic redundancy checksum (CRC) without using any computation equivalent to PRF.

The RFID research community has in recent years realized the importance of implementing strong and yet lightweight cryptographic primitives for low-cost RFID tags [Eisenbarth et al. 2007] and significant progress has been made in this area. For instance, the efficient hardware implementation for the Advanced Encryption Standard (AES) has 3,400 gate equivalents (GEs) [Feldhofer et al. 2005]. A specially designed block cipher, PRESENT, can further reduce the hardware requirement to as few as 1,570 GEs with reasonable security (80 bits) and performance [Bogdanov et al. 2007]. For asymmetric cryptography, a minimum 113-bit ECC can be realized in hardware with a much larger chip area (at least 10,000 GEs) [Kumar and Paar 2006].

We stress that the minimal requirement does not imply that every RFID system constructed based on PRF or its equivalents is unp*-privacy. For example, the RFID systems given in Ohkubo et al. [2004] and Peris-Lopez et al. [2006] are reported to have privacy vulnerabilities, though they are constructed based on symmetric encryption schemes or cryptographic hash functions. How to apply PRF or its equivalents to design

an efficient and low-cost RFID system with unp*-privacy remains an interesting area for further investigation.

The new protocol we provided in Section 6.2 (also see Figure 11) can be considered as an example of such design. One advantage of our protocol is that it is most efficient in identifying a tag in normal situations in which desynchronization does not happen frequently; it resorts occasionally to exhaustive search to identify a tag that has been desynchronized, but resumes to exact match of index again after a successful read of the tag until the next desynchronization attack.

7. CONCLUSION AND OPEN PROBLEM

In this article, we presented the limitations of the existing unpredictability-based RFID privacy models and proposed a new unpredictability-based privacy model, denoted as unp*-privacy, based on the indistinguishability of a real tag and a virtual tag. We investigated the relationship between unp*-privacy and ind-privacy, which is another RFID privacy model based on the indistinguishability of two tags. We proved that ind-privacy is weaker than unp*-privacy. We further investigated the minimal requirement on RFID tags for enforcing unp*-privacy. Our result shows that RFID tags must be empowered with the ability to compute a PRF family or its equivalents so as to construct a complete and sound RFID system with provable unp*-privacy. This result can be used to explain why many existing lightweight RFID protocols have privacy flaws.

Our minimal condition reflects the equivalence between the unp*-privacy and the PRF family. According to our results, PRF can also be used to construct RFID systems with ind-privacy. However, the other direction is uncertain. An open problem is to find the minimal condition for enforcing ind-privacy in RFID systems.

ACKNOWLEDGMENTS

The authors are grateful to the editor-in-chief, the associate editor, and the anonymous reviewers for their valuable comments.

REFERENCES

ATENIESE, G., CAMENISCH, J., AND DE MEDEIROS, B. 2005. Untraceable RFID tags via insubvertible encryption. In *Proceedings of the ACM Conference on Computer and Communications Security*. 92–101.

AVOINE, G. 2005. Adversarial model for radio frequency identification. Cryptology ePrint Archive, Report 2005/049. http://eprint.iacr.org/.

AVOINE, G., DYSLI, E., AND OECHSLIN, P. 2005. Reducing time complexity in RFID systems. In *Proceedings of the 12th Annual Workshop on Selected Areas in Cryptography*. 291–306.

BOGDANOV, A., KNUDSEN, L. R., LEANDER, G., PAAR, C., POSCHMANN, A., ROBSHAW, M. J. B., SEURIN.. Y., AND VIKKELSOE, C. 2007. PRESENT: An ultra-lightweight block cipher. In *Proceedings of the Workshop on Cryptographic Hardware and Embedded Systems*. 450–466.

CHIEN, H.-Y. AND CHEN, C.-H. 2007. Mutual authentication protocol for RFID conforming to EPC class 1 generation 2 standards. *Comput. Stand. Interf. 29*, 2, 254–259.

DAMGÄRD, I. AND PEDERSEN, M. O. 2008. RFID security: Tradeoffs between security and efficiency. In *Proceedings of the Cryptographers' Track of the RSA Conference*. 318–332.

DUC, D. N., PARK, J., LEE, H., AND KIM, K. 2006. Enhancing security of EPCglobal gen-2 RFID tag against traceability and cloning. In *Proceedings of the Symposium on Cryptography and Information Security*.

EISENBARTH, T., KUMAR, S., PAAR, C., POSCHMANN, A., AND UHSADEL, L. 2007. A survey of lightweight-cryptography implementations. *IEEE Des. Test. Comput. 24*, 6, 522–533.

FELDHOFER, M., WOLKERSTORFER, J., AND RIJMEN, V. 2005. AES implementation on a grain of sand. *IEE Proc. Inform. Sec. 152*, 1, 13–20.

GARFINKEL, S. L., JUELS, A., AND PAPPU, R. 2005. RFID privacy: An overview of problems and proposed solutions. *IEEE Sec. Priv. 3*, 3, 34–43.

GOLDREICH, O., GOLDWASSER, S., AND MICALI, S. 1986. How to construct random functions. *J. ACM 33*, 4, 792–807.

HA, J., MOON, S.-J., ZHOU, J., AND HA, J. 2008. A new formal proof model for RFID location privacy. In *Proceedings of the European Symposium on Research in Computer Security (ESORICS)*. 267–281.

HOPPER, N. J. AND BLUM, M. 2001. Secure human identification protocols. In *Proceedings of the Annual Cryptology Conference (ASIACRYPT)*. 52–66.

JUELS, A. 2004. Minimalist cryptography for low-cost RFID tags. In *Proceedings of the Conference on Security in Communication Networks*. 149–164.

JUELS, A. 2006. RFID security and privacy: a research survey. *IEEE J. Select. Areas Comm. 24*, 2, 381–394.

JUELS, A., PAPPU, R., AND PARNO, B. 2008. Unidirectional key distribution across time and space with applications to RFID security. In *Proceedings of the USENIX Security Symposium*. 75–90.

JUELS, A., RIVEST, R. L., AND SZYDLO, M. 2003. The blocker tag: selective blocking of RFID tags for consumer privacy. In *Proceedings of the ACM Conference on Computer and Communications Security*. 103–111.

JUELS, A. AND WEIS, S. A. 2005. Authenticating pervasive devices with human protocols. In *Proceedings of the Annual Cryptology Conference (CRYPTO)*. 293–308.

JUELS, A. AND WEIS, S. A. Defining strong privacy for RFID. In P*roceedings of the IEEE Pervasive Computing and Communication Conference*. 342–347.

KARTHIKEYAN, S. AND NESTERENKO, M. 2005. RFID security without extensive cryptography. In *Proceedings of the ACM Workshop on Security of Ad Hoc and Sensor Networks*. 63–67.

KATZ, J. AND SHIN, J. S. 2006. Parallel and concurrent security of the hb and hb+ protocols. In *Proceedings of the Annual Cryptology Conference (EUROCRYPT)*.73–87.

KONIDALA, D. M., KIM, Z., AND KIM, K. 2007. A simple and cost-effective RFID tag-reader mutual authentication scheme. In *Proceedings of the Conference on RFID Security*. 141–152.

KUMAR, S. AND PAAR, C. 2006. Are standards compliant elliptic curve cryptosystems feasible on RFID? In *Proceedings of the Workshop on RFID Security*.

MA, C., LI, Y., DENG, R. H., AND LI, T. 2009. RFID privacy: relation between two notions, minimal condition, and efficient construction. In *Proceedings of the ACM Conference on Computer and Communications Security*. 54–65.

MOLNAR, D. AND WAGNER, D. 2004. Privacy and security in library RFID: issues, practices, and architectures. In *Proceedings of the ACM Conference on Computer and Communications Security*. 210–219.

NG, C. Y., SUSILO, W., MU, Y., AND SAFAVI-NAINI, R. 2008. RFID privacy models revisited. In *Proceedings of the European Symposium on Research in Computer Security*. 251–266.

OHKUBO, M., SUZUKI, K., AND KINOSHITA, S. 2004. Efficient hash-chain based RFID privacy protection scheme. In *Proceedings of the International Conference on Ubiquitous Computing—Ubicomp, Workshop Privacy: Current Status and Future Directions*.

PAISE, R.-I. AND VAUDENAY, S. 2008. Mutual authentication in RFID: security and privacy. In *Proceedings of the Asian Conference on Computer Security*. 292–299.

PERIS-LOPEZ, P., CASTRO, J. C. H., ESTEVEZ-TAPIADOR, J. M., AND RIBAGORDA, A. 2006. RFID systems: A survey on security threats and proposed solutions. In *Proceedings of the 11th IFIP International Conference on Personal Wireless Communications*. 159–170.

PERIS-LOPEZ, P., LI, T., TONG LEE, L., HERNANDEZ-CASTRO, J. C., AND ESTEVEZ-TAPIADOR, J. M. 2008. Vulnerability analysis of a mutual authentication scheme under the EPC Class-1 Generation-2 Standard. In *Proceedings of the Workshop on RFID Security.*

SAMARATI, P. AND SWEENEY, L. 1998. Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. Tech. rep., SRI International.

SARMA, S. E., WEIS, S. A., AND ENGELS, D. W. 2003. Radio-frequency identification: Security risks and challenges. *Crytobytes, RSA Labs. 6*, 1, 2–9.

SPIEKERMANN, S. AND EVDOKIMOV, S. 2009. Privacy enhancing technologies for RFID—A critical investigation of state of the art research. *IEEE Priv. Sec*.

TSUDIK, G. 2006. YA-TRAP: Yet another trivial RFID authentication protocol. In *Proceedings of the Intemational Conference on Pervasive Computing and Communications*. 640–643.

TSUDIK, G. 2007. A family of dunces: Trivial RFID identification and authentication protocols. In *Proceedings of the 7th International Conference on Privacy Enhancing Technologies.* 45–61.

VAN DEURSEN, T. AND RADOMIROVIC, S. 2008. Attacks on RFID protocols. Cryptology ePrint Archive, Report 2008/310. http://eprint.iacr.org/.

VAN DEURSEN, T. AND RADOMIROVIC, S. 2009. On a new formal proof model for RFID location privacy, *Inform. Process. Lett. 110*, 2, 57–61.

VAUDENAY, S. 2007. On privacy models for RFID. In *Proceedings of the Annual Cryptology Conference (ASIACRYPT'07)*. K. Kurosawa, Ed., Lecture Notes in Computer Science, vol. 4833, Springer, 68–87.