RFID Privacy: Relation Between Two Notions, Minimal Condition, and Efficient Construction

Changshe Ma[*]
School of Information Systems
Singapore Management University
80 Stamford Road, Singapore 178902
changshema@smu.edu.sg

Yingjiu Li[†]
School of Information Systems
Singapore Management University
80 Stamford Road, Singapore 178902
yjli@smu.edu.sg

Robert H. Deng
School of Information Systems
Singapore Management University
80 Stamford Road, Singapore 178902
robertdeng@smu.edu.sg

Tieyan Li
Institute for Infocomm Research
1 Fusionopolis Way
Singapore 138632
litieyan@i2r.a-star.edu.sg

ABSTRACT

Privacy of RFID systems is receiving increasing attention in the RFID community. Basically, there are two kinds of RFID privacy notions: one based on the indistinguishability of two tags, denoted as ind-privacy, and the other based on the unpredictability of the output of a protocol, denoted as unp-privacy. In this paper, the definition of unp-privacy is refined and the relation between the two notions is clarified: 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 has strong (or weak) unp-privacy *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 strong (or weak) unp-privacy. In this sense, a pseudorandom function family is the minimal requirement on an RFID tag's computational power for enforcing strong RFID system privacy. Finally, a new RFID protocol is proposed to satisfy the minimal requirement, which also outperforms the state-of-the-art RFID protocols in terms of computational cost and communication overhead.

[*]Dr. Ma's original affiliation is School of Computer, South China Normal University, Guangzhou, China, 510631.

[†]Contact author.

CCS'09, November 9–13, 2009, Chicago, Illinois, USA.
Copyright 2009 ACM 978-1-60558-352-5/09/11 ...$10.00.

Categories and Subject Descriptors

C.2.0 [**General**]: Security and protection; D.4.6 [**Operation Systems**]: Security and protection-cryptographic controls

General Terms

Security, design

Keywords

RFID, privacy, pseudorandom function

1. INTRODUCTION

Radio Frequency IDentification (RFID) [8] is an automated object identification technology, where a reader identifies tags via wireless channels. If an RFID system is not appropriately desinged or implemented, the absence of physical contact during the identification process may cause *privacy* issues [15, 28] of the tags and hence, of their owners or bearers. Much effort [2, 4, 9, 16, 17, 24, 33] has been made to address the privacy issues in RFID systems. The effort has been mostly focused in two aspects: one is to construct RFID protocols [27, 34, 28] 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 RFID privacy notions have been proposed: one based on the indistinguishability of two tags [18], denoted as ind-privacy, and the other based on the unpredictability of the output of a protocol [12], denoted as unp-privacy. In this paper, we closely examine the privacy notions, explain why many existing protocols have privacy flaws, and construct an efficient protocol with strong privacy.

One fundamental problem we investigate regards the relationship between the two notions of RFID system privacy. The intuition of ind-privacy [18] is that none can link a tag and its behaviors without learning its internal states, while the essence of unp-privacy [12] is that no adversary can predict the output of a tag or a reader when engaging in an RFID protocol. It is not clear in the literature whether these two notions are equivalent or one implies the other, since it is difficult to bridge the gap between the adversary's power in the two privacy definitions. To understand which level of privacy an RFID system provides, it is critical to clarify the relationship between the two 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 strong 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 paper, we address the above two basic problems for RFID privacy and make the following contributions:

1. We refine the unp-privacy model for RFID systems. As pointed out in [35], the unp-privacy notion originally proposed in [12] is incomplete. We reconsider it based on the fact that privacy is relative to the behaviors of the whole RFID system, not only of the tags. A complete definition of unp-privacy is introduced through the infeasibility to infer the output of an RFID protocol rather than the output of any tag. This definition is compatible with the privacy notion, unobservability, in Common Criteria [1].

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 an extended unp-privacy model, named as eunp-privacy, which is proven to be equivalent to unp-privacy and to imply ind-privacy. Moreover, we show that ind-privacy does not imply unp-privacy by constructing an RFID system which is of ind-privacy but not unp-privacy.

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 of strong (or weak) unp-privacy, 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 each tag is endowed with the power to compute a PRF or its equivalents, then an RFID system with strong (or weak) unp-privacy can be constructed accordingly. The minimal requirement on the computational power for RFID tags shows that (even weak) 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 construct an efficient RFID protocol with strong unp-privacy (see section 5.2). Our protocol requires a minimum of two rounds of communication and two PRF computations in each invocation. In the case that a tag has not been desynchronized (e.g., due to attacks) since the last successful read of the tags, our protocol requires the minimal computational cost for identifying the tag (in exact match). In the case that the tag has just been desynchronized, our protocol requires exhaustive search for identifying the tag as in most of the existing protocols.

For ease of reference, we summarize our findings in Figure 1 regarding the relationships among privacy notions and tag's ability to compute PRF.

1.2 Related Work

The work most related to ours is the formalization of privacy model for RFID systems. Avoine [3] first formalized the adversary model in RFID systems. Based on the adversary model, Juels and Weis defined the notion of strong privacy [18], and Damgård and Østergaard considered the completeness and soundness [7] for RFID systems. In [36], Vaudenay considered the side-channel attacks in the privacy model and proposed eight classes of privacy levels. The eight classes were later refined to three by Ng et al. [25]. The privacy notions used in these works are all based on the indistinguishability of two tags in RFID communications. In [12], Ha et al. proposed a different privacy model based on the unpredictability of tag outputs, though this model was later shown to be incomplete [35]. In the literature, the relationship between the two privacy models has not been rigorously addressed. In this paper, we show that the unpredictability-based definition, after refinement, implies the indistinguishability-based definition.

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 [8]. Sarma et al. analyzed the gate complexity of the embedded chip with respect to the cost per tag [31, 32]. The gate count of low-cost tags is $5,000-10,000$ [8]. 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 (e.g. [20, 26, 6, 22]) 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 [29]. In [14], Juels 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 (even weak 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 strong privacy with minimal requirement on RFID tags.

1.3 Organization of the Paper

The rest of the paper is organized as follows. In section 2, we define the mathematical notations and pseudorandom functions used in this paper. In section 3, we introduce two privacy models, ind-privacvy and unp-privacy, for RFID

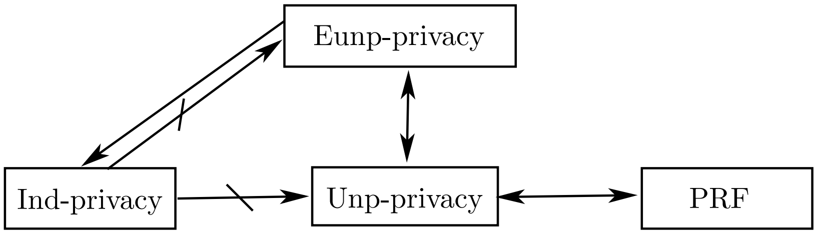


Figure 1: Relationships Among Privacy Notions

systems. In section 4, we clarify and prove the relationship between the two privacy models. In section 5, we show that the minimal requirement to guarantee strong (or weak) unp-privacy is equipping each tag with the ability to compute a pseudorandom function. We also provide an efficient construction of RFID protocol (in section 5.2) with strong unp-privacy according to the minimal requirement on tags. In section 6, we conclude this paper and discuss some open problems.

2. PRELIMINARIES

2.1 Mathematical Notations

If $A(\cdot, \cdot, ...)$ is a randomized algorithm, then $y \leftarrow A(x_1, x_2, ...; cn)$ means that y is assigned with the unique output of the algorithm A on inputs $x_1, x_2, ...$ and coins cn, while $y \overset{\$}{\leftarrow} A(x_1, x_2, ...)$ is a shorthand for first picking cn at random and then setting $y \leftarrow A(x_1, x_2, ...; cn)$. Let $y \leftarrow A^{O_1,...,O_n}(x_1, x_2, ...)$ denote that y is assigned with the output of the algorithm A which takes $x_1, x_2, ...$ as inputs and has oracle accesses to $O_1, ..., O_n$. If S is a set, then $s \in_R S$ indicates that s is chosen uniformly at random from S. If $x_1, x_2, ...$ 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 $f : \mathcal{N} \rightarrow R$ 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

```
$\mathrm{Exp}_T^{ptpt}(F, \gamma, m, n, j)$
    1. $k \in_R \mathcal{K}$ and set $f = F_k$
    2. $x \leftarrow T^{O_f}(\gamma, m, n, j)$
    3. $b \in_R \{0, 1\}$
    4. if $b = 1$ then $y \leftarrow f(x)$, otherwise $y \in_R \mathcal{R}$
    5. $b' \leftarrow T(y)$
```

Figure 2: Polynomial Time Predictable Test

Let $F : \mathcal{K} \times \mathcal{D} \rightarrow \mathcal{R}$ 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 $|\mathcal{R}| = n$. A *polynomial time predictable test* (*PTPT*) for F is an experiment, where a probabilistic polynomial time algorithm T, given γ, m, n, j as input and with access to an oracle O_f for a function $f \in F$, outputs either 0 or 1. Figure 2 shows a PTPT for F. At first, algorithm T queries the oracle O_f about $x_1, ..., x_j$. Then, it outputs $x \in \mathcal{D}$ such that $x \neq x_1, ..., x_j$. This x is called the *chosen exam.* At this point, algorithm T is not allowed to query oracle O_f any more. The experiment tosses a random bit $b \in \{0, 1\}$. If $b = 1$, then $f(x)$ is given to the algorithm T; otherwise, $y \in_R \mathcal{R}$ is given to T. Finally, the algorithm T is required to output a bit b' by guessing which of the two values is given to it: $b' = 1$ for $f(x)$, and $b' = 0$ for y.

Definition 2.2. An algorithm T passes the *PTPT* for the function family F if it correctly guesses which of the two values ($f(x)$ and y) is the function value $f(x)$, i.e. $b' = b$. The advantage of algorithm T is defined as

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

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 $F : \mathcal{K} \times \mathcal{D} \rightarrow \mathcal{R}$ 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. PRIVACY DEFINITIONS OF RFID SYSTEMS

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

3.1 Model of RFID Systems

For simplicity, we consider an RFID system comprising of a single legitimate reader[1] R and a set of ℓ tags $\mathcal{T}_1, ..., \mathcal{T}_\ell$. The reader and the tags are probabilistic polynomial time

[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.

interactive Turing machines. T̃ypically, 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 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).

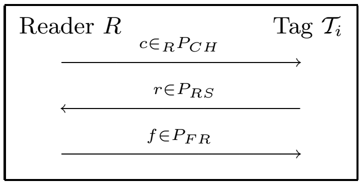


Figure 3: Canonical RFID Protocol

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 $r = (r_1, cn_{\mathcal{T}_i}, s_{\mathcal{T}_i})$, where r_1 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 $s_{\mathcal{T}_i}$. As an abusing of the notation, we allow the coin toss and/or the internal state in the response message r to be empty string in some cases. We write r_1 as $r_1 = F_{k_{\mathcal{T}_i}}(c, cn_{\mathcal{T}_i}, s_{\mathcal{T}_i})$, where $F_{k_{\mathcal{T}_i}}$ is a function computed by the tag. This protocol can be executed in two or three rounds. In the third round, if exits, 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 massage c, and the tag's response r. We write it as $f = \tilde{F}_{k_R}(s_R, c, r, cn_R)$, where $\tilde{F}_{k_R}$ 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 $P_{CH}, P_{FT}, P_{FR}, P_{CN}, P_S$ denote the challenge message space, the range of function $F_{k_{\mathcal{T}_i}}$, the final message space, the coin space of the tag, and the state information space of the tag, respectively. Let $P_{RS} = P_{FT} \times P_{CN} \times P_S$. The view of an adversary about the protocol π is the set $\{(c, r, f)\}$. Throughout this paper, we only consider RFID protocols in this canonical form.

Definition 3.1. An RFID system RS is defined to be a tuple $(R, \mathcal{T}, \mathsf{InitializeR}, \mathsf{InitializeT}, \pi)$, where

InitializeR(κ) 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.

InitializeT($\mathcal{T}_i, \kappa$) 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.

Protocol $\pi(R, \mathcal{T}_i)$ 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 the 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 protocol message 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 $\pi(R, \mathcal{T}_i)$.

3.2 Description of the 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 $r_{sid} \in P_{RS}$.

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

Let O_1, O_2, O_3 and O_4 denote $\mathsf{InitReader}$, $\mathsf{InitTag}$, SetTag and $\mathsf{SendRes}$ 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: first call $\mathsf{InitReader}()$ to get (sid, c_{sid}), then call $\mathsf{InitTag}(sid, c_{sid})$ to get (sid, r_{sid}), and finally call $\mathsf{SendRes}(sid, c_{sid}, r_{sid})$ 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.

[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 [13, 19, 21], which require multiple rounds to authenticate each tag in a statistical sense. We consider it an open problem to extend our research to those protocols.

3.3 Completeness and Soundness of RFID Systems

Here, we review the definitions of completeness and soundness of RFID systems presented in [7]. 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} was correctly generated by a tag. Completeness means that the reader outputs "accept" with probability 1 for any such session.

```
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.$\{(c_{sid^*}, r_{sid^*}, f_{sid^*}), \mathcal{T}_j\} \leftarrow \mathcal{A}^{O_1,O_2,O_4}(R, \mathcal{T})$.
```

Figure 4: Soundness Experiment

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 $\mathcal{T}_j$ 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 Definitions of Privacy

We now present "privacy experiments" which is similar to the classical definition of indistinguishability of objects. We define two kinds of privacy experiments for RFID systems, and hence provide two notions of privacy for RFID systems, which summarize the work of [18] and refine the work of [12]. In the next section, we will clarify the relations between these two notions.

3.4.1 Indistinguishability-Based Privacy

We first consider the ind-privacy experiment for defining the ind-privacy of RFID system *RS*. Figure 5 illustrates the ind-privacy experiment $\mathbf{Exp}_{\mathcal{A}}^{ind}[\kappa, \ell, q, s, u, v]$ ($\mathbf{Exp}_{\mathcal{A}}^{ind}$, 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 $\mathcal{T} = \{\mathcal{T}_1, ..., \mathcal{T}_\ell\}$ 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($\mathcal{T}_c$).

```
Experiment $\mathbf{Exp}_{\mathcal{A}}^{ind}[\kappa, \ell, q, s, u, v]$
  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.
```

Figure 5: Ind-Privacy Experiment

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

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

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

Also, we define **weak $(\epsilon, t, q, s, 0, v)$-ind-privacy** the same as the strong $(\epsilon, t, q, s, u, v)$-ind-privacy except that the adversary is not allowed to corrupt any tags (hence $u = 0$).

Remark 2. The indistinguishability-based 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 has not been corrupted. This latter case may be considered as an application of the notion of ι-privacy (or ι-anonymity) [30] in the RFID system we defined.

3.4.2 Unpredictability-Based Privacy

```
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 $(c_0, r_0, f_0) \leftarrow \pi(R, \mathcal{T}_c, sid)$ and $(r^*, f^*) = (r_0, f_0)$;
  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.
```

Figure 6: Unp-Privacy Experiment

[3] Our definition of soundness is compatible with the weak soundness introduced in [7], 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 $\mathcal{T}_j$).

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

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

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

Also, we define **weak $(\epsilon, t, q, s, 0, v)$-unp-privacy** the same as the strong $(\epsilon, t, q, s, u, v)$-unp-privacy except that the adversary is not allowed to corrupt any tags.

Remark 3. Our strong privacy definitions can be extended to model forward privacy and backward privacy. The only difference is that the adversary is allowed to corrupt the challenge tag(s) in the *learning* stage of backward privacy experiment and in the *guess* stage of forward privacy experiment, respectively, and that the experiment is required to send SetTag queries to update the selected tag(s) to a new state before it proceeds to generate a challenge tag (for ind-privacy) or challenge messages (for unp-privacy) for the adversary. It is out of the scope of this paper to investigate such extended privacy model, which can be used to formalize secure ownership transfer of RFID tags among multiple parties.

4. RELATIONS

In this section, we investigate the relations between the ind-privacy and unp-privacy. We introduce an extended unp-privacy model as a "bridge" to show that it is equivalent to unp-privacy and it implies ind-privacy.

4.1 Extended Unp-Privacy

It is difficult to prove that unp-privacy implies ind-privacy directly, because there is essential difference between the adversary's power in ind-privacy experiment and that in unp-privacy experiment. During the *guess stage*, the adversary is allowed to query O_1, O_2 and O_4 oracles to the challenge tag $\mathcal{T}_c$ in the ind-privacy experiment, while it is not allowed to query any oracle to $\mathcal{T}_c$ in the unp-privacy experiment. Hence, it is impossible to answer the adversary's queries related to the challenge tag during *guess stage* in the ind-privacy experiment via the unp-privacy experiment. To circumvent this difficulty, we extend the power of the adversary in the unp-privacy experiment by allowing it to query multiple *test* messages in the *guess stage*. This extension will help us to answer the adversary's queries in *guess stage* in the ind-privacy experiment with a probability at least $\frac{1}{2}$. Moreover, eunp-privacy can be proven to be equivalent to unp-privacy via the hybrid argument approach [10].

```
Experiment $\mathbf{Exp}_{\mathcal{A}}^{eunp}[\kappa, \ell, q, s, u, v, w]$
  1. setup the reader $R$ and a set of tags $\mathcal{T}$ with $|\mathcal{T}| = \ell$;
  2. $\{\mathcal{T}_c, st\} \leftarrow \mathcal{A}_1^{O_1,O_2,O_3,O_4}(R, \mathcal{T})$; //learning stage
  3. set $\mathcal{T}' = \mathcal{T} - \{\mathcal{T}_c\}$;
  4. $b \in_R \{0, 1\}$;
  5. let $st_0 = st$ and $cs = \{\varepsilon\}$, for $i = 1$ to $w$
    5.1 $(c_i, st_i) \leftarrow \mathcal{A}_2^{O_1,O_2,O_3,O_4}(R, \mathcal{T}', st_{i-1}, cs)$;
    5.2 if $b = 0$ then $(r_i^*, f_i^*) \in_R P_{RS} \times P_{FR}$,
        else $(c_i, r_i, f_i) \leftarrow \pi(R, \mathcal{T}_c, sid_i)$ and $(r_i^*, f_i^*) = (r_i, f_i)$;
    5.3 $cs = cs \cup \{(r_i^*, f_i^*)\}$
  6. $b' \leftarrow \mathcal{A}_2^{O_1,O_2,O_3,O_4}(R, \mathcal{T}', st_w, cs)$; //guess stage
  7. the experiment outputs 1 if $b' = b$, 0 otherwise.
```

Figure 7: Eunp-Privacy Experiment

Extended Unp-Privacy. Figure 7 shows the extended unp-privacy experiment $\mathbf{Exp}_{\mathcal{A}}^{eunp}[\kappa, \ell, q, s, u, v, w]$ ($\mathbf{Exp}_{\mathcal{A}}^{eunp}$, for simplicity), which is the same as unp-privacy experiment except step (5). In the extended unp-privacy experiment, step (5) is defined as follows. The adversary is allowed to challenge for w *test* messages rather than only one *test* message as in the unp-privacy experiment. For all the w *test* messages, the experiment uses the same coin $b \in_R \{0, 1\}$. If $b = 1$, algorithm $\mathcal{A}_2$ is given challenge messages which are all selected from protocol messages; otherwise, $\mathcal{A}_2$ is given random challenge messages all selected from $P_{RS} \times P_{FR}$. Let st_i denote the state information generated by algorithm $\mathcal{A}_2$ when it generates the ith *test* message c_i. Let *cs* denote the set of challenge messages which are given to $\mathcal{A}_2$. Algorithm $\mathcal{A}_2$ may choose the w *test* messages adaptively: it may choose c_i according to the state information st_{i-1}, the previous challenge message set *cs*, and its own strategy.

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

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

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 strong eunp-privacy of RFID system *RS* if its advantage $\mathrm{Adv}_{\mathcal{A}}^{eunp}(k, \ell, q, s, u, v, w)$ in the experiment $\mathbf{Exp}_{\mathcal{A}}^{eunp}$ is at least ϵ and its running time is at most t.

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

Also, we define **weak $(\epsilon, t, q, s, 0, v, w)$-eunp-privacy** the same as the strong $(\epsilon, t, q, s, u, v, w)$-eunp-privacy except that the adversary is not allowed to corrupt any tags.

4.2 Unp-Privacy $\Longleftrightarrow$ Eunp-Privacy

Although the ability of the adversary in eunp-privacy experiment is different from that in unp-privacy experiment, we can still use unp-privacy experiment to simulate eunp-privacy experiment through the hybrid argument approach [10] and derive the following

Theorem 1. *For an RFID system $RS = (R, \mathcal{T}, \mathsf{InitializeR}, \mathsf{InitializeT}, \pi)$, strong (or weak) unp-privacy is equivalent to strong (or weak) eunp-privacy.*

Proof. It is obvious that strong eunp-privacy $\Longrightarrow$ strong unp-privacy holds.

Now we prove that strong eunp-privacy $\Longleftarrow$ strong unp-privacy. Assume that RS is not strong eunp-private. That is, there exists an adversary $\mathcal{A}$ such that it $(\epsilon, t, q_1, s, u, v, w)$-breaks the eunp-privacy of RS. We construct an algorithm $\mathcal{B}$ that uses $\mathcal{A}$ as a subroutine and $(\frac{\epsilon}{2w}, t, q_2, s, u, v)$-breaks the unp-privacy of RS, where $s_1 + w \leqslant s_2$. The algorithm $\mathcal{B}$ proceeds as follows. On the input of the RFID system RS and the security parameter κ, it first chooses an *index* i between 0 and $w-1$ with uniform probability. Next, algorithm $\mathcal{B}$ invokes adversary $\mathcal{A}$ with input RS and κ and conducts the eunp-privacy experiment with $\mathcal{A}$ as follows.

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

Simulate the challenge messages: When adversary $\mathcal{A}$ outputs the challenge tag $\mathcal{T}_c$, algorithm $\mathcal{B}$ also sets the tag $\mathcal{T}_c$ as its challenge tag. Then, it generates the challenge messages for $\mathcal{A}$'s subsequent w *test* messages as follows.

1. Algorithm $\mathcal{B}$ answers $\mathcal{A}$'s first i queries by asking the same queries to the unp-privacy experiment.

2. When adversary $\mathcal{A}$ asks its $(i+1)$-th query c_{i+1}, algorithm $\mathcal{B}$ sets c_{i+1} as its *test* message and ends the *learning stage* with the output $(\mathcal{T}_c, c_{i+1})$. Upon receiving the challenge message (r_{i+1}, f_{i+1}) from the unp-privacy experiment, $\mathcal{B}$ gives it to $\mathcal{A}$ as the challenge message for $\mathcal{A}$'s *test* message c_{i+1}.

3. Next, algorithm $\mathcal{B}$ continues to answer $\mathcal{A}$'s *test* messages $c_{i+2}, \ldots, c_w$ by randomly selecting pairs $(r, f) \in_R P_{RS} \times P_{FR}$.

Output: If $\mathcal{A}$ outputs a bit b', then $\mathcal{B}$ outputs a bit $b = b'$.

Probability Analysis: Prior to assess the success probability of algorithm $\mathcal{B}$, we consider the following (RS, κ, i)-*experiment*:

Run $\mathcal{A}$ with the input of RS and κ and follow the eunp-privacy experiment except for the step (5). Let c_j be the jth *test* message of $\mathcal{A}$. The step (5) proceeds as follows:

If $j \leqslant i$, then answer with (r_j, f_j) such that $(c_j, r_j, f_j) \leftarrow \pi(R, \mathcal{T}_c, sid)$; else answer with a pair $(r_j, f_j) \in_R P_{RS} \times P_{FR}$.

Let p_κ^i be the probability that $\mathcal{A}$ outputs 1 in the (RS, κ, i)-*experiment*. Note that p_κ^0 (or p_κ^w) is the probability that $\mathcal{A}$ outputs 1 in eunp-privacy experiment with random bit $b = 0$ (or 1). Let the random bit in unp-privacy experiment be b''. We can calculate the probability that algorithm $\mathcal{B}$ makes a correct guess of b'' on input RS and κ in unp-privacy experiment. Consider the executions of $\mathcal{B}$. Let $\mathcal{B}_i$ denote the event "Algorithm $\mathcal{B}$ chooses *index* $= i$." Then

$$\Pr[\mathcal{B} \text{ is correct}] = \sum_{i=0}^{w-1} \Pr[\mathcal{B} \text{ is correct}|\mathcal{B}_i]\Pr[\mathcal{B}_i]$$

$$= \frac{1}{w}\sum_{i=0}^{w-1}(\Pr[b = 1 \wedge b'' = 1|\mathcal{B}_i] + \Pr[b = 0 \wedge b'' = 0|\mathcal{B}_i])$$

$$= \frac{1}{w}\sum_{i=0}^{w-1}(\frac{1}{2}\Pr[\mathcal{A} \text{ outputs } 1|b'' = 1 \wedge \mathcal{B}_i]$$

$$+ \frac{1}{2}\Pr[\mathcal{A} \text{ outputs } 0|b'' = 0 \wedge \mathcal{B}_i])$$

$$= \frac{1}{w}\sum_{i=0}^{w-1}\frac{1}{2}(p_\kappa^{i+1} + 1 - p_\kappa^i)$$

$$\geqslant \frac{1}{2} + \frac{\epsilon}{2w}$$

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

It is not hard to show that weak unp-privacy is equivalent to weak eunp-privacy according to the method mentioned above. ■

4.3 Eunp-Privacy $\Longrightarrow$ Ind-Privacy

Theorem 2. *Assume that the RFID system $RS = (R, \mathcal{T}, \mathsf{InitializeR}, \mathsf{InitializeT}, \pi)$ is $(\frac{\epsilon}{q_2}, t, q_2, s_2, v_2)$-sound and complete. If it is strong (or weak) $(\frac{\epsilon}{6}, t, q_1, s_1, u_1, v_1, w)$-eunp-private, then it is strong (or weak) $(\epsilon, t, q_2, s_2, u_2, v_2)$-ind-private, where $q_1 \geqslant q_2$, $s_1 \geqslant s_2$, $u_1 \geqslant u_2$, $v_1 \geqslant v_2$ and $w \geqslant q_2$.*

Proof. Here, we only consider the proof for the case of strong privacy, as the proof for the case of weak privacy can be carried out similarly. Assume that RS is not strong ind-private. That is, there exists an adversary $\mathcal{A}$ which can $(\epsilon, t, q_2, s_2, u_2, v_2)$-break the ind-privacy of RS. Then, we construct an algorithm $\mathcal{B}$ which runs $\mathcal{A}$ as a subroutine and $(\frac{\epsilon}{6}, t, q_1, s_1, u_1, v_1, w)$-breaks the eunp-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 queries: Algorithm $\mathcal{B}$ answers adversary $\mathcal{A}$'s queries by asking them to the eunp-privacy experiment.

Simulate the guess stage: When adversary $\mathcal{A}$ submits two challenge tags $\mathcal{T}_i$ and $\mathcal{T}_j$, algorithm $\mathcal{B}$ selects a random bit $b \in_R \{0, 1\}$ and returns $\mathcal{T}_c$ to $\mathcal{A}$, where $\mathcal{T}_c = \mathcal{T}_i$ if $b = 0$, otherwise $\mathcal{T}_c = \mathcal{T}_j$. Algorithm $\mathcal{B}$ ends the *learning stage* and outputs $\mathcal{T}_b$ as the challenge tag for the eunp-privacy experiment. After that, when adversary $\mathcal{A}$ issues a query of $\mathsf{InitTag}(\mathcal{T}_c, sid, c)$, algorithm $\mathcal{B}$ sends a *test* message query of c to the eunp-privacy experiment, returns the first part r of the response to $\mathcal{A}$, and stores the second part f for answering $\mathcal{A}$'s subsequent query of $\mathsf{SendRes}(sid, c, r)$. If $\mathcal{A}$ issues queries related to other tags (not to the tag $\mathcal{T}_c$), algorithm $\mathcal{B}$ answers them by asking the same queries to the eunp-privacy experiment.

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

Let the internal random bit of the eunp-privacy experiment be $\hat{b}$. Next, we assess the probability that algorithm $\mathcal{B}$ makes a correct guess of $\hat{b}$.

$$
\begin{aligned}
&\Pr[\mathcal{B} \text{ is correct}] = \Pr[\bar{b} = \hat{b}] \\
&= \Pr[(\bar{b} = 0|\hat{b} = 0) \wedge \hat{b} = 0] + \Pr[(\bar{b} = 1|\hat{b} = 1) \wedge \hat{b} = 1] \\
&= \frac{1}{2}(\Pr[(b \neq b'|\hat{b} = 0)] + \Pr[(b = b'|\hat{b} = 1)]) \\
&\geqslant \frac{1}{2} + \frac{\epsilon}{6} \qquad (2)
\end{aligned}
$$

The inequality (2) holds due to the following two inequalities

$$\Pr[(b \neq b'|b'' = 0)] \geqslant \frac{1}{2} - \frac{2\epsilon}{3} \qquad (3)$$

and

$$\Pr[(b = b'|b'' = 1)] \geqslant \frac{1}{2} + \epsilon. \qquad (4)$$

It is clear that inequality (4) holds. Now, we justify the inequality of (3). After adversary $\mathcal{A}$ receives the challenge tag $\mathcal{T}_b$, it can query $\mathsf{InitTag}(\mathcal{T}_b, sid, c_{sid})$ for at most q_2 times. When $b'' = 0$, the eunp-privacy experiment answers random message pair (r, f) to $\mathcal{B}$'s *test* message query c_{sid}, which implies that $\mathcal{B}$ also answers random message to $\mathcal{A}$'s every $\mathsf{InitTag}(\mathcal{T}_b, \cdot, \cdot)$ query. For a random message pair (r, f), the probability that $(c_{sid}, r, f) = \pi(R, \mathcal{T}_b, sid)$ is at most $\frac{\epsilon}{q_2}$, since the RFID system is $(\frac{\epsilon}{q_2}, t, q_2, s_2, v_2)$-sound. Hence, all $\mathcal{B}$'s answers are not protocol messages with a probability at least $(1 - \frac{\epsilon}{q_2})^{q_2}$. Under the condition that all $\mathcal{B}$'s answers are not protocol messages, the adversary $\mathcal{A}$ learns nothing about $\mathcal{T}_b$ and hence the probability that its output equals to the random bit b is exactly $\frac{1}{2}$. We have,

$$\Pr[b = b'|b'' = 0] \leqslant \frac{1}{2}(1 - \frac{\epsilon}{q_2})^{q_2} + (\frac{1}{2} + \epsilon)(1 - (1 - \frac{\epsilon}{q_2})^{q_2}).$$

Therefore,

$$
\begin{aligned}
\Pr[b \neq b'|b'' = 0] &\geqslant \frac{1}{2} + (\frac{1}{e} - 1)\epsilon \\
&\geqslant \frac{1}{2} - \frac{2}{3}\epsilon.
\end{aligned}
$$

where e is the Euler's constant (note that $e \geq (1 + 1/\iota)^{\iota}$ for any integer ι). According to the above analysis, we conclude that $\mathcal{B}$'s advantage is $\Pr[\mathcal{B}$ is correct$]$-$\frac{1}{2} \geqslant \frac{\epsilon}{6}$. Moreover, the running time of $\mathcal{B}$ is exactly equal to that of $\mathcal{A}$. ■

4.4 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 complete and sound. If RS is strong (or weak) unp-private, then it is strong (or weak) ind-private.* ■

4.5 Ind-Privacy $\not\Longrightarrow$ Unp-privacy

Let $RS = \{R, \mathcal{T}, \mathsf{InitializeR}, \mathsf{InitializeT}, \pi\}$ be any RFID system. We construct a new RFID system $RS' = \{R, \mathcal{T}, \mathsf{InitializeR}, \mathsf{InitializeT}, \pi'\}$ 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 strong (or weak) ind-private, then the RFID system RS' is also strong (or weak) ind-private, but it is not strong (or weak) unp-private.*

Proof. It is easy to see that RS' is strong (or weak) ind-private if RS is strong (or weak) ind-private. We proceed to show that it is not strong or weak unp-private. Since every protocol message of π' is of 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 strong (or weak) unp-private. ■

This theorem indicates that ind-privacy does not imply unp-privacy. In 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.

5. 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 each 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.

5.1 Unp-Privacy $\Longrightarrow$ PRF

Given an RFID system RS with unp-privacy, we show that each tag's computation function $F_{k_{\mathcal{T}_i}}()$ can be used to construct a PRF family. To this end, we first construct a noninteractive protocol by simulating the conversations between the reader and a tag in RS. Then, we define a PRF family via the simulated noninteractive protocol. Note that it is difficult to define a PRF family directly from a tag's outputs of the interactive protocol π in RS since a tag outputs differently in different interrogations even given as input the same challenge message.

Noninteractive Protocol. Given an interactive protocol $\pi(R, T_i)$, one can construct a noninteractive one $\pi'(R, \mathcal{T}_i)$ as follows:

- $\mathcal{T}_i$ sends its key $k_{\mathcal{T}_i}$ and initial state $s^0_{\mathcal{T}_i}$ to the reader R such that the function $F_{k_{\mathcal{T}_i}}()$ originally computed by $\mathcal{T}_i$ can be computed by the reader R.
- The reader R simulates the conversations between the reader R and the tag $\mathcal{T}_i$ in the original protocol.

Obviously, the distribution of the output of the simulated noninteractive protocol $\pi'(R, \mathcal{T}_i)$ is indistinguishable from that of the output of the interactive protocol $\pi(R, \mathcal{T}_i)$. Hence, if the protocol $\pi(R, \mathcal{T}_i)$ is strong (or weak) unp-private, then the noninteractive protocol $\pi'(R, \mathcal{T}_i)$ is also strong (or weak) unp-private.

Without loss of generality, let $P_{CH} = \{0,1\}^{\alpha_1}$, $P_{CN} = \{0,1\}^{\alpha_2}$, and $P_{FT} = \{0,1\}^{\alpha_1+\alpha_2}$, where α_1 and α_2 are two polynomials of κ. For a string $x \in P_{CH} \times P_{CN}$, assume that x can be uniquely represented by $x_C||x_N$ (i.e. $|x_C| = \alpha_1$ and $|x_N| = \alpha_2$), where $x_C \in P_{CH}$ and $x_N \in P_{CN}$.

Given an RFID system $RS = (R, \mathcal{T}, \mathsf{InitializeR}, \mathsf{InitializeT}, \pi)$, we construct a function family $G : \mathcal{K} \times \mathcal{D} \longrightarrow \mathcal{R}$ as follows. At first, choose a tag $\mathcal{T}_i \in_R \mathcal{T}$. Then, construct the following function $J(x)$ by running the simulated noninteractive protocol $\pi'(R, \mathcal{T}_i)$:

1. If the tag $\mathcal{T}_i$ is stateless (i.e. $s^0_{\mathcal{T}_i} = \varepsilon$), then for every $x \in \{0,1\}^{\alpha_1+\alpha_2}$ define $J(x) = F_{k_{\mathcal{T}_i}}(x_C, x_N)$, where $F_{k_{\mathcal{T}_i}}(x_C, x_N) = r_1$ is obtained by running $\pi'(R, \mathcal{T}_i)$ with challenge message x_C and tag's coin toss x_N.

2. If the tag $\mathcal{T}_i$ is stateful (i.e. $s^0_{\mathcal{T}_i} \neq \varepsilon$), define the function $J(x)$ according to the following two cases.

   2.1 If the tag does not toss coins, i.e. $cn_{\mathcal{T}_i} = \varepsilon$ and $\alpha_2 = 0$, for every $c = x \in \{0,1\}^{\alpha_1}$ define
   $$J(x) = F_{k_{\mathcal{T}_i}}(c, s^0_{\mathcal{T}_i}),$$
   where c is the challenge message of the tag $\mathcal{T}_i$.

   2.2 If $cn_{\mathcal{T}_i} \neq \varepsilon$, for every $x \in \{0,1\}^{\alpha_1+\alpha_2}$, define
   $$J(x) = F_{k_{\mathcal{T}_i}}(x_C, x_N, s^0_{\mathcal{T}_i}),$$
   where x_C and x_N are the challenge message and coin toss of $\mathcal{T}_i$, respectively.

Given a tag $\mathcal{T}_i$, it is easy to see that $J(x)$ is a function mapping from $\mathcal{D}$ to $\mathcal{R}$, where $\mathcal{D} = P_{CH} \times P_{CN}$ and $\mathcal{R} = P_{FT}$. Now, a function family $G_\lambda(x) : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$ can be defined as

$$G_\lambda(x) = J(J(\lambda) \oplus x), \qquad (5)$$

where $\lambda \in \mathcal{K} = \{0,1\}^{\alpha_1+\alpha_2}$. 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 $RS = (R, \mathcal{T}, \mathsf{InitializeR}, \mathsf{InitializeT}, \pi)$ is complete, sound, and weak unp-private, then the constructed function family $G : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$ is a PRF family.*

Proof. Here, we only consider the proof for case 1, as the proof for case 2 can be carried out similarly. 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 $F_{k_{\mathcal{T}_i}}()$ cannot be an empty function (i.e. $r_1 \neq \varepsilon$) 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 simulated 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 $\mathcal{K}$. Finally, 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 an advantage at least ϵ and within a time at most t. We construct an algorithm $\mathcal{B}$ which runs T as a subroutine and $(\epsilon, t, j+1, j+1, 0, 0)$-breaks the weak unp-privacy of RS, where j is the number of queries that T asks in the $PTPT$ experiment.

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. Next, $\mathcal{B}$ constructs the noninteractive protocol $\pi'(R, \mathcal{T}_i)$ and selects a random $\lambda \in \mathcal{K}$ and computes $J(\lambda)$. Then, algorithm $\mathcal{B}$ invokes algorithm T with the input function $G_\lambda(\cdot)$ and answers T's queries $(x_1, ..., x_j)$ using function $J(\cdot)$. When algorithm T outputs the *chosen exam* x^* (let $y^* = J(\lambda) \oplus x^*$), algorithm $\mathcal{B}$ sets y^*_C as the *test* message and sets tag $\mathcal{T}_i$'s coin toss in the next interrogation to be y^*_N. Then, it sends $(\mathcal{T}_i, y^*_C)$ to the unp-privacy experiment. Upon receiving the challenge message (r^*, f^*), where $r^* = (r^*_1, cn^*_{\mathcal{T}_i})$, algorithm $\mathcal{B}$ returns r^*_1 to T as an answer to x^*. It is easy to see that if (r^*, f^*) is chosen from the protocol messages then $r^*_1 = F_{k_{\mathcal{T}_i}}(y^*_C, y^*_N) = J(y^*) = G_\lambda(x^*)$. 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. According to the above simulation algorithm, $\mathcal{B}$ 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 (which is at least $\frac{1}{2}+\epsilon$). Hence, the advantage of $\mathcal{B}$ is at least ϵ. Furthermore, it is obvious that the running time of algorithm $\mathcal{B}$ is the same as that of T. ■

5.2 Unp-Privacy $\Longleftarrow$ PRF

Now, we construct an RFID system with strong unp-privacy by implementing a PRF on each tag. Let κ be a security parameter and let κ_1 and κ_2 be two polynomials of κ. Let $F : \{0,1\}^{\kappa_1} \times \{0,1\}^{2\kappa_1} \to \{0,1\}^{\kappa_1}$ be a PRF family. Let $ctr \in \{0,1\}^{\kappa_2}$ be a counter[4] and $\kappa_2 < \kappa_1$. Let pad_1 and pad_2 be two pads such that $|ctr||pad_1| = 2\kappa_1$ and $|ctr||pad_2| = \kappa_1$. The RFID system is constructed as follows.

InitializeR(κ): Setup a reader R with $\sigma = \{F, pad_1, pad_2\}$ according to security parameter κ.

InitializeT(R, κ): When a tag $\mathcal{T}_i$ with identity ID registers to the reader R, choose a key $k \in_R \{0,1\}^{\kappa_1}$ and a counter $ctr = 1$; set the key and the internal state of the tag $\mathcal{T}_i$ to be k and ctr, respectively; compute $I = F_k(ctr||pad_1)$ and store the tuple (I, k, ctr, ID) in a database for the reader.

Protocol $\pi(R, \mathcal{T}_i)$: First, the reader R sends a challenge $c \in_R \{0,1\}^{\kappa_1}$ to the tag $\mathcal{T}_i$. Upon receiving the challenge message c, the tag computes $I = F_k(ctr||pad_1)$ and responds with $r_1||I$, where $r_1 = F_k(c||I) \oplus (ctr||pad_2)$. Then, it updates ctr by increasing 1. Upon receiving the response $r_1||I$, the reader identifies the tag from its database as follows:

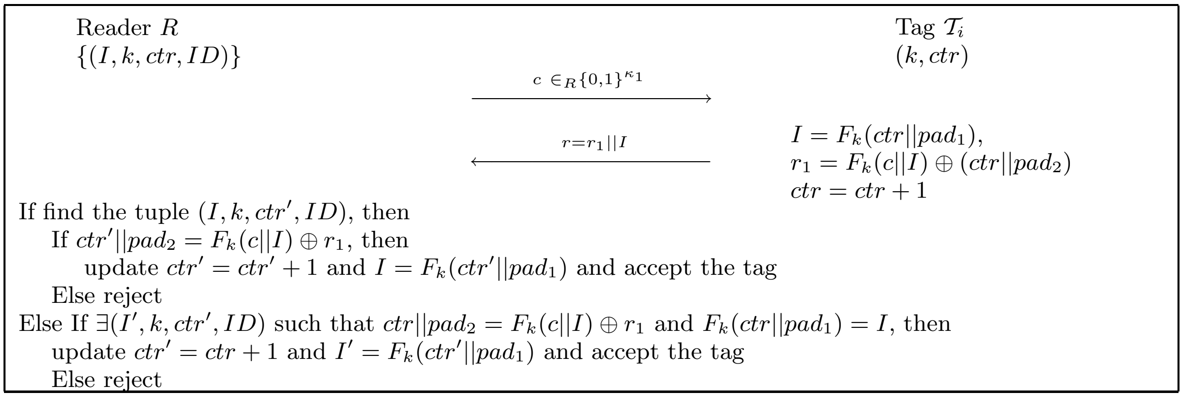
1. (Exact match) The reader searches for the tuple (I, k, ctr', ID) using I as an index in an exact match. If such a tuple exists, the reader computes $F_k(c||I)$ and proceeds as follows:

   1.1 If $ctr'||pad_2 = F_k(c||I) \oplus r_1$, then it updates $ctr' = ctr' + 1$ and $I = F_k(ctr'||pad_1)$ and accepts the tag,

   1.2 Else it rejects the tag.

2. (Exhaustive search) Else the reader looks up for a tuple (I', k, ctr', ID) in an exhaustive search such that $ctr||pad_2 = F_k(c||I) \oplus r_1$ and $F_k(ctr||pad_1) =$

[4] The counter in a tag should not repeat throughout the lifetime of the tag. The size κ_2 of the counter ctr should be large enough so that it is infeasible for an adversary to encounter a repeated protocol message (derived from the same counter value) for the same tag in online attacks (note that offline attacks are thwarted using a long-enough tag secret key). If it takes 0.01 second for each protocol invocation, for example, it would take an adversary at least 348 years to encounter a repeated protocol message for $\kappa_2 = 40$ in online attacks.



```
If find the tuple $(I, k, ctr', ID)$, then
    If $ctr'||pad_2 = F_k(c||I) \oplus r_1$, then
        update $ctr' = ctr' + 1$ and $I = F_k(ctr'||pad_1)$ and accept the tag
    Else reject
Else If $\exists(I', k, ctr', ID)$ such that $ctr||pad_2 = F_k(c||I) \oplus r_1$ and $F_k(ctr||pad_1) = I$, then
    update $ctr' = ctr + 1$ and $I' = F_k(ctr'||pad_1)$ and accept the tag
    Else reject
```

Figure 8: The New RFID Protocol

I. If such a tuple exists, then it updates $ctr' = ctr + 1$ and $I' = F_k(ctr'||pad_1)$ and accepts the tag; else it rejects the tag.

This RFID protocol is shown in Figure 8. Next, we prove that the constructed RFID system is of strong unp-privacy.

Theorem 6. *If the function family $F : \{0,1\}^{\kappa_1} \times \{0,1\}^{2\kappa_1} \to \{0,1\}^{\kappa_1}$ is a PRF family, then the RFID system $RS = (R, \mathcal{T}, \mathsf{InitializeR}, \mathsf{InitializeT}, \pi)$ defined above is of strong unp-privacy.*

Proof. Assume that RS is not strong unp-private. That is, there exists an adversary $\mathcal{A}$ which can $(\epsilon, t, q, s, u, v)$-break the unp-privacy of RS, where $s < 2^{\kappa_2}$. We construct an algorithm $\mathcal{B}$ that can pass the $PTPT$ for the function family F.

On the input of an oracle $\mathcal{O}_F$ of the function $F_k()$, algorithm $\mathcal{B}$ selects a number $n \in_R \{0, 1\}$ and plays the following $Game_n$.

1. Initialize a reader R with $\sigma = \{F, pad_1, pad_2\}$ according to security parameter κ.

2. Select an index i between 1 and ℓ and set 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 $\mathcal{B}$.

3. For $1 \leqslant j \leqslant \ell$ and $j \neq i$, select a random key (index) $k_j \in_R \{0, 1\}^{\kappa_1}$, then set the key and the internal state of the tag $\mathcal{T}_j$ as k_j and $ctr_j = 1$, respectively.

4. If $\mathcal{A}$ asks a query related to tag $\mathcal{T}_i$, $\mathcal{B}$ answers it via oracle $\mathcal{O}_F$.

5. $\mathcal{B}$ can answer $\mathcal{A}$'s queries related to other tags (except $\mathcal{T}_i$) since it knows the keys $k_1, ..., k_{i-1}, k_{i+1}, ..., k_\ell$.

6. When $\mathcal{A}$ outputs the challenge tag $\mathcal{T}_c$ and the *test* message c_0, $\mathcal{B}$ checks whether $c = i$.

7. If $c \neq i$, $\mathcal{B}$ stops.

8. If $c = i$, $\mathcal{B}$ continues the unp-privacy experiment.

    8.1 If $n = 0$, $\mathcal{B}$ submits $(ctr_i||pad_1)$ as the *chosen exam* and receives the response I_i^*, where ctr_i is the current internal state of the tag $\mathcal{T}_i$. Next, it selects $r_1^* \in_R \{0, 1\}^{\kappa_1}$ and returns the pair $(r_1^* \oplus (ctr_i||pad_2), I_i^*)$ to $\mathcal{A}$.

    8.2 If $n = 1$, $\mathcal{B}$ first obtains $I_i^* = F_k(ctr_i||pad_1)$ by querying the oracle $\mathcal{O}_F$. Then, it submits (c_0, I_i^*) as the *chosen exam* and receives the response r_1^*. Finally, it returns $(r_1^* \oplus (ctr_i||pad_2), I_i^*)$ to $\mathcal{A}$.

9. **Output:** When adversary $\mathcal{A}$ outputs a bit b', $\mathcal{B}$ also outputs the bit b'.

Let b denote the random bit in the $PTPT$ experiment. Assuming that the algorithm $\mathcal{B}$ does not stop, we can evaluate its success probability as follows

$$\begin{aligned}
\Pr[\mathcal{B} \text{ succeeds}] &= \frac{1}{2}(\Pr[\mathcal{B} \text{ succeeds in } Game_0] \\
&\qquad + \Pr[\mathcal{B} \text{ succeeds in } Game_1]) \\
&= \frac{1}{2}(\Pr[b' = 0 \wedge b = 0 | n = 0] + \Pr[b' = 1 \wedge b = 1 | n = 0] \\
&\quad + \Pr[b' = 0 \wedge b = 0 | n = 1] + \Pr[b' = 1 \wedge b = 1 | n = 1]) \\
&= \frac{1}{4}(2 + \Pr[b' = 1 | b = 1 \wedge n = 1] \\
&\quad - \Pr[b' = 1 | b = 0 \wedge n = 0]) \\
&\geqslant \frac{1}{2} + \frac{\epsilon}{4}
\end{aligned}$$

Thus, if $\mathcal{A}$ succeeds, algorithm $\mathcal{B}$ also succeeds. The probability that $\mathcal{B}$ does not stop is at least $\frac{1}{\ell}$. Therefore, the advantage of $\mathcal{B}$ is at least $\frac{\epsilon}{4\ell}$. ■

5.3 Minimal Requirement on RFID Tags for Unp-Privacy

Combining Theorems 5 and 6, one can derive the following

Theorem 7. The Minimal Requirement for RFID Unp-Privacy: *An RFID system $RS = (R, \mathcal{T}, \mathsf{InitializeR}, \mathsf{InitializeT}, \pi)$ with strong (or weak) 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 one way function and cryptographically strong pseudorandom generator [11].

This minimal requirement highlights why many lightweight RFID protocols (e.g. [20, 26, 6, 22]) have privacy flaws [29], as these protocols are constructed based on simple operations such as XOR, bit inner product, 16-bit pseudorandom number generator (PRNG), and cyclic redundancy checksum (CRC) without using any computation equivalent to PRF. It also eliminates the need to conduct further research in this direction. However, this minimal requirement does not imply that every RFID system constructed based on PRF or its equivalents is of strong or weak privacy. For example, the RFID systems given in [27, 34, 28] are reported to have privacy vulnerabilities, though they are constructed based on symmetric encryption schemes and/or cryptographic hash functions. How to apply PRF or its equivalents to design an efficient and low-cost RFID system with strong or weak privacy remains an interesting area for further investigation.

The new protocol we provided in Section 5.2 (also see Figure 8) can be considered as an example of such design. While the privacy of this protocol has been proven in Theorem 6, we now analyze its efficiency in terms of tag cost, communication cost, and reader cost. This protocol requires each tag to compute two PRFs in each invocation and store a secret key and a counter value in the tag's memory. A minimum of two rounds of communication is required for identifying each tag. The communication cost in each protocol invocation is constant. In the case that a tag has not been desynchronized[5] since last successful read, our protocol requires a computational cost $O(\log \ell)$ for identifying the tag on the reader side, which is the cost of searching for index I in exact match among ℓ records in the reader's database plus the cost of two PRF computations. In the case that a tag has been desynchronized, our protocol requires exhaustive search $O(\ell)$ in ℓ records for identifying the tag as in most of the existing protocols. 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.

Our protocol is unique in comparison with typical lightweight protocols, including OSK [27], YA-TRAP [34], MSW [23], Hash-Locks [37], Improved Hash-Locks [18], and O-TRAP [5]. In terms of tag computational cost, our protocol is similar to OSK and O-TRAP (which require two hash computations), better than MSW (which requires $O(\log \ell)$ hash computations), but worse than YA-TRAP, Hash-Locks, and Improved Hash-Locks (which require only one hash computation). In terms of tag storage cost, our protocol is similar to YA-TRAP, requiring less storage than O-TRAP ($2\kappa_1$), Hash-Locks ($3\kappa_1$), and MSW ($O(\log \ell)\kappa_1$), but more storage than OSK and Improved Hash-Locks (κ_1), where κ_1 denotes the length of PRF (or its equivalent), reader challenge, or tag secret key.

The communication cost of our protocol is $3\kappa_1$, which is similar to Hash-Locks, Improved Hash-Locks, and O-TRAP, much better than MSW ($O(\log \ell)\kappa_1$), but slightly worse than OSK and YA-TRAP (1 to 2 κ_1). In terms of reader cost, our protocol is among the best (similar to YA-TRAP, Hash-Locks, and O-TRAP) in situations where there is no desynchronization attack. In such case, our protocol only requires searching for an index among ℓ records so as to identify a tag; thus, it is more efficient than MSW, which requires computing $O(\log \ell)$ hash values. In desynchronization attacks, the reader's cost of our protocol is similar to OSK, Improved Hash-Locks, and O-TRAP, as an exhaustive search of a tag key among ℓ records is involved until certain condition is met.

Finally, we note that OSK, YA-TRAP, and Hash-Locks do not offer ind-privacy, while MSW and Improved Hash-Locks offer weak and strong ind-privacy, respectively [18]. Overall, our protocol is among the most efficient protocols with provably strong unp-privacy.

6. CONCLUSION AND OPEN PROBLEM

In this paper, we investigated the relationships between two types of privacy notions for RFID systems. 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. This result also enables us to construct an efficient RFID protocol with low tag cost, communication cost, and reader cost for strong unp-privacy.

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 strong ind-privacy. However, the other direction is uncertain. An open problem is to find the minimal condition for enforcing (strong or weak) ind-privacy in RFID systems. A technical challenge 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.

Acknowledgment: The authors would like to thank Dr. Ari Juels for his helpful comments on an early draft of this paper. This work is partly supported by A*Star SERC Grant No. 082 101 0022 in Singapore. Dr. Ma's work is also partly supported by NSFC under granted No. 60703094 in China.

7. REFERENCES

[1] ISO/IEC-15408 (1999). ISO/IEC-15408 Common Criteria for Information Technology Security Evaluation v2.1. http://csrc.nist.gov/cc, 1999.

[2] G. Ateniese, J. Camenisch, and B. de Medeiros. Untraceable RFID Tags via Insubvertible Encryption. In *Conference on Computer and Communications Security – CCS'05*, pages 92–101, 2005.

[3] G. Avoine. Adversary Model for Radio Frequency Identification. Technical Report LASEC-REPORT-2005-001, Swiss Federal Institute of Technology (EPFL), Security and Cryptography Laboratory (LASEC), 2005.

[4] G. Avoine, E. Dysli, and P. Oechslin. Reducing Time Complexity in RFID Systems. In *Selected Areas in Cryptography – SAC 2005*, 2005.

[5] Mike Burmester, Tri van Le, and Breno de Medeiros. Provably Secure Ubiquitous Systems: Universally Composable RFID Authentication Protocols. In *Conference on Security and Privacy for Emerging*

[5] By "desynchronizing a tag" we mean the counter for the tag in the reader's database is different from the counter in the tag's storage.

Areas in Communication Networks – SecureComm, pages 1–9, 2006.

[6] H.-Y. Chien and C.-H. Chen. Mutual Authentication Protocol for RFID Conforming to EPC Class 1 Generation 2 standards. *Computer Standars and Interfaces, Elsevier Science Publishers*, 29(2):254–259, 2007.

[7] I. Damgård and M. Østergaard. RFID Security: Tradeoffs between Security and Efficiency. In *Topics in Cryptology–CT-RSA 2008*, volume 4964 of *Lecture Notes in Computer Science*, pages 318–332, 2008.

[8] EPCglobal. Class-1 generation-2 UHF RFID protocol for communications at 860 MHz-960 MHz, version 1.0.9. EPC radio-frequency identity protocols (2005), January 2005. www.epcglobalinc.org.

[9] S. Garfinkel, A. Juels, and R. Pappu. RFID Privacy: An Overview of Problems and Proposed Solutions. *IEEE Security and Privacy*, 3(3):34–43, 2005.

[10] O. Goldreich. *The Foundations of Cryptography*, volume I, Basic Tools. Cambridge University Press, 2001.

[11] O. Goldreich, S. Goldwasser, and S. Micali. How to construct random functions. *J. ACM*, 33(4):792–807, 1986.

[12] J. Ha, S. Moon, J. Zhou, and J. Ha. A new formal proof model for RFID location privacy. In *European Symposium on Research in Computer Security (ESORICS) 2008*, volume 5283 of *Lecture Notes in Computer Science*.

[13] Nicholas J. Hopper and Manuel Blum. Secure human identification protocols. In *ASIACRYPT*, pages 52–66, 2001.

[14] A. Juels. Minimalist Cryptography for Low-Cost RFID Tags. In *International Conference on Security in Communication Networks – SCN 2004*.

[15] A. Juels. RFID Security and Privacy: A Research Survey. *IEEE Journal on Selected Areas in Communications*, 24(2):381–394, 2006.

[16] A. Juels, R. Pappu, and B. Parno. Unidirectional key distribution across time and space with applications to RFID security. In *17th USENIX Security Symposium*, pages 75–90, 2008.

[17] A. Juels, R. L. Rivest, and M. Szydlo. The blocker tag: Selective blocking of RFID tags for consumer privacy. In *8th ACM Conference on Computer and Communications Security – ACM CCS*, pages 103–111. ACM Press, 2003.

[18] A. Juels and S. Weis. Defining Strong Privacy for RFID. In *International Conference on Pervasive Computing and Communications – PerCom 2007*.

[19] Ari Juels and Stephen A. Weis. Authenticating pervasive devices with human protocols. In *CRYPTO*, pages 293–308, 2005.

[20] S. Karthikeyan and M. Nesterenko. RFID Security without Extensive Cryptography. In *Workshop on Security of Ad Hoc and Sensor Networks – SASN'05*.

[21] Jonathan Katz and Ji Sun Shin. Parallel and concurrent security of the HB and HB^+ protocols. In *EUROCRYPT*, pages 73–87, 2006.

[22] D. Konidala, Z. Kim, and K. Kim. A Simple and Cost-Effective RFID Tag-Reader Mutual Authentication Scheme. In *Conference on RFID Security 2007*.

[23] D. Molnar, A. Soppera, and D. Wagner. A Scalable, Delegatable Pseudonym Protocol Enabling Ownership Transfer of RFID Tags. In *Selected Areas in Cryptography – SAC 2005*.

[24] D. Molnar and D. Wagner. Privacy and Security in Library RFID: Issues, Practices, and Architectures. In *Conference on Computer and Communications Security – ACM CCS*, 2004.

[25] C. Yu Ng, W. Susilo, Y. Mu, and R. Safavi-Naini. RFID privacy models revisited. In *European Symposium on Research in Computer Security (ESORICS) 2008*, volume 5283 of *Lecture Notes in Computer Science*.

[26] D. Nguyen Duc, J. Park, H. Lee, and K. Kim. Enhancing Security of EPCglobal Gen-2 RFID Tag against Traceability and Cloning. In *Symposium on Cryptography and Information Security 2006*.

[27] M. Ohkubo, K. Suzuki, and S. Kinoshita. Efficient Hash-Chain Based RFID Privacy Protection Scheme. In *International Conference on Ubiquitous Computing – Ubicomp, Workshop Privacy: Current Status and Future Directions*, 2004.

[28] P. L. Pedro, H. C. J. Cesar, M. E. T. Juan, and R. Arturo. RFID Systems: A Survey on Security Threats and Proposed Solutions. In *11th IFIP International Conference on Personal Wireless Communications – PWC'06*.

[29] P. L. Pedro, T. Li, T. Lim, H. C. J. Cesar, and M. E. T. Juan. Vulnerability Analysis of a Mutual Authentication Scheme under the EPC Class-1 Generation-2 Standard. In *Workshop on RFID Security*, 2008.

[30] P. Samarati and L. Sweeney. Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. Technical report, SRI International, 1998.

[31] S. Sarma. Towards the 5 cents Tag. White Paper, Auto-ID Center, 2001. http://www.autoidlabs.org/whitepapers/mit-autoid-wh-006.pdf.

[32] S. Sarma, S. Weis, and D. Engels. Radio-Frequency Identification: Security Risks and Challenges. *Cryptobytes, RSA Laboratories*, 6(1):2–9, 2003.

[33] S. Spiekermann and S. Evdokimov. Privacy Enhancing Technologies for RFID - A Critical State-of-the-Art Report. *IEEE Security and Privacy*, 7(2):56–62, 2009.

[34] G. Tsudik. YA-TRAP: Yet Another Trivial RFID Authentication Protocol. In *International Conference on Pervasive Computing and Communications – PerCom 2006*, pages 640–643, 2006.

[35] T. van Deursen and R. Saša. On a New Formal Proof Model for RFID Location Privacy. Cryptology ePrint Archive, Report 2008/477.

[36] S. Vaudenay. On Privacy Models for RFID. In *Advances in Cryptology - Asiacrypt 2007*.

[37] S. Weis, S. Sarma, R. Rivest, and D. Engels. Security and Privacy Aspects of Low-Cost Radio Frequency Identification Systems. In *International Conference on Security in Pervasive Computing – SPC 2003*.