A Forward Privacy Model for RFID Authentication Protocols

Daisuke Moriyama, Miyako Ohkubo, Shin'ichiro Matsuo

A Forward Privacy Model for RFID Authentication Protocols

Daisuke Moriyama[1] and Miyako Ohkubo[1] and Shin'ichiro Matsuo[1]

NICT, 4-2-1, National Institute of Information and Communication s Technology,
2-2-1 Nukui-Kitamachi, Koganei-shi, Tokyo 184-8795 Japan
{dmoriyam, m.ohkubo, smatsuo}@nict.go.jp

Abstract. In this paper, we propose a new variant of indistinguishability-based security model for the RFID authentication protocol, which allows an adversary to obtain an authentication result and secret key of a target tag. Ng et al. showed that symmetric-key based RFID authentication protocols cannot be resilient to the above information leakage simultaneously in the Paise-Vaudenay security model. We review the existing result and extend the Juels-Weis security model to satisfy these properties by using a suitable restriction. Moreover, we give two example protocols that satisfy the modified security model.

Keywords: RFID, authentication, security model, forward-privacy

1 Introduction

The RFID authentication protocol is an authentication protocol in which an RFID reader communicates with RFID tags and authenticates them. This protocol is considered to be one of the most important techniques for construct the world of "Internet of Things", in which all objects automatically interact with each other with wireless communication [19]. However, many people worry about their privacy with RFID attached objects (e.g., CASPIAN). Therefore, the RFID authentication protocol requires that the identity of RFID tags should be kept secret (anonymity) and any transaction should not be linked (unlinkability), except from legitimate RFID readers.

Since 2003, many RFID authentication protocols have been investigated and several protocols have been shown to be insecure (see [5]). The security of these protocols is evaluated by using a cryptographic security model and one of the goals of the RFID authentication protocol is to satisfy this model. The Paise-Vaudenay security model [17] divides the privacy level into eight categories and these are roughly divided into four groups: whether a malicious adversary can obtain an authentication result

or not (wide/narrow), and whether the internal secret key of the target tag is finally revealed or not (forward/weak). However, Ng et al. classified symmetric-key based RFID authentication protocols into four types and showed that these protocols only satisfy either narrow-forward privacy or wide-weak privacy [8]. In particular, their result showed that authentication protocols which have a key update mechanism and resynchronization property, satisfy only wide-weak privacy, which can be achieved by protocols in which the secret key is always fixed. Coisel and Martin suspected that the provable security level of the SK-based protocol [9] (which does not require any secret key update) and O-FRAP (key update mechanism is clearly shown) are equivalent [5].

In this paper, we propose a variant of the Juels-Weis security model [11] to investigate a suitable security model for RFID authentication protocol. Our model allows an adversary to obtain an authentication result and the secret key of a target tag. Note that we cannot simply send the secret key to the adversary since the resulting model is not achievable. Instead, we add the rule that the reader and the target tags must properly execute a session before the adversary obtains the secret key of the target tag. If the current secret key is appropriately updated in an honest execution, the adversary cannot distinguish which tag communicated with the reader in the past executions. We show that O-FRAP [7] and a variant of the OSK protocol [15] satisfy the modified security model. Furthermore, we show another separation based on the classification in [8] in which the RFID tag can interact with the reader concurrently.

2 Preliminaries

2.1 Notations

We denote a set of k-bit string as $\{0,1\}^k$. 1^k is a k-bit string of 1. $x \xleftarrow{\mathsf{U}} X$ means that variable x is randomly chosen from set X. If f is a probabilistic algorithm, $b \xleftarrow{\mathsf{R}} f(a)$ denotes that the output from f on input a is assigned to b. $\Pr[f(a) \rightarrow b]$ evaluates the probability that the algorithm f outputs b on input a. We say that a probability $P(k)$ is negligible in k if for any polynomials f it holds that $P(k) \leq 1/f(k)$ for sufficient large k.

2.2 Pseudo-Random Generator

Let k be a security parameter. The pseudo-random generator is a deterministic algorithm g that takes as input 1^k and truly random secret $x \in \{0,1\}^k$ and outputs $g(x)$ which is computationally indistinguishable

from random string y ($|g(x)| > |x|$). In this paper, we treat the expansion factor as k and consider pseudo-random generator $g : \{0,1\}^k \to \{0,1\}^{2k}$. The advantage of a probabilistic polynomial time (PPT) adversary $\mathcal{A}$ for pseudo-random generator g is defined by $\mathsf{Adv}^{\mathsf{PRG}}_{\mathcal{A},g}(k) := |\Pr[\mathcal{A}(1^k, g(x)) \to 1 \mid x \xleftarrow{\mathsf{U}} \{0,1\}^k] - \Pr[\mathcal{A}(1^k, y) \to 1 \mid y \xleftarrow{\mathsf{U}} \{0,1\}^{2k}]|$.

Definition 1. *A deterministic algorithm $g : \{0,1\}^k \to \{0,1\}^{2k}$ is pseudo-random generator if for any PPT adversary $\mathcal{A}$, $\mathsf{Adv}^{\mathsf{PRG}}_{\mathcal{A},g}(k)$ is negligible in k.*

We split the above pseudo-random generator g and consider two functions (g_1, g_2) such that the output of these functions are k-bit where $y_1 = g_1(x), y_2 = g_2(x)$ and $y_1 \| y_2 := g(x)$. Let $x_i := g_2^i(x_0)$ be i-round iterated function with input x_0. When we consider $y_i := g_1(x_i)$, Berbain et al. proved that the function $G : \{0,1\}^k \to \{0,1\}^{2kn+k}$ s.t. $(\{(x_i, y_i)\}_{0 \le i \le n}, x_{n+1}) := G(x_0)$ is also the pseudo-random generator [2]. In particular, they showed that for any PPT adversary $\mathcal{A}$, there exists a PPT time algorithm $\mathcal{B}$ such that $\mathsf{Adv}^{\mathsf{PRG}}_{\mathcal{A},G}(k) \le n^2 \cdot \mathsf{Adv}^{\mathsf{PRG}}_{\mathcal{B},g}(k)$.

2.3 Pseudo-Random Function

The pseudo-random function is a function that takes as input a truly random secret (seed) $x \in \{0,1\}^k$ and k-bit string, which output is computationally indistinguishable from truly random function RF. The advantage of an adversary $\mathcal{A}$ against the pseudo-random function $f : \{0,1\}^k \to \{0,1\}^k \to \{0,1\}^k$ is defined as $\mathsf{Adv}^{\mathsf{PRF}}_{\mathcal{A},f}(k) := |\Pr[\mathcal{A}^{f(x,\cdot)}(1^k) \to 1 \mid x \xleftarrow{\mathsf{U}} \{0,1\}^k] - \Pr[\mathcal{A}^{\mathsf{RF}}(1^k) \to 1]|$.

Definition 2. *We say that $f : \{0,1\}^k \to \{0,1\}^k$ is pseudo-random function if for any PPT adversary $\mathcal{A}$, $\mathsf{Adv}^{\mathsf{PRF}}_{\mathcal{A},f}(k)$ is negligible in k.*

3 Security Model for RFID Authentication

The RFID authentication protocol is an authentication protocol between the RFID reader $\mathcal{R}$ and RFID tag $t \in \mathcal{T}$. The RFID reader runs a setup algorithm Setup and generates a public parameter and secret key $(pk, sk) \xleftarrow{\mathsf{R}} \mathsf{Setup}(1^k)$. The reader and each tag share a secret key in the symmetric-key based protocol. After the initialization, the reader and each tag communicate with each other in a wireless setting during the authentication phase. Finally, they output "accept" or "reject" as a result. Following the previous security models [11, 17], we assume that the tag

only performs the sequential session with the reader, though the reader can interact with a lot of tags concurrently. We consider an active adversary as one who can modify any communication message between the reader and tags.

The RFID authentication protocol requires correctness, security and privacy and many security models are provided to formalize them [1, 2, 4, 6, 7, 10, 11, 14, 17, 18]. In this paper, we omit detailed definitions for correctness and security since almost all security models commonly define them. Correctness means that the reader accepts the RFID tag when the session is completed and the communication message is not modified. Security requires that the reader reject the session when the communication is modified by the adversary. In the following, we concentrate on the definition of *privacy* in the security model, which we call the "privacy model".

3.1 Juels-Weis Privacy Model

Juels and Weis introduced an indistinguishability based privacy definition for the RFID authentication protocol in 2007 [10, 11].

Consider the following experiment $\mathsf{Exp}^{\mathsf{IND}\text{-}b}_{\Pi,\mathcal{A}}(k)$ between adversary $\mathcal{A} := (\mathcal{A}_1, \mathcal{A}_2)$ and a challenger in an RFID authentication protocol Π.

$\underline{\mathsf{Exp}^{\mathsf{IND}\text{-}b}_{\Pi,\mathcal{A}}(k)}$
$(pk, sk) \xleftarrow{\mathsf{R}} \mathsf{Setup}(1^k)$;
$(t_0^*, t_1^*, st_1) \xleftarrow{\mathsf{R}} \mathcal{A}_1^{\mathsf{ReaderInit},\mathsf{Send},\mathsf{Corrupt},\mathsf{Result}}(pk, \mathcal{R}, \mathcal{T})$;
$\mathcal{T}' := \mathcal{T} \setminus \{t_0^*, t_1^*\}$;
$b' \xleftarrow{\mathsf{R}} \mathcal{A}_2^{\mathsf{ReaderInit},\mathsf{Send},\mathsf{Corrupt},\mathsf{Result}}(\mathcal{R}, \mathcal{T}', \mathcal{I}(t_b^*), st_1)$;
Output b'

The adversary $\mathcal{A}$ can interact with the reader and many tags in $\mathcal{T}$ by using the oracle query $\{\mathsf{ReaderInit}, \mathsf{Send}, \mathsf{Corrupt}, \mathsf{Result}\}$. $\mathsf{ReaderInit}(1^k)$ activates the session by the reader and $\mathcal{A}$ sends an arbitrary message m by using $\mathsf{Send}(t_i, m)$ $(t_i \in \mathcal{T})$. In addition, the secret key of the tag can be obtained by $\mathsf{Corrupt}(t_i)$ and $\mathsf{Result}(\mathsf{sid})$ outputs the authentication result (namely, accept or reject) of the session sid. When $\mathcal{A}$ outputs two tags (t_0^*, t_1^*) in $\mathcal{T}$, the challenger flips a coin $b \xleftarrow{\mathsf{U}} \{0, 1\}$ and $\mathcal{A}$ interacts with t_b^* anonymously ($\mathcal{I}(t_b^*)$ means the anonymous access to the tag). $\mathcal{A}$ cannot issue $\mathsf{Corrupt}$ query to (t_0^*, t_1^*) in this model. The advantage

of $\mathcal{A}$ in this experiment is defined by $\mathsf{Adv}^{\mathsf{IND}}_{\Pi,\mathcal{A}}(k) = |\Pr[\mathsf{Exp}^{\mathsf{IND\text{-}0}}_{\Pi,\mathcal{A}}(k) \rightarrow 1] - \Pr[\mathsf{Exp}^{\mathsf{IND\text{-}1}}_{\Pi,\mathcal{A}}(k) \rightarrow 1]|$.

Definition 3. *An RFID authentication protocol Π satisfies Juels-Weis privacy model if for any PPT adversary $\mathcal{A}$, $\mathsf{Adv}^{\mathsf{IND}}_{\Pi,\mathcal{A}}(k)$ is negligible in k.*

3.2 Paise-Vaudenay Privacy Model

Paise and Vaudenay defined a simulation based privacy model for the RFID authentication protocol. In this model, the adversary can issue a $\{\mathsf{CreateTag}, \mathsf{DrawTag}, \mathsf{Free}\}$ query in addition to $\{\mathsf{ReaderInit}, \mathsf{Send}, \mathsf{Corrupt}, \mathsf{Result}\}$ in the Juels-Weis privacy model. The $\mathsf{CreateTag}$ query registers a new free tag but this tag still cannot communicate with the reader. Instead, the $\mathsf{DrawTag}$ query converts the free tag into a virtual tag, which can interact with the reader (the adversary can input arbitrary tags and a distribution to transform the tag) and the Free query converts the virtual tag into the original free tag. This model considers a 4×2 matrix for the adversary's capabilities in order to classify the privacy level as follows.

1. For the Result query:
   (a) *Wide* — The adversary can issue the result query.
   (b) *Narrow* — The adversary cannot issue the result query.
2. For the $\mathsf{Corrupt}$ query:
   (a) *Strong* — The adversary can issue the corrupt query at any time.
   (b) *Destructive* — Corrupted tags execute no session.
   (c) *Forward* — The adversary cannot issue any other queries after the corrupt query is sent to a tag.
   (d) *Weak* — The adversary cannot issue the corrupt query.

When we consider $\mathcal{O}_1 := \{\mathsf{CreateTag}, \mathsf{DrawTag}, \mathsf{Free}, \mathsf{Corrupt}\}$ and $\mathcal{O}_2 := \{\mathsf{Launch}, \mathsf{Send}, \mathsf{Result}\}$, wide-strong privacy is described by the following experiment against an RFID authentication protocol Π.

$\underline{\mathsf{Exp}^{\mathsf{SIM\text{-}0}}_{\Pi,\mathcal{A}}(k)}$	$\underline{\mathsf{Exp}^{\mathsf{SIM\text{-}1}}_{\Pi,\mathcal{A},\mathcal{S}}(k)}$
$(pk, sk) \xleftarrow{\mathsf{R}} \mathsf{Setup}(1^k)$;	$(pk, sk) \xleftarrow{\mathsf{R}} \mathsf{Setup}(1^k)$;
$b \xleftarrow{\mathsf{R}} \mathcal{A}^{\mathcal{O}_1,\mathcal{O}_2}(pk, \mathcal{R})$:	$b \xleftarrow{\mathsf{R}} \mathcal{A}^{\mathcal{O}_1,\mathcal{S}(pk)}(pk)$:
Output b	Output b

The adversary $\mathcal{A}$ can access the reader and tags with $\mathcal{O}_2$ directly in the experiment on the left-side, but the right-side experiment requires a simulator $\mathcal{S}$ to simulate its interaction. Note that $\mathcal{S}$ can obtain arbitrary information corresponding to the $\mathcal{O}_1$ query issued by $\mathcal{A}$. The advantage of the adversary is defined by $\mathsf{Adv}^{\mathsf{SIM}}_{\Pi,\mathcal{A},\mathcal{S}}(k) = |Pr[\mathsf{Exp}^{\mathsf{SIM\text{-}0}}_{\Pi,\mathcal{A}}(k) \rightarrow 1] - \Pr[\mathsf{Exp}^{\mathsf{SIM\text{-}1}}_{\Pi,\mathcal{A},\mathcal{S}}(k) \rightarrow 1]|$.

Definition 4. *An RFID authentication protocol Π satisfies the Paise-Vaudenay privacy model if for any PPT adversary $\mathcal{A}$, there exists an algorithm $\mathcal{S}$ such that $\mathsf{Adv}^{\mathsf{SIM}}_{\Pi,\mathcal{A},\mathcal{S}}(k)$ is negligible in k.*

Note that we can define the other privacy levels in a similar way. In particular, we focus mainly on wide-forward privacy and narrow-weak privacy in the Paise-Vaudenay privacy model.

Recently, Moriyama et al. showed several relationships between Juels-Wies and Paise-Vaudenay privacy model [13]. But the strongest privacy level requires public key cryptography [16] and the weakest privacy model does not include any tag's corruption. The motivation of the paper is to investigate a suitable privacy model which a symmetric-key based protocol can satisfy the provably security.

4 Desynchronization Problem in RFID Authentication

In 2009, Ng et al. revisited the Paise-Vaudenay privacy model from the perspective of key synchronization [8]D They classified the symmetric key based RFID authentication protocols on the basis of the key update and showed the limitation of archiving the privacy level in the Paise-Vaudenay privacy model.

Type 0. Any secret key of the protocol is not updated.
Type 1. The tag always updates the secret key whenever the session is executed. The tag does not authenticate the reader in these types of protocols.
Type 2a. After the reader authenticates the tag, the secret key of the tag is updated when the tag authenticates the reader.
Type 2b. Before the reader authenticates the tag, the secret key of the tag is updated. Different from Type 1, the tag authenticates the reader in this type of protocol. If the tag downgrades its secret key when the reader authentication fails (e.g., the tag keeps the early state of the secret key for this property), we call it the "type 2b′ protocol".

Type 0: Type 0 protocols do not support narrow-forward privacy since there is no key update mechanism. Consider the following steps to break narrow-forward privacy: Register two tags (t_0, t_1) with the CreateTag query and generate one virtual tag with the DrawTag query with input (t_0, t_1) and uniformly random distribution. Observe a session between the virtual tag and the reader, convert the virtual tag to the free tag with the Free query and obtain the secret keys with the Corrupt query . Then, the adversary easily checks which tag is communicated to the reader, which

no simulator can do without a probability higher than 1/2.

Type 1: Type 1 protocols do not support wide-weak privacy since the tag always updates its secret key. From its property, this kind of protocol generally defines $q_{\max}$ which is the upper bound for resynchronizing the secret key between the reader and tag. Therefore, Ng et al. described the following steps:

1. Generate two tags (t_0, t_1) with the CreateTag query.
2. Convert t_0 to $vtag$ with the DrawTag query.
3. Issue the SendTag query to $vtag$ to execute $q_{\max} + 1$ sessions, but interfere with the message being sent to the reader after the tag updates its secret key.
4. Convert $vtag$ back to t_0 with the Free query.
5. Generate one virtual tag with the DrawTag query with input (t_0, t_1) and a uniformly random distribution.
6. Execute a session between the virtual tag and the reader normally (the adversary only eavesdrops on the communication).
7. Obtain the authentication result of the above session with the Result query.

When t_0 is chosen by the second DrawTag query, the reader cannot resynchronize with the tag, so it outputs "reject". However, the reader accepts the virtual tag when t_1 is chosen by the query. Since the adversary can learn which tag is chosen, type 1 protocols cannot satisfy wide-weak privacy.

Type 2a: A key update algorithm for the tag is executed only if the tag authenticates the reader in the type 2a protocols. When the adversary modifies the communication messages from the reader and forces the tag to reject all sessions, the tag cannot update its secret key, so it executes all sessions with a fixed secret key. The adversary can learn which tag executes the session when the adversary issues the Corrupt query in the final step of the experiment. Therefore these protocols do not satisfy narrow-forward privacy.

Type 2b and 2b′: In these types, the tag runs the key update algorithm before the reader. Type 2b protocols cannot satisfy wide-weak privacy since the secret key transition for these protocols is equivalent to that for the type 1 protocols, regardless of the reader authentication. When the authentication protocol is classified in type 2b′, the adversary can

fix the secret key of the tag when the communication message from the reader is erased and the adversary sends a random message. Eventually, the privacy level of the type 2b′ protocols is the same as that of the type 2a protocols (narrow-forward privacy failure).

This result shows that any symmetric key based RFID authentication protocol cannot satisfy both wide-weak privacy and narrow-forward privacy in the Paise-Vaudenay privacy model. However, we consider that the privacy level of the type 2a and 2b′ protocols are not equivalent to type 0 protocols. The type 2a and 2b′ protocols clearly specify the key update/downgrade procedure to resynchronize the secret key and provide the notion of "forward-privacy".

5 The Modified Forward Privacy Model

In this section, we illustrate a variant of the Juels-Weis privacy model that allows the adversary to issue the Corrupt query to the challenge tags. Note that the Paise-Vaudenay privacy model does not explicitly define which and when a tag is considered to be the target. In contrast, the Juels-Weis privacy model is easy to modify since the interaction between the challenge tag and adversary is clear. The classification described in Section 4 is generally applied to the RFID authentication protocol. If we purposely add the Corrupt query during the anonymous access phase of this model, no symmetric key based RFID authentication protocol satisfies the modified model because the Result query is given to the adversary.

Instead, if the RFID tag can normally interact with the reader (this means the adversary only eavesdrops on the communication) and resynchronization is completed, the adversary cannot trace which tag interacts with the reader even when the current secret key of the tag is revealed (of course, we assume that the updated key is hard to invert).

Consider the following game between the challenger and adversary $\mathcal{A} := (\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3)$ based on the Juels-Weis privacy model. Let $\pi \xleftarrow{R} \mathsf{Execute}(\mathcal{R}, t)$ be one normal execution of the session between the reader and tag t, and π denotes the communication message.

$\underline{\mathsf{Exp}^{\mathsf{IND}^*\text{-}b}_{\Pi,\mathcal{A}}(k)}$
$(pk, sk) \xleftarrow{R} \mathsf{Setup}(1^k)$;
$(t_0^*, t_1^*, st_1) \xleftarrow{R} \mathcal{A}_1^{\mathsf{ReaderInit},\mathsf{Send},\mathsf{Corrupt},\mathsf{Result}}(pk, \mathcal{R}, \mathcal{T})$;
$\mathcal{T}' := \mathcal{T} \setminus \{t_0^*, t_1^*\}$;

$st_2 \xleftarrow{R} \mathcal{A}_2^{\mathsf{ReaderInit},\mathsf{Send},\mathsf{Result}}(\mathcal{R}, \mathcal{T}', \mathcal{I}(t_b^*), st_1)$;
$\pi_b^* \xleftarrow{R} \mathsf{Execute}(\mathcal{R}, t_b^*), \pi_{1-b}^* \xleftarrow{R} \mathsf{Execute}(\mathcal{R}, t_{1-b}^*)$;
$b' \xleftarrow{R} \mathcal{A}_3^{\mathsf{ReaderInit},\mathsf{Send},\mathsf{Corrupt},\mathsf{Result}}(\mathcal{R}, \mathcal{T}, \pi_b^*, \pi_{1-b}^*, st_2)$:
Output b'

The advantage of the adversary in the modified model is defined by $\mathsf{Adv}_{\Pi,\mathcal{A}}^{\mathsf{IND}^*}(k) = |\Pr[\mathsf{Exp}_{\Pi,\mathcal{A}}^{\mathsf{IND}^*\text{-}0}(k) \to 1] - \Pr[\mathsf{Exp}_{\Pi,\mathcal{A}}^{\mathsf{IND}^*\text{-}1}(k) \to 1]|$.

We add the honest execution which the reader and tag communicate without the adversary's interruption. Instead, the adversary can issue the $\mathsf{Corrupt}$ query to target tags t_0^* and t_1^* after the honest execution from the original Juels-Weis privacy model. If the normal execution *securely* updates the secret key of the reader and tag, this results in the privacy of the challenge tags for the type 2a and 2b′ protocols. The adversary $\mathcal{A}_3$ can obtain the secret key of the target tag and easily break the privacy for type 0 protocols (e.g., SK-based protocol [9, 5]).

Definition 5. *An RFID authentication protocol Π satisfies the modified Juels-Weis privacy model if for any PPT adversary $\mathcal{A}$, $\mathsf{Adv}_{\Pi,\mathcal{A}}^{\mathsf{IND}^*}(k)$ is negligible in k.*

In this paper, we only concentrate on the modification of the Juels-Weis privacy model because the simulation-based privacy model does not explicitly define which is the actual target from the malicious adversary. So it is difficult to insert the honest execution to the Paise-Vaudenay privacy model Note that if we omit the honest execution in the modified model, we can easily describe an attack scenario for any symmetric key based RFID authentication protocols as in Section 4. Also note that the honest communication between the reader and tag must be executed *after* the anonymous communication phase is finished. Billet, Etrog and Gilbert formalized the forward privacy model such that the honest execution occurs before the anonymous communication [3]. However, their model is not achievable since such a honest execution is useless and does not prevent any of the attacks described in Section 4 (Specifically, their proposed protocol PEPS falls into a type 2a protocol and does not satisfy their privacy model).

6 Suitable RFID Authentication Protocols

6.1 O-FRAP

O-FRAP is an RFID authentication protocol proposed by Li, Burmester and de Medeiros in 2007 [7]. This protocol falls into type 2a and does not

hold narrow-forward privacy [17]. In this paper, we show that O-FRAP satisfies the modified privacy model. Remark that O-FRAP satisfies universally composable (UC) security, but [7] does not provide any relationship between their model and other security models. So the adequateness of the UC definition and its protocol described in [7] is not investigated. Our result shows that a UC-secure protocol also holds provable security in the indistinguishability-based privacy model.

Let k be a security parameter, $f : \{0,1\}^k \times \{0,1\}^{2k} \to \{0,1\}^{4k}$ be pseudo-random function and ℓ be the total number of tags in the protocol. Each tag contains its secret key sk_i and nonce r_2. The reader keeps its database $\{sk_i, sk_{i.old}\}_{1 \le i \le \ell}$ which contains the current and previous secret keys of each tag. The reader and a tag t_i execute the following authentication.

1. The reader chooses $r_1 \xleftarrow{\cup} \{0,1\}^k$ and sends it to the tag.
2. When the tag t_i obtains r_1, it computes $(r_{temp}, s_2, s_3, sk_i') := f(sk_i, r_1 \| r_2)$ and sends (r_2, s_2) to the reader. Then the nonce r_2 is updated as $r_2 := r_{temp}$.
3. Upon receiving (r_2, s_2), the reader performs the following:
   - Compute $(\cdot, s_2'', s_3'', sk_i'') := f(sk_i, r_1 \| r_2)$ for $1 \le i \le \ell$ and check if $s_2'' = s_2$. If so, set $sk_{i.old} := sk_i, sk_i := sk_i'', s_3' := s_3''$ and accept the tag.
   - If $s_2'' = s_2$ where $(\cdot, s_2'', s_3'', sk_i'') := f(sk_{i.old}, r_1 \| r_2)$ for some $1 \le i \le \ell$, the tag sets $sk_i := sk_i'', s_3' := s_3''$ and accepts the tag.
   - Otherwise, select $s_3' \xleftarrow{\cup} \{0,1\}^k$ and reject all tags.

   Finally, the reader sends s_3' to the tag[1].
4. When the tag receives s_3', it checks whether $s_3' = s_3$. If the equation holds, then the tag updates the secret key as $sk_i := sk_i'$ and accepts the reader. Otherwise, the tag outputs "reject".

The main building block of this protocol is the challenge-response authentication with the pseudo-random function f. In addition, the reader keeps the previous secret key of each tag to resynchronize the secret key to the tag when the desynchronization occurs.

Theorem 1. *Assume that f is a pseudo-random function. Then, O-FRAP satisfies the modified Juels-Weis privacy model.*

[1] We slightly modify O-FRAP so that the reader does not abort the session and output a random variable even when the search procedure is failed. Otherwise, the Result query becomes meaningless since the adversary can trivially know the authentication result.

Proof. Let q be a upper bound by which the RFID tag performs the key update procedure. Note that q is bounded by the adversarial oracle query and it is at most polynomial in k. S_i is the event that the adversary outputs 1 in Game i.

Game 0. This is the original game between the challenger and adversary in the modified Juels-Weis privacy model.
Game 1-(i,j). We gradually change the output of all the sessions executed in this game:

1. For any session executed between the reader and tag $t_{i'}$ $(t' < i)$, the output of the pseudo-random function is changed with a uniformly random string over $\{0,1\}^{4k}$.
2. For i-th tag t_i,
   2-1. When the number of key update is less than j $(j' < j)$, the output of the pseudo-random function is changed with a uniformly random string $(r_{temp}, s_2, s_3, sk'_{j'}) \xleftarrow{\mathsf{U}} \{0,1\}^{4k}$.
   2-2. If the number of key update j' holds $j' \geq j$, the output of the session is computed by the pseudo-random function f.
3. For any session executed between the reader and tag $t_{i'}$ $(i' > i)$, all output and update keys are computed by the pseudo-random function.

Lemma 1. $\Pr[S_0] = \Pr[S_{1\text{-}(1,0)}]$.

This transformation is purely conceptual and there is no difference between these games.

Lemma 2. *For all* $1 \leq i \leq \ell$ *and* $0 \leq j \leq q$*, there exists an algorithm* $\mathcal{B}$ *for pseudo-random function* f *such that* $|\Pr[S_{1\text{-}(i,j)}] - \Pr[S_{1\text{-}(i,j+1)}]| \leq \mathsf{Adv}^{\mathsf{PRF}}_{\mathcal{B},f}(1^k)$.

Proof. If the adversary can distinguish the difference between Game 1-(i,j) and Game 1-$(i,j+1)$ with non-negligible probability, then there exists an algorithm $\mathcal{B}$ that breaks the security of the pseudo-random function. The only difference between these games is that the outputs come from the pseudo-random function or truly random string when the tag t_i executes the sessions with j-th updated key.

The oracle queries which algorithm $\mathcal{B}$ can issue is either the pseudo-random function $f(sk_i, \cdot)$ or truly random function RF. $\mathcal{B}$ generates secret key of the tag except t_i in the set up and simulates the session between the reader and tags. If the number of key update on t_i is less than j, we set uniformly random string over $\{0,1\}^{4k}$ as the output of the session.

Similarly, if the key update on t_i is executed more than j, we computes the output with the pseudo-random function. If the number of key update on t_i is j, $\mathcal{B}$ computes the output as follows. When $\mathcal{A}$ issues $\mathsf{ReaderInit}$, $\mathcal{B}$ generates $r_1 \xleftarrow{\mathsf{U}} \{0,1\}^k$ and sends r_1 to $\mathcal{A}$. Upon receiving $\mathsf{SendTag}(t_i, r'_1)$ from the adversary, $\mathcal{B}$ chooses $r_2 \xleftarrow{\mathsf{U}} \{0,1\}^k$ and issues $r'_1 \| r_2$ to the oracle query. The output of the tag in this session consists of the response from the oracle $(r_{temp}, s_2, s_3, sk'_{j'}) \in \{0,1\}^{4k}$. When the adversary sends (r_1, r'_2, s'_2) to the reader, $\mathcal{B}$ issues $r_1 \| r'_2$ to the oracle and verifies the message as the protocol specification. If the verification is accepted, $\mathcal{B}$ set s_3 as the output of the reader. Otherwise, the output of the reader is set as $s_3 \xleftarrow{\mathsf{U}} \{0,1\}^k$. Finally, when $\mathcal{A}$ outputs a bit b, $\mathcal{B}$ outputs the same bit.

If the pseudo-random function is given to $\mathcal{B}$, the above game is equivalent to Game 1-(i,j) from the view point of the adversary $\mathcal{A}$. Otherwise, the distribution of the message is the same as Game 1-$(i, j+1)$. Therefore we have $|\Pr[S_{1\text{-}(i,j)}] - \Pr[S_{1\text{-}(i,j+1)}]| \leq \mathsf{Adv}^{\mathsf{PRF}}_{\mathcal{B},f}(1^k)$.

Lemma 3. *For any* $1 \leq i \leq \ell$, *we have* $\Pr[S_{1\text{-}(i,q+1)}] = \Pr[S_{1\text{-}(i+1,0)}]$.

It is clear that 3 hold since the game transformation between them is purely conceptual (the distribution is equivalent).

From these lemmas, we can transform Game 0 to Game 1-$(\ell+1, 0)$. This time, all the output of the session in the last game is independently chosen and truly random. That is, the probability that $\mathcal{A}$ can guess which tag is selected in the anonymous communication phase is $1/2$. Thanks to $\mathsf{Execute}$, the secret key of the tag is completely updated and its secret key is uniformly chosen and independent from the previous sessions. Therefore the adversary can obtain no information about the past session of the tag.

Finally, we obtain $\mathsf{Adv}^{\mathsf{IND}^*}_{\Pi,\mathcal{A}}(k) \leq \ell(q+1) \cdot \mathsf{Adv}^{\mathsf{PRF}}_{\mathcal{B}}(1^k)$.

6.2 The Modified OSK Protocol

The OSK Protocol is an RFID authentication protocol provided by Ohkubo et al. in 2003 and this is one of the type 1 protocols [15]. In this paper, we add the reader authentication and the roll back property for the secret key to the OSK protocol. We show that the modified protocol satisfies IND*-privacy.

Consider that k is a security parameter, $g : \{0,1\}^k \to \{0,1\}^k \times \{0,1\}^k$ is a pseudo-random generator and $f : \{0,1\}^k \times \{0,1\}^k \to \{0,1\}^{2k}$ is a pseudo-random function. The reader keeps $\{sk_i\}_i$ and each tag contains its secret key sk'_i where $(\cdot, sk'_i) = g(sk_i)$.

1. The reader chooses $r_1 \xleftarrow{\cup} \{0,1\}^k$ and sends it to the tag.
2. When the tag t_i obtains r_1, it performs the following:
   (a) Compute $(u_1', u_2') := g(sk_i')$.
   (b) Select $r_2 \xleftarrow{\cup} \{0,1\}^k$ and compute $s_2 := f(u_1', r_1 \| r_2)$.
   (c) Set $sk_{i.temp} := sk_i$ and $sk_i := u_2'$.
   (d) Send (r_2, s_2) to the reader.
3. Upon receiving (r_2, s_2), the reader computes $(u_1, u_2) := g(sk_i)$ for $1 \leq i \leq \ell$ and performs the following:
   – Compute $(u_1'', u_2'') := g(u_2)$ and check $s_2 = f(u_1'', r_1 \| r_2)$. If so, compute $s_3' := f(u_1'', r_2 \| r_1)$, set $sk_i := u_2$ and accept the tag.
   – If $s_2 = f(u_1, r_1 \| r_2)$ compute $s_3' := f(u_1, r_2 \| r_1)$ and accept the tag.
   – Otherwise, select $s_3' \xleftarrow{\cup} \{0,1\}^k$ and reject all tags.

   Finally, the reader sends s_3' to the tag.
4. When the tag receives s_3', it checks whether $s_3' = f(sk_{i.temp}, r_2 \| r_1)$ or $s_3' = f(sk_i, r_2 \| r_1)$. If either equation holds, then the tag accepts the reader. Otherwise, the tag sets $sk_i := sk_{i.temp}$ and output "reject".

Similar to the original OSK protocol, the updated secret key is deterministically defined by the current secret key. Thus the reader can precompute this value if the reader has enough resources. Therefore the reader can find the tag's identity from its database.

Theorem 2. *If g is a pseudo-random generator and f is a pseudo-random function, the above protocol satisfies the modified Juels-Weis privacy model.*

Proof. Let q be an upper bound by which the RFID tag performs the key update procedure. Note that q is bounded by the adversarial oracle query and it is at most polynomial in k. S_i is the event that the adversary outputs 1 in Game i.

Game 0. This is the original attack game in the modified Juels-Weis privacy model.
Game 1-*i*. We gradually change the output of the pseudo-random generator g as follows:

1. For any session executed between the reader and tag $t_{i'}$ $(t' < i)$, the output of the pseudo-random generator g is changed with a uniformly random string over $\{0,1\}^{2k}$.
2. For any session executed between the reader and tag $t_{i'}$ $(t' \geq i)$, all output and update keys are computed by the pseudo-random generator.

Game 2-(i, j). We modify the output of the pseudo-random function f in the same fashion as O-FRAP.

Lemma 4. *We have* $\Pr[S_0] = \Pr[S_{1\text{-}0}]$.

This change is purely conceptual and it is clear that $\Pr[S_0] = \Pr[S_{1\text{-}0}]$.

Lemma 5. *For any* $0 \leq i \leq \ell - 1$*, there exists an algorithm* $\mathcal{B}$ *such that* $|S_{1\text{-}i} - S_{1\text{-}(i+1)}| \leq \mathsf{Adv}^{\mathsf{PRG}}_{\mathcal{B},G}(k)$.

Proof. If the adversary can distinguish between Game 1-i and Game 1-$(i+1)$, we construct an algorithm $\mathcal{B}$ that breaks the security of the pseudo-random generator G (see Section 2.3).

$\mathcal{B}$ obtains $\delta := (\{x_i, y_i\}_{0 \leq i \leq q}, x_{q+1})$, where $\delta := G(x_1)$ or $\delta \xleftarrow{\mathsf{U}} \{0,1\}^{nk+k}$. $\mathcal{B}$ computes all the secret keys of the tag except t_{i+1} and simulates the session as Game 1-i. The initial secret key of the tag t_i is set as x_1 and the reader's secret key corresponding to the tag is x_0. $\mathcal{B}$ sets (u_1, u_2) as $(u_1, u_2) := (y_{j+1}, x_{j+2})$ when t_i updated the secret key j times. When $\mathcal{A}$ outputs a bit b', $\mathcal{B}$ outputs the same bit.

If $\mathcal{B}$ is given pseudo-random variables, then the above game is equivalent to Game 1-i from the view point of the adversary. Otherwise, the distribution of the above game is the same as Game 1-$(i+1)$. Therefore we have $|S_{1\text{-}i} - S_{1\text{-}(i+1)}| \leq \mathsf{Adv}^{\mathsf{PRG}}_{\mathcal{B},G}(k)$.

Lemma 6. *We have* $\Pr[S_{1\text{-}\ell}] = \Pr[S_{2\text{-}(1,0)}]$.

Lemma 7. *For any* $1 \leq i \leq \ell$*, there exists an algorithm* $\mathcal{B}$ *such that* $|\Pr[S_{1\text{-}(i,j)}] - \Pr[S_{1\text{-}(i,j+1)}]| \leq \mathsf{Adv}^{\mathsf{PRF}}_{\mathcal{B},f}(1^k)$ *(*$0 \leq j \leq q$*).*

Lemma 8. *For any* $1 \leq i \leq \ell$*, we have* $\Pr[S_{1\text{-}(i,q+1)}] = \Pr[S_{1\text{-}(i+1,0)}]$.

It is clear that Lemma 4 and 6 hold since these game transformations are purely conceptual. We can easily prove Lemma 7 and 8 on the basis of the proof for Lemma 2 and Lemma 3 in a similar fashion.

Finally, we obtain

$$\mathsf{Adv}^{\mathsf{IND}^*}_{\mathcal{A},\Pi}(k) \leq \ell \cdot \mathsf{Adv}^{\mathsf{PRG}}_{\mathcal{B},G}(k) + \ell(q+1) \cdot \mathsf{Adv}^{\mathsf{PRF}}_{\mathcal{B},f}(k).$$

7 Further Separation in Concurrent Execution

The above privacy definition and other previous privacy models [11, 17, 18] assume that the RFID tag runs only one session at a time (e.g., sequential execution). However, an adversary can issue many oracle queries and

interrupt the communication, so the tag may receive the message to start a new session during the execution of another session. Thus we consider the case that the RFID tag can perform the concurrent execution in this section. Of course, this situation may not practical since the resource of cheap tags is limited. Nonetheless, this situation is useful to show the gap between the type 2a and 2b$'$ protocols.

While these two types of protocols contain the key update mechanism and the secret key of the tag is synchronized to the reader, the secret key input to the computation of the output message on the tag is different in the concurrent setting. Consider that (m_0, m_1) is the output from the tag in the concurrent execution. In the type 2a protocols, both messages are computed with fixed secret key since the tag does not update the secret key before the reader authentication. Even when the adversary obtains an updated secret key, it is difficult to distinguish the challenge tag from these messages. On the other hand, the secret key is always updated and m_1 is computed by the updated secret key in type 2b$'$ protocols. If the adversary responds with a random message to the tag in the concurrent session, the secret key of the tag is roll backed and the adversary obtains the secret key which is used to compute m_1. Therefore the adversary can distinguish which tag is selected in the anonymous communication phase in the type 2b$'$ protocols if we consider the concurrent setting.

8 Concluding Remarks

In this paper, we proposed a new variant of Juels-Wies privacy model that allows an adversary to issue the result and corrupt queries on the basis of the Juels-Weis privacy model. The RFID tag is quite cheap device and it is hard to implement secure module (e.g., Trusted Platform Module). Thus the secret key leakage is critical issue for RFID authentication protocol. Independently, we can observe the authentication result in many situations (automatic ticket gate, entrance of the private sector, etc). Though Ng et al. showed the separation result described in Section 4, there is a achievable security model that the adversary can obtain these information. We showed two examples and provide concrete security proof for these protocols.

References

1. Akgün, M., Çağlayan, M.U.: Extending an RFID security and privacy model by considering forward untraceability. In: STM 2010. LNCS, vol. 6710, pp. 239–254. Springer Heidelberg (2011)

2. Berbain, C., Billet, O., Etrog, J., Gilbert, H.: An efficient forward private RFID protocol. In: ACMCCS 2009. pp. 43–53. ACM (2009)
3. Billet, O., Etrog, J., Gilbert, H.: Lightweight privacy preserving authentication for RFID using a stream cipher. In: FSE 2010. LNCS, vol. 6147, pp. 55–74. Springer Heidelberg (2010)
4. Burmester, M., Le, T.V., Medeiros, B.D., Tsudik, G.: Universally composable RFID identification and authentication protocols. ACM TISSEC, vol. 12, No. 4 (21). ACM (2009)
5. Coisel, I., Martin, T.: Untangling RFID privacy models. ePrint Archive, 2011/636 (2011)
6. Hermans, J., Pashalidis, A., Vercauteren, F., Preneel, B.: A new RFID privacy model. In: ESORICS 2011. LNCS, vol. 6879, pp. 568–587. Springer Heidelberg (2011)
7. Le, T.V., Burmester, M., Medeiros, B.D.: Universally composable and forward-secure RFID authentication and authenticated key exchange. In: ASIACCS 2007. pp. 242–252. ACM (2007)
8. Ng, C.Y., Susilo, W., Mu, Y., Safavi, R.: New privacy results on synchronized RFID authentication protocols against tag tracing. In: ESORICS 2009. LNCS, vol. 5789, pp. 321–336. Springer Heidelberg (2009)
9. International organization for standardization. ISO/IEC 9798: Information technology – Security techniques – Entity authentication, 1991-2010.
10. Juels, A., Weis, S.A.: Defining strong privacy for RFID. In: PerCom 2007. pp. 342–347. IEEE (2007)
11. Juels, A., Weis, S.A.: Defining strong privacy for RFID. ACM TISSEC, vol. 12, No. 1 (7). ACM (2009)
12. Lim, C.H., Kwon, T.: Strong and robust RFID authentication enabling perfect ownership transfer. In: ICICS 2006. LNCS, vol. 4307, pp. 1–20. Springer Heidelberg (2006)
13. Moriyama, D., Matsuo, S., Ohkubo, M.: Relations among notions of privacy for RFID authentication protocols. In: ESORICS 2012. LNCS, vol. 7459, pp.661–678. Springer Heidelberg (2012)
14. Ouafi, K., Phan, R.C.W.: Traceable privacy of recent provably-secure RFID protocols. In: ACNS 2008. LNCS, vol. 5037, pp. 479–489. Springer Heidelberg (2008)
15. Ohkubo, M., Suzuki, K., Kinoshita, S.: Cryptographic approach to privacy-friendly tags. RFID Privacy Workshop (2003)
16. Ouafi, K., Vaudenay, S.: Strong privacy for RFID systems from plaintext-aware encryption. In: CANS 2012. LNCS, vol. 7712, pp. 247–262. Springer Heidelberg (2012)
17. Paise, R., Vaudenay, S.: Mutual authentication in RFID: security and privacy. In: ASIACCS 2008. pp. 292–299. ACM (2008)
18. Vaudenay, S.: On privacy models for RFID. In: ASIACRYPT 2007. LNCS, vol. 4833, pp. 519–535. Springer Heidelberg (2007)
19. Welbourne, E., Battle, L., Cole, G., Gould, K., Rector, K., Raymer, S., Balazinska, M., Borriello, G. Building the internet of things using RFID: The RFID ecosystem experience. IEEE Internet Computing. IEEE (2009)