Leakage-Resilient RFID Authentication with Forward-Privacy

Shin'ichiro Matsuo[1], Le Trieu Phong[1], Miyako Ohkubo[1], and Moti Yung[2,3]

[1] National Institute of Information and Communications Technology (NICT), Japan
[2] Columbia University
[3] Google Inc.

Abstract. Low power devices, such as smart-card and RFID-tags, will be used around our life including in commercial and financial activities. A prime application of such devices is entity authentication in pervasive environment. The obvious concerns in this environment involves getting security against tag-forgery (even by adversary controlled readers) and, on the other hand, giving users privacy against linking of different authentication transcripts. Many cryptographic protocols have realizes such requirements. However, there is no scheme which realizes, both, forward-privacy and tag-forgery right after some leakage is occurred. Since some devices among the huge quantity of expected devices will surely be compromised, it seems highly important, from an engineering point of view, to deal with limited damage of such exposures. In this paper, we address the gap by proposing the first RFID scheme that realizes both requirements.

Keywords: RFID authentication, leakage-resilience and forward-privacy.

1 Introduction

1.1 Background

In coming years, more and more devices are going to be put at the hand of consumers, and are and will be used for authentication (smartcards, RFID-tags, etc.) for applications combining cyber as well as the physical world like point-of-sale authentication during shopping. This new technology poses increasingly important security and privacy issues. In these environments, cryptographic authentication protocols are used by users holding devices (e.g. mobile phones) with smart-cards and RFID-tags. They are also used for many services such as digital cash, transportation card and key-less entry system. As a consequence, these protocols become fundamentals of our activities and physical security.

In this integrated world, a typical inter-collaboration is performed between servers with huge computational power and a huge number of low-end devices. The weakest link in this environment is the low-end devices and it is crucial to provide security to the device and simultaneously privacy to its user, and mitigate properly security failure of some devices. Thus, main considerations

S.B. Ors Yalcin (Ed.): RFIDSec 2010, LNCS 6370, pp. 176–188, 2010.

in RFID-tag authentication schemes are achieving tag-unforgeability as well as forward-privacy using modest computational resources. For authentication purpose, we must prevent forgery (where a forged tag is authenticated as valid). This requirement is general for authentication protocol.

Furthermore, a special requirement for RFID authentication, since it indicates location of the tag in the real world, is preserving privacy. In particular, two types of privacy are considered in previous researches. The first one is personal information disclosure. Namely, obtaining identifier from the tag (for example, we can obtain names of goods, amount of money a person posses and the name of a person, if the tag records and give them as answers for tag-reader). This privacy issue can be prevented by encrypting the data by using secure cryptographic algorithm. However, if we use deterministic algorithm for such encryption, the second type of privacy issue, that of tracing, occurs. If an adversary reads the same tag at two different times, he can trace activity of the owner of the tag by linking different protocol messages sent from the tag. Thus, "unlinkability" is also required for RFID-tag authentication protocol.

Let us elaborate on the intuition of unlinkability. Namely, it ensures past protocol messages are kept unlinkable *even when* the tag is corrupted and its internal state is given to the adversary. This is the standard "forward privacy" notion, first introduced by Ohkubo, Suzuki and Kinoshita in [20]. The OSK scheme was analysed in the random oracle model where hash functions are treated as truly random ones. We note that forward privacy is a stringent notion, so there are only a few schemes satisfying it, among which let us mention the works [4,3].

Another important concern is about the limited computational power of RFID tags. Generally, the tags have relatively limited capability of computation compared to personal computers and smart-phones. The limitation is inherent from their gate sizes and power supply, which is of greatly small amount. As a consequence, it is difficult for RFID-tags to perform public-key cryptography which requires modulo exponentiations. Also, it is not easy to execute cryptographic algorithms in RFID-tags. We therefore assume that allowed computations for the RFID-tags are XOR operations, small stream ciphers, small block ciphers and resulting hash functions (for example, Bogdanov et al [5] showed that 128-bit output hash functions can be implemented by 4,000 gates from the PRESENT block cipher).

1.2 Why Leakage-Resilient for RFID-World

As described above, in forward privacy, we allow the adversary to obtain the full internal state of the tag, denoted as "full leakage". Surely, we must consider this type of attack as the worst case. It is worth noting that, in the real usage of RFID-tags, it apparently takes quite much time and effort to conduct attacks leading to full leakage. (For example, the adversary steals the tag and brings it to his laboratory to obtain the internal state.) However, the adversary certainly has no chance to give the tag to the original owner again. Therefore, we can limit the number of full leakage to only one time.

In this paper, we additionally consider an attack scenario which we call "multi-time partial leakage". Namely, in the life time of a tag, its internal state may be partially leaked in a gradual way. It is obvious that partial leakage is more likely to occur than full leakage, because the adversary can conduct such attacks in a shorter time, with cheap and small-size devices. Furthermore, the adversary has enough time to bring back the tag to the original owner. Therefore, it is practical to consider the multi-time partial-leakage scenarios. However, despite many works on RFID in the literature, there is no scheme which is provably secure against general side channel attacks (see, e.g., [11] for an extensive list) which cause both partial leakage and full leakage of tags' state. Namely, there is not yet any work considering the situation where some information (say some bits, or the Hamming weight) of the tag state is leaked to the adversary. The goal of this work is to fill the gap by constructing a leakage-resilient RFID protocol.

The above discussion focused on forward privacy, and later on, we will formalize the notion of leakage-resilient forward privacy. We also do the same with tag unforgeability, a (known) notion ensuring that no-one except the tag can make the reader output OK. Namely, we formalize the notion of leakage-resilient tag unforgeability, assuring that even the internal state is partially leaked, no-one is able to make the reader output OK.

1.3 Our Contribution

In this paper, we propose the first leakage-resilient RFID authentication protocol, which fulfills rigorously both forward-privacy and security. Our security analysis is simple, and is in the standard model. Our proposal is also very modest in tag computation, in which the tag only needs to compute two PRFs (e.g., AES). We use the recent stream cipher of Pietrzak [21] as the main building block. We compare our proposal with some schemes with forward privacy in Table 1.

Table 1. A comparison of schemes with forward privacy

Schemes	Provable Security on Leakage Resiliency	Security model	Ingredient
OSK [20]	Only privacy against full-leakage	Random Oracle	2 random oracles
Berbain et al [3]	Only privacy against full-leakage	Standard	1 PRNG + 1 universal hash
Burmester et al [7]	Only privacy against full-leakage	Standard	5 PRNGs
Our proposal	Privacy against full-leakge and past partial-leakage Tag-unforgeability against partial-leakage	Standard	1 PRF + 1 wPRF

Above, PRNG = psuedo-random number generator, (w)PRF = (weak) psuedo-random function.

Organization of this paper is as follows. In Section 2, we define system model of RFID authentication protocol and present some definitions of leakage-resilient security and privacy suitable for RFID authentication. Then, we show our proposal in Section 3. We conclude this paper in Section 4.

1.4 Related Works

Since Juels et al pointed out privacy issue in RFID authentication protocol [14], many RFID authentication protocol studies have been conducted, such as [20,19,25,13,2]. Most protocols are based on hash functions, some scheme uses pseudo-random functions and pseudo-random number generator instead of hash functions [6,4,16,3]. There are three major RFID authentication schemes related to our result.

The first scheme which realizes "forward-privacy" against leakage of internal state was proposed by Ohkubo et al [20]. This protocol uses a hash-chain constructed by one-way hash function and random oracle for processing protocol message. The authenticity and indistinguishability can be proven in random oracle model. Forward privacy is mainly based on one-way function. Roughly speaking, the proof involves creating an adversary who breaks the one-wayness by using another adversary who breaches the forward privacy of this scheme.

The second scheme realizes forward-privacy in the standard model. Berbain et al in [3] proposed the first scheme in the standard model. The basic idea of the scheme is same as OSK protocol, however the chain for one-wayness is constructed by using pseudo-random number generator and the function for processing protocol message is realized by universal hash function and challenge-and-response protocols.

Let us also mention the recent work of Burmester and Munila [7]. They proposed a protocol, using pseudo-random functions, which has forward-privacy and tag unforgeability in the universally composable setting. However, the scheme assumes certain refresh operations *external* to the tag, which seems hard to be easily realized. Namely, the scheme security is based on periodically updating the random number generators with fresh randomness, which seems required some trusted device to handle the job.

This paper belongs to the so-called leakage-resilient cryptography, aiming at preventing side-channel attacks, and is a very current area of research. In ordinary cryptographic research, the security model does not consider leakage of secret information. In the symmetric world, Petit et al. [23] proposed a leakage-resilient pseudo-random generator from ideal ciphers. Dziembowski et al [10] proposed a leakage-resilient stream cipher based on pseudo-random generator in the standard model. Then Pietrzak [21] proposed simplified leakage-resilient stream cipher from wPRF. We will use the same model of leakage as [10,21] in this paper.

2 Model and Security Definitions

First, let us show the system model of RFID authentication. There exist three types of entities in this authentication protocol: a tag, a reader, and an authentication server. The functions and conditions for each entity are as follows:

Tag: We assume that the RFID tag $\mathcal{T}$ is a passive tag. It can operate only when interrogated by a reader and only for a short time. The most important limitation is computational power. Each tag can perform only basic cryptographic calculations: hash functions, pseudo-random number generation and symmetric encryption, as well as simple XOR calculations. It is not tamper-proofed. An adversary can obtain some of (or all) the information stored in the tag, for example via side channel attacks.

Reader: A reader communicates with each tag and the authentication server. The reader acts as an intermediary between the tag and the authentication server. It does not retain any secret information or execute any cryptographic operation.

Authentication server: An authentication server $\mathcal{S}$ is used to evaluate the correctness of $\mathcal{T}$ upon receiving protocol messages from $\mathcal{T}$. The authentication server has huge computational power and storage and can be used to carry out any cryptographic computation. When the protocol message is valid for the tag, its output is 1; otherwise, its output is 0. An adversary cannot corrupt $\mathcal{S}$.

Communication channel. The tag and the reader communicate over a wireless channel. Thus, an adversary can eavesdrop, modify, intercept, and insert any data in this channel. On the other hand, the reader and the authentication server communicate over a wired channel. We can easily establish a virtual private network between them. Thus, we assume that this channel is a secure channel, that is, both entities are authenticated and nobody can obtain plaintexts. For simplicity, we will think the reader and the server as one entity for the rest of this paper.

Now we consider leakage-resilient security and forward privacy of RFID tags. We follow the leakage-resilient model in [10,21] where the leakage-resilient property is captured by allowing the adversary to access to an oracle $\mathsf{Leakage}(\cdot)$, by which the adversary can gain information on the tag internal state. Formally, as in [10,21], the adversary can submit a function f of its choice, and receives $f(\mathtt{TState}^+)$ where $\mathtt{TState}^+$ is the active part of the tag state. The adversary can repeat the submission many times, with different f's. One restriction is that for each f the length $|f(\mathtt{TState}^+)|$ must be bounded away from $|\mathtt{TState}^+|$ or otherwise no security is guaranteed. Our RFID proposal will tolerate the same type and amount of leakage as Pietrzak [21] stream cipher, which is briefly recalled Sect. 3.1. We will also describe the concrete type of leakage information in our RFID protocol later in the proofs.

We now adapt the (standard, no-leakage-resilient) security and forward privacy definitions (see, e.g. [3]) to leakage-resilient world. Below, we denote by

$A \leftrightarrow B$ interactions between the parties A and B; and by $\mathcal{A}^{\mathsf{Leakage}(\cdot)}$ we mean that $\mathcal{A}$ has access to the oracle $\mathsf{Leakage}(\cdot)$.

First, security of an RFID tag essentially means that no-one, except legitimate tags, can make the reader outputs OK.

Definition 1 (Leakage-resilient tag unforgeability). *The adversary $\mathcal{A}$ runs in two phases. In phase 1 (learning phase), it interacts with the tag and the reader in a man-in-the-middle way, and furthermore has access to a leakage oracle: Tag $\leftrightarrow \mathcal{A}^{\mathsf{Leakage}(\cdot)} \leftrightarrow$ Reader. In phase 2 (impersonation), $\mathcal{A}$ interacts only with the reader only once, and it wins if the reader outputs OK. An RFID protocol has leakage-resilient security iff the probability* $\Pr[A \text{ wins}]$ *is negligible.*

Above, we assume that in phase 2, the adversary interacts with the reader only once. One may also let the adversary play polynomial times with the readers in the phase, but this case can be reduced to the above definition [3]. We will stick to the above for simplicity.

Second, forward privacy essentially means that no-one, even having the current state of a tag, can trace its past interactions, and is formalized in the definition below, which at the same time captures the intuition of unlinkability.

Definition 2 (Leakage resilient forward privacy). *The adversary $\mathcal{A}$ runs in two phases. In phase 1 (learning phase), it interacts with two tags: $Tag_0 \leftrightarrow \mathcal{A}^{\mathsf{Leakage}(\cdot)} \leftrightarrow$ Reader, and $Tag_1 \leftrightarrow \mathcal{A}^{\mathsf{Leakage}(\cdot)} \leftrightarrow$ Reader. (Recall that the $\mathsf{Leakage}(\cdot)$ oracle models the partial leakage gained by the adversary by side channel attacks.) In phase 2 (guessing phase), a bit d is chosen randomly, and now $\mathcal{A}$ interacts with tag d: $Tag_d \leftrightarrow \mathcal{A}^{\mathsf{Leakage}(\cdot)} \leftrightarrow$ Reader. At the end of phase 2, $\mathcal{A}$ is given the full internal state of tag d, and outputs a bit d' as a guess for d. The RFID protocol has leakage-resilient forward privacy iff the probability* $\Pr[d' = d]$ *is negligibly close to* $1/2$.

3 Our Proposal

3.1 Building Block

Recall the min-entropy of a random variable X is defined as

$$H_\infty(X) = -\log(\max_x \Pr[X = x]).$$

Below, $F : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^{k+n}$ is a weak PRF, which is intuitively a function returning a random output when the input is random. The difference between weak PRFs and normal PRFs is that, normal PRFs will output a random value on any (not just random) input. An adversary $\mathcal{A}$ against F will try to distinguish its outputs from random numbers. In particular, F is called (ϵ, q)-secure if the value

$$\left| \Pr[A(X_1 \dots X_q, Y_1 \dots Y_q) \to 1] - \Pr[A(X_1 \dots X_q, R_1 \dots R_q) \to 1] \right| \leq \epsilon$$

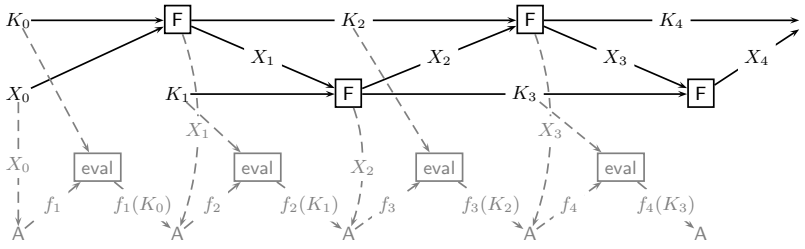

Fig. 1. Leakage resilient stream cipher in [21]. The gray, dashed lines show the leakage information the adversary gets in each round.

is negligible in the following experiment: $K \xleftarrow{\$} \{0,1\}^k$, $X_1, \ldots X_q \xleftarrow{\$} \{0,1\}^n$, $Y_i \leftarrow F(K, X_i)$, $R_i \xleftarrow{\$} \{0,1\}^{k+n}$ $(1 \le i \le q)$. When the value q can be set large and is not important in the context, we will omit to write it. The following theorem is an interesting fact on weak PRFs with *non-uniform* keys.

Theorem 1 (wPRF with non-uniform keys [21]). *A weak PRF, on random inputs, still returns random outputs if the key has high entropy (yet is non-uniform). More precisely, an ϵ-secure wPRF on random keys K, will become $(2^\lambda \cdot \epsilon)$-secure for keys K' with $H_\infty(K') \ge H_\infty(K) - \lambda$.*

We now consider the stream cipher of Pietrzak [21] based on any weak PRF $F$and is denoted as SC^F. In Fig. 1, the stream cipher is in black, while the related attack is in gray with dashed lines. The precise description is as follows.

Initialization: The initial state is $S_0 = [K_0, K_1, X_0]$, where $K_0, K_1 \xleftarrow{\$} \{0,1\}^k$ and $X_0 \xleftarrow{\$} \{0,1\}^n$. Only K_0, K_1 must be kept secret; X_0 can be public.
State: The state before the i-th round begins is $S_{i-1} = [K_{i-1}, K_i, X_{i-1}]$.
Computation: In the i-th round, the stream cipher SC^F on input of state S_{i-1}, computes

$$(K_{i+1}, X_i) := F(K_{i-1}, X_{i-1})$$

and outputs X_i. Then, the state $S_{i-1} = [K_{i-1}, K_i, X_{i-1}]$ is replaced with $S_i = [K_i, K_{i+1}, X_i]$.

Consider a side-channel adversary against the stream cipher; namely an adversary $\mathcal{A}$ who attacks SC^F by choosing an arbitrary function $f_i : \{0,1\}^k \to \{0,1\}^\lambda$ for fixed $\lambda < n$ before round i begins, and receives the output X_i of S^F and also leakage $\Lambda_i \stackrel{def}{=} f_i(K_{i-1})$ at the end of the round. Let $\mathsf{view}_l^{\mathsf{SC}}$ denote the view of the adversary after X_l has been computed, i.e.,

$$\mathsf{view}_l^{\mathsf{SC}} = [X_0, \ldots, X_l, \Lambda_1, \ldots, \Lambda_l].$$

The following theorem, which summarizes the results of [21], is the starting point of our work.

Theorem 2 ([21]). *Assume that* $F : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^{k+n}$ *is a secure weak PRF, the values* X_l, K_{l+1} *are indistinguishable from random, even when* $\mathsf{view}^{\mathsf{SC}}_{l-1}$ *is given to the adversary. Moreover, the value* X_l *still holds random even if the future states* $S_j = [K_j, K_{j+1}, X_j]$ *for* $j \geq l+1$ *are additionally given to the adversary.*

In addition to $\mathsf{view}^{\mathsf{SC}}_{l-1}$ *and* $S_j = [K_j, K_{j+1}, X_j]$ *for* $j \geq l+1$*, when the leakage* $\Lambda_l = f_l(K_{l-1})$ *(of* λ *bits) is given to the adversary, the value* X_l*, while not random anymore, still has high entropy (of about* $n - \lambda - 80$ *bits with probability* $1 - 2^{-80}$*).*

As estimated in [21], the leakage amount λ can reach $\Omega(|k|)$ if F is exponentially hard (like DES or AES).

3.2 Our Leakage-Resilient RFID Protocol

We provide in this section our RFID protocol secure against side chanel attacks with security proofs in the standard model. In essence, we build the scheme in a challenge-response manner, while utilizing Pietrzak mode of operation (Eurocrypt '09) as the main building block. The proposal is depicted in Fig. 2, and an imaginative illustration is in Fig. 3.

Let us mention some intuitions why the scheme is secure. The challenge-response construction helps the scheme resist against replay attack in which the adversary re-uses past transcripts. Security and forward privacy are ensured by the usage of Pietrzak mode in the tag, as well as an additional psuedo-random function F_2, which makes the responses look random.

$\underline{\text{Tag}(ID, K_0, K_1, X_0)}$		$\underline{\text{Reader}(\{\ldots, (ID, K_0, K_1, X_0), \ldots\})}$
Let $\texttt{TState} = (K_{i-1}, K_i, X_{i-1})$		$a \xleftarrow{\$} \{0,1\}^n$
	$\xleftarrow{a}$	
$(K_{i+1}, X_i) \leftarrow F_1(K_{i-1}, X_{i-1})$ $b \leftarrow F_2(X_i, a)$. Erase (K_{i-1}, X_{i-1})		
	$\xrightarrow{b}$	
$\texttt{TState} = (K_i, K_{i+1}, X_i)$		For each (ID, K_0, K_1, X_0), compute $X_1, \ldots, X_L$ and check $b \stackrel{?}{=} F_2(X_i, a)$. Return OK soon after the first match.

Fig. 2. Our proposal in standard model. $F_1, F_2 : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^{k+n}$ (for $k = n$) are weak PRF, and PRF respectively. L is a big and fixed threshold. The notation $a \xleftarrow{\$} \{0,1\}^n$ stands for picking a randomly from the set. The tag runs Pietrzak mode of operation. For an imaginative illustration, see Fig. 3. The reader may be speeded-up as in Fig. 4, but for simplicity, we will stick to the above when proving securities.

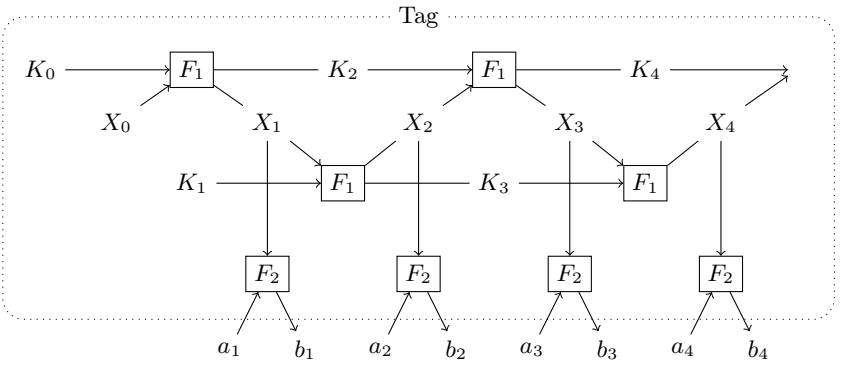

Fig. 3. An imaginative illustration of our leakage-resilient RFID protocol

Before stating security theorems, let us define the leakage to which our proposal is resilient. Mimicking the notation in Sect.3.1, denote

$$\mathsf{view}_l^{\mathsf{Tag}}(\mathcal{A}) = [g_0(X_0), \ldots, g_l(X_l), f_1(K_0), \ldots, f_l(K_{l-1})]$$

as the view in round l of an adversary $\mathcal{A}$ attacking our RFID scheme. The functions $g_0, \ldots g_l, f_1, \ldots, f_l : \{0,1\}^n \to \{0,1\}^\lambda$ are chosen by $\mathcal{A}$, representing the information $\mathcal{A}$ obtains from the tag's states up to the current round. Again, the value λ represents the leakage information on each value in the internal state of the tag, and is estimated as $\Omega(|k|)$ if F_1 is exponentially hard.

Theorem 3 (Tag unforgeability). *The RFID scheme has security even if for any round l in the learning phase (see Def.1), the adversary $\mathcal{A}$ is given* $\mathsf{view}_{l-1}^{\mathsf{Tag}}(\mathcal{A})$.

Proof. Recall that the adversary $\mathcal{A}$, playing actively between the tag and the reader for a while (learning phase), finally wants to impersonate the tag (impersonation phase). To impersonate the tag, the adversary has to create a value b satisfying $b = F_2(X_i, a)$ for random a (from the reader) and some X_i (unknown to the adversary). Note that the information on the value X_i may be partially leaked to $\mathcal{A}$, and since it is used just once, it may be leaked by at most λ bits, so its min-entropy is at least $n - \lambda$. Theorem 1 allows us to say that $F_2(X_i, a)$ is random-like for random a, so that the probability $\Pr[b = F_2(X_i, a)]$ is negligible since F_2 is a PRF.

Of course, before the above, we need to simulate the interaction (Tag $\leftrightarrow$ $\mathcal{A}^{\mathsf{Leakage}(\cdot)} \leftrightarrow$ Reader) in the learning phase of $\mathcal{A}$, where the adversary plays between the tag and the reader. However, the simulation is easy, since what sends from the reader is random, and what sends from the tag is random-like (F_2 is a PRF with random key), even if giving $\mathcal{A}$ the same leakage information as in Pietrzak [21]. More formal arguments are as follows. First, the simulator chooses X_0, K_0, K_1 randomly. In any subsequent round $l(\geq 1)$, the leakage

on internal states given to $\mathcal{A}$ is $\mathsf{view}_{l-1}^{\mathsf{Tag}}(\mathcal{A})$=$[g_0(X_0), \ldots, g_{l-1}(X_{l-1}), f_1(K_0), \ldots, f_{l-1}(K_{l-2})]$ for adversarily-chosen $g_0, \ldots, g_{l-1}, f_1, \ldots, f_{l-1}$ (which is adaptively submitted to the oracle $\mathsf{Leakage}(\cdot)$). It is ensured by Theorem 2 that given $\mathsf{view}_{l-1}^{\mathsf{Tag}}(\mathcal{A}) \subseteq \mathsf{view}_{l-1}^{\mathsf{SC}}$, the value X_l, K_{l+1} still looks random. Therefore, when receiving a challenge a'_l from $\mathcal{A}$ (who in turn has a_l from the reader), the simulator chooses $X_l = X \xleftarrow{\$} \{0,1\}^n$ and returns $b_l = F_2(X, a'_l)$ to $\mathcal{A}$. Since F_2 is a PRF and X is random, the value b_l looks random from the view of $\mathcal{A}$, and hence it gives the adversary essentially no information. The adversary then send b'_l to the reader, who returns OK iff $b'_l = F_2(X, a_l)$. (Certainly, if $\mathcal{A}$ did nothing, then $a'_l = a_l$ and $b'_l = b_l$, so OK will be returned.) The point here is in the random key X, which we can safely choose for simulation thanks to Theorem 2. □

Remark. Our RFID tag, with a slight modification, can tolerate more leakage, of the form $\mathsf{view}'_{l-1}(\mathcal{A}) = [g_0(X_0), \ldots, g_{l-1}(X_{l-1}), f_1(K_0), \ldots, f_{l-1}(K_{l-2})] \cup [f_l(K_{l-1})]$. In this case, Theorem 2 ensures that X_l still has high entropy of $n - \lambda - 80$ bits with overwhelming probability. Now, in order to gain a random key for F_2, we can use a strong extractor [24] applied to X_l. The remark applies as well to the proof of forward privacy below. The trade-off for this bigger leakage amount is in the efficiency of the tag, since we need an extractor and additional randomness.

Theorem 4 (Forward privacy). *The RFID scheme has leakage-resilient forward privacy even if for any round l (before the exposure of the internal state), the adversary $\mathcal{A}$ is given* $\mathsf{view}_{l-1}^{\mathsf{Tag}}(\mathcal{A})$.

Proof. We first recall the definition of forward privacy for RFID tag. In the learning phase, the adversary $\mathcal{A}$ interacts with two tags and with the reader: $\mathrm{Tag}_0 \leftrightarrow \mathcal{A}^{\mathsf{Leakage}(\cdot)} \leftrightarrow$ Reader, $\mathrm{Tag}_1 \leftrightarrow \mathcal{A}^{\mathsf{Leakage}(\cdot)} \leftrightarrow$ Reader. And then a bit $d \xleftarrow{\$} \{0,1\}$, and $\mathcal{A}$ continues: $\mathrm{Tag}_d \leftrightarrow \mathcal{A}^{\mathsf{Leakage}(\cdot)} \leftrightarrow$ Reader. Finally in the guessing phase, the internal state of the tag d is given to $\mathcal{A}$, whose goal is to guess the bit d.

We now proceed to the proof. Note that, in the learning phase, the adversary obtain almost no information from our RFID system. Again, the reason is in the fact that the communication between the reader and the tag consists of random-like values. Formally, for each tag i $(= 0, 1)$, the simulation goes as follows: at the beginning, random values X_0, K_0, K_1 are chosen randomly. For any subsequent round $l(\geq 1)$, the values X_l, K_{l+1} are also randomly picked to answer the query from $\mathcal{A}$ (with $\mathsf{view}_{l-1}^{\mathsf{Tag}_i}(\mathcal{A})$) in the round in the following manner: the adversary $\mathcal{A}$ (receiving a random a from the reader) sends its decided a', for which the adversary gets $b' = F_2(X_l, a')$ from the simulator. The adversary now decide the value b sent to the tag, and if $b = F_2(X_l, a)$, then OK will be returned to $\mathcal{A}$.

Furthermore, in some adversarily-chosen round l^* of the guessing phase, the simulator gives $\mathcal{A}$ randomly chosen values $[K_{l^*}, K_{l^*+1}, X_{l^*}]$ as the current internal state of Tag_d. The reason behind this simulation is that the current internal state of the tag is always random-like, even conditioned on the view so far of $\mathcal{A}$, which includes $\mathsf{view}_{l^*-1}^{\mathsf{Tag}_d}(\mathcal{A})$ (and the leakage $\mathcal{A}$ obtains from Tag_{1-d}, which is

independent of $\mathsf{view}_{l^*-1}^{\mathsf{Tag_d}}(\mathcal{A})$). Also here, we make use of the fact that Pietrzak mode, as used in our proposal, is one-way: from the i-th state (K_i, K_{i+1}, X_i), no-one cannot compute the $(i-1)$-th state (K_{i-1}, K_i, X_{i-1}), because the key K_{i-1} have been deleted, and F_1 (a weak PRF) is one-way without the key K_{i-1}. (To see why one-wayness is needed, imagine the case $\mathcal{A}$ obtain the initial state (K_0, K_1, X_0) of one tag. It is then clear that $\mathcal{A}$ can easily trace back past action of that tag.)

Based on the above arguments, we conclude that the guess bit d' output by $\mathcal{A}$ is computationally independent of d and hence $\Pr[d' = d] \overset{c}{\approx} 1/2$, ending the proof. □

Fig. 4 shows a speed-up version of our proposal, where the server begins its computation from the most recent tags' state (instead starting from the initial state). Both security and forward privacy are proven similarly as the original version.

Tag(ID, `TState`)		Reader(L, $\{\ldots, (ID, l(=1), K_0, K_1, X_0), \ldots\}$)
Retrieve `TState` $= (K_{i-1}, K_i, X_{i-1})$	$\xleftarrow{a}$	$a \xleftarrow{\$} \{0,1\}^n$
$(K_{i+1}, X_i) \leftarrow F_1(K_{i-1}, X_{i-1})$ $b \leftarrow F_2(X_i, a)$. Erase (K_{i-1}, X_{i-1})	$\xrightarrow{b}$	

`TState` $\leftarrow (K_i, K_{i+1}, X_i)$		For each ID retrieve $(l, K_{l-1}, K_l, X_{l-1})$.
		Set $j \leftarrow l$.
		`while` $(j < L)$
		  $(K_{j+1}, X_j) \leftarrow F_1(K_{j-1}, X_{j-1})$
		  `if` $b = F_2(X_j, a)$ `then`
		  `update` ID's state to
		  $(ID, l \leftarrow j+1, K_j, K_{j+1}, X_j)$
		  `return` OK and `halt`
		  `else` $j \leftarrow j+1$
		`endwhile`
		`return` NOK and `halt`

Fig. 4. Speed-up version for the reader, who keeps track of the most recent tag state (indexed by l), instead of starting from the initial (K_0, K_1, X_0) as in Fig. 2

3.3 Relation with Existing Schemes

Here, we show advantages of our proposal against existing schemes. Compared to Berbain et al. [3] scheme, our scheme has leakage-resiliency for security. Pietrzak's leakage-resilient stream cipher helps our scheme to realize this characteristic. Moreover, leakage model of our proposal for forward privacy is extended from existing schemes (OSK and Berbain et al.) In existing scheme, the adversary obtains full internal state at the end of the attack. However, he cannot

obtain leakages on internal states of previous moments. On the other hand, our proposal is secure even if the adversary obtains partial internal states of previous protocol executions.

Security of our protocol is rigorously proven thanks to Pietrzak's work. Leakage-resiliency is involved in the protocol and we need no refresh operations, as compared to [7].

3.4 Using Other Leakage Resilient Stream Cipher

Recently, Yu et al. proposed leakage resilient stream cipher with less secret information [27]. We can also replace Pietrzak's leakage resilient stream cipher with this new stream cipher in the same manner as our proposed RFID authentication protocol.

4 Conclusion

In this paper, we propose a concept of leakage-resiliency suitable for RFID-authentication protocol. Then, we propose the first RFID authentication scheme with leakage resilience for security and forward privacy. Our protocol has an additional functionality, i.e., leakage resilience for both security and privacy in contrast to existing protocols [20,3,7] with forward privacy. The security of our protocol is proven based on Pietrzak's pseudo-random generator.

References

1. Alwen, J., Dodis, Y., Wichs, D.: Leakage-Resilient Public-Key Cryptography in the Bounded-Retrieval Model. In: Halevi, S. (ed.) Advances in Cryptology - CRYPTO 2009. LNCS, vol. 5677, pp. 36–54. Springer, Heidelberg (2009)
2. Avoine, G., Oechslin, P.: A Scalable and Provably Secure Hash Based RFID Protocol. In: Proc. of IEEE Int. Workshop on Pervasive Computing & Communication Security (PerSec 2005). IEEE Computer Society Press, Los Alamitos (2005)
3. Berbain, C., Billet, O., Etrog, J., Gilbert, H.: An efficient forward private RFID protocol. In: ACM Conference on Computer and Communications Security 2009 (ACM CCS 2009), pp. 43–53 (2009)
4. Burmester, M., van Le, T., De Medeiros, B.: Provably Secure Ubiquitous Systems: Universally Composable RFID Authentication Protocols. In: Proc. of 2nd IEEE Create Net Int. Conf. on Security and Privacy in Networks (SECURECOMM 2006). IEEE Press, Los Alamitos (2006)
5. Bogdanov, A., Leander, G., Paar, C., Posehmann, A., Robshaw, M.J.B., Seurin, Y.: Hash Functions and RFID Tags: Mind the Gap. In: Oswald, E., Rohatgi, P. (eds.) CHES 2008. LNCS, vol. 5154, pp. 283–299. Springer, Heidelberg (2008)
6. Burmester, M., De Medeiros, B.: The Security of EPC Gen2 Compliant RFID Protocols. In: Bellovin, S.M., Gennaro, R., Keromytis, A.D., Yung, M. (eds.) ACNS 2008. LNCS, vol. 5037, pp. 490–506. Springer, Heidelberg (2008)
7. Burmester, M., Munila, J.: A Flyweight RFID Authentication Protocol. In: Workshop on RFID Security, RFIDSec 2009, Leuven, Belgium (July 2009), `http://eprint.iacr.org/2009/212.pdf`

8. Cash, D., Ding, Y.Z., Dodis, Y., Lee, W., Lipton, R., Walfish, S.: Intrusion-Resilient Key Exchange in the Bounded Retrieval Model. In: Vadhan, S.P. (ed.) TCC 2007. LNCS, vol. 4392, pp. 479–498. Springer, Heidelberg (2007)
9. Di Crescenzo, G., Lipton, R., Walfish, S.: Perfectly Secure Password Protocols in the Bounded Retrieval Model. In: Halevi, S., Rabin, T. (eds.) TCC 2006. LNCS, vol. 3876, pp. 225–244. Springer, Heidelberg (2006)
10. Dziembowski, S., Pietrzak, K.: Leakage-resilient cryptography. In: Proc. In FOCS (2008), October 25-28, pp. 293–302 (2008)
11. European Network of Excellence (ECRYPT). The side channel cryptanalysis lounge, `http://www.crypto.ruhr-uni-bochum.de/en_sclounge.html`
12. Goldreich, O., Goldwasser, S., Micali, S.: How to construct pseudo-random functions. Journal of ACM 33(4) (1986)
13. Henrici, D., Muller, P.M.: Hash-based enhancement of location privacy for radio-frequency identification devices using varying identifiers. In: Proc. of IEEE Int. Conf. on Pervasive Computing and Communications, pp. 149–153 (2004)
14. Juels, A., Pappu, R.: Squealing Euros:Privacy-Protection in RFID-Enabled Banknotes. In: Wright, R.N. (ed.) FC 2003. LNCS, vol. 2742, pp. 103–121. Springer, Heidelberg (2003)
15. Juels, A., Weis, S.A.: Defining Strong Privacy for RFID, `http://eprint.iacr.org/2006/137`
16. Le, T.V., Burmester, M., de Medeiros, B.: Universally Composable and Forward-secure RFID Authentication and Authenticated Key Exchange. In: Proc. of ASIACCS 2007, pp. 242–252 (2007)
17. Naor, M., Segev, G.: Public-Key Cryptosystem Resilient to Key leakage. In: Halevi, S. (ed.) CRYPTO 2009. LNCS, vol. 5677, pp. 18–35. Springer, Heidelberg (2009)
18. Ng, C.Y., Susilo, W., Mu, Y., Safavi-Naini, R.: RFID Privacy Models Revisited. In: Jajodia, S., Lopez, J. (eds.) ESORICS 2008. LNCS, vol. 5283, pp. 251–266. Springer, Heidelberg (2008)
19. Ohkubo, M., Suzuki, K.: Forward Security RFID Privacy Protection Scheme with Restricted Traceability. In: Proc. of ACNS 2006 in Industrial Track, pp. 1–16 (2006)
20. Ohkubo, M., Suzuki, K., Kinoshita, S.: Cryptographic Approach to a Privacy Friendly Tags. Presented at the RFID Privacy Workshop, MIT, USA (2003)
21. Pietrzak, K.: A Leakage-Resilient Mode of Operation. In: Joux, A. (ed.) EUROCRYPT 2009. LNCS, vol. 5479, pp. 462–482. Springer, Heidelberg (2010)
22. Pietrzak, K., Sjodin, J.: Range Extension for Weak PRFs; The Good, the Bad, and the Ugly. In: Naor, M. (ed.) EUROCRYPT 2007. LNCS, vol. 4515, pp. 517–533. Springer, Heidelberg (2007)
23. Petit, C., Standaert, F.-X., Pereira, O., Malkin, T., Yung, M.: A Block Cipher based Pseudo Random Number Generator Secure against Side-channel Key Recovery. In: Proc. of ASIACCS 2008, pp. 56–65 (2008)
24. Shaltiel, R.: Recent developments in explicit constructions of extractors. Bulletin of the EATCS 77, 67–95 (2002)
25. Sharma, S.E., Weiss, S.A., Engels, D.W.: RFID systems and security and privacy implications. In: Kaliski Jr., B.S., Koç, Ç.K., Paar, C. (eds.) CHES 2002. LNCS, vol. 2523, pp. 454–469. Springer, Heidelberg (2003)
26. Vaudenay, S.: On Privacy Models for RFID. In: Kurosawa, K. (ed.) ASIACRYPT 2007. LNCS, vol. 4833, pp. 68–87. Springer, Heidelberg (2007)
27. Yu, Y., Standaert, F.-X., Pereira, O., Yung, M.: Practical Leakage-Resilient Pseudorandom Generators. In: Proc. of ACM CCS 2010 (to appear, 2010)