Strong Privacy for RFID Systems from Plaintext-Aware Encryption

Khaled Ouafi[1] and Serge Vaudenay[2]

[1] IP Video SA, CH-1205 Geneva, Switzerland
[2] EPFL, CH-1015 Lausanne, Switzerland
http://lasec.epfl.ch

Abstract. The Vaudenay model for RFID privacy from Asiacrypt 2007 suffers from the impossibility to address strong privacy. It has however been shown by Ng et al. at ESORICS 2008 that the impossibility result leads to no practical threat, so that the definition from 2007 may be unnecessarily strong. This paper proposes a slight change in the definition of privacy from the Vaudenay model (Asiacrypt 2007). Then, we show that by adding a plaintext-aware assumption on the public-key cryptosystem, the proposed protocol always achieves strong privacy with our new definitions.

1 Introduction

An RFID system consists of 3 components: a *back-end database*, a number of *readers* and *tags*. Tags communicate with readers through a *wireless link* to authenticate themselves. While tags can only maintain one session, readers can communicate with several tags in parallel. Without loss of generality, we assume that the RFID system has only one reader. So, during the authentication process, the reader queries the back-end database through a secure link. Clearly, RFID tags face two contradictory requirements: on the one hand, they must securely identify to a reader; on the other hand, they must hide any traceable information to observers or adversaries.

RFID tags are identified by a unique ID. Cheap tags may be corruptible: it may be possible to open them and read the content of their non-volatile memory. Additionally, they have no internal clock. Such a tag has limited memory and computational power. It can perform symmetric-key based operations: pseudo-random generation, hash computations [10], symmetric-key encryption [9,19] and message authentication codes [33]. However, these limitations vary depending on the application and the allocated budget. Best scenarios allow the tag to use elliptic-curve cryptography [21].

Privacy Models. Several efforts were put in transposing the notion of privacy for RFID, which resulted in several models [3,28,2,25,30,34,15,16,17,11,27,13,31].

Arguably, the definition given by Vaudenay [34] is the most general one. It considers concurrence, tampering (i.e., getting the internal state of an anonymous tag), and the return channel from the reader (i.e., whether a protocol session on the reader side is accepting or not). Contrarily to several other models, it allows adversaries to interact with many concurrent anonymous tags sampled with arbitrary distributions. One difficulty

J. Pieprzyk, A.-R. Sadeghi, and M. Manulis (Eds.): CANS 2012, LNCS 7712, pp. 247–262, 2012.

is to identify non-trivial leakage. For instance, let us assume an adversary interacting concurrently with four different anonymous tags. The first three are known to be thrown in a set of three tags numbered 1, 2, and 3, and the last anonymous tag is known to be thrown among a pair of tags numbered 1 and 4. In this case, the adversary trivially infers that this last anonymous tag must be the tag numbered 4. Following simulation-based notions, an information is trivial if the same one could be obtained when the protocol messages are simulated by an additional process called a blinder. The blinder is separate from the adversary and the system. It has therefore neither secrets. However, it knows all interactions.

In [34], Vaudenay shows that Strong privacy (i.e., privacy when adversaries can corrupt any anonymous tags and read the return channel) cannot be achieved. Intuitively, if an adversary creates a legitimate tag then corrupts it, he can then simulate either this tag or an illegitimate one to a reader and the return channel will tell them apart. However, no blinder should be able to do it (otherwise, we would define another adversary from it). So, it may be considered as some non-trivial information. This was quite puzzling since this adversary would by no mean be any threat in practice. Indeed, this example heavily relies on the adversary knowing the expected behavior of the environment and the impossibility to simulate it without guessing what the adversary expects. For this reason, several papers were dedicated to fixing the shortcomings of this model.

At first, Ng et al. [29] proposed at ESORICS 2008 the notion of a "wise adversary", modeling adversaries who cannot guess the behavior of the environment. This fix consists of not allowing adversaries to ask questions for which they know the answer. Canard et al. [12] imposed a different restriction on the adversary called "future-untraceability". This requires, for every adversary, the existence of a simulator for which the output of the adversary is unaffected.

At ESORICS 2011, Hermans et al. [22] proposed a simpler reformulation of Vaudenay's privacy definition that would allow Strong privacy from being achievable by getting rid of the simulation-based approach.

Vaudenay's privacy model was also extended to the case of mutual authentication in a work by Paise and Vaudenay [32]. However, some results were flawed, as discussed by Armknecht et al. [1]. Actually, they show that no RFID protocol with mutual authentication can achieve strong privacy and security at the same time for reasons which are essentially similar to the ones in [34].

In this paper, we use knowledge extractors from plaintext-aware encryption schemes [4,5,6]. Loosely speaking, plaintext-aware encryption schemes are public-key cryptosystems in which the only way for an adversary to produce a valid ciphertext is to choose a plaintext and to encrypt it. So, by reading the adversary's mind, one could extract the corresponding plaintext.

Our contributions. We propose to update the Vaudenay model by changing the definition of the blinder. In short, we allow the blinder to access the random coins used by the adversary so that he could "read his mind" and predict the behavior of the environment as well. This could fix the impossibility result from [34] and [1]. Then, we show by using plaintext aware encryption techniques that Strong privacy can be achieved in our model with the simple protocol (called PKC protocol herein) of [34]. Our result provides strong confidence in the privacy protection deployed by the PKC protocol.

In Appendix, we further show that IND-CCA security is not enough for the PKC protocol to reach strong privacy in the sense that the system may leak some non-simulatable information. To show this, we construct a cryptosystem which is IND-CCA secure but not plaintext-aware.

2 Preliminaries

A function $f(k)$ is said to be polynomial if there exists a constant $n \in \mathbb{N}$ such that $f(k)$ is $O(k^n)$. Similarly, $f(k) = \mathsf{negl}(k)$ if, for every $n \in \mathbb{N}$, $f(k)$ is $O(k^{-n})$.

For an algorithm $\mathcal{A}$, $\mathcal{A}(y;r) \to x$ represents the output after running $\mathcal{A}$ on input y with coins r. The view of $\mathcal{A}$, denoted $\mathsf{view}_{\mathcal{A}}$, is defined to include all the inputs and random coins of $\mathcal{A}$ along with the list of the messages $\mathcal{A}$ received. The ability of an algorithm to query an oracle O is denoted $\mathcal{A}^O$.

Given two algorithms $\mathcal{A}_0$ and $\mathcal{A}_1$ of same input/output domains, we define a probabilistic polynomial-time algorithm $\mathcal{D}$ and its advantage

$$\mathsf{Adv}_{\mathcal{D}}^{\mathcal{A}_0,\mathcal{A}_1}(k) = \left|\Pr[\mathcal{D}^{\mathcal{A}_0}(1^k) \to 1] - \Pr[\mathcal{D}^{\mathcal{A}_1}(1^k) \to 1]\right|,$$

with the probability being taken over the random tape of all the algorithms. $\mathcal{A}_0$ and $\mathcal{A}_1$ are said to be computationally indistinguishable, if for every distinguisher $\mathcal{D}$, we have $\mathsf{Adv}_{\mathcal{D}}^{\mathcal{A}_0,\mathcal{A}_1}(k) = \mathsf{negl}(k)$.

Sampling Algorithms. An efficient sampling algorithm for a probability distribution p is a polynomial-time probabilistic algorithm, in k, denoted Samp, that, on input random coins $\rho \in \{0,1\}^{\ell(k)}$, with $\ell(\cdot)$ being a polynomial function, outputs vector elements from X such that $|\Pr_{\rho}[\mathsf{Samp}(\rho) = x] - p(x)| = \mathsf{negl}(k)$. We say that a sampling algorithm Samp is inverse-samplable if it is invertible and some conditions on the distributions are fulfilled.

Definition 1 (Inverse-Sampling Algorithm [23]). *We say that an efficient sampling algorithm* Samp *is inverse-samplable if there exists a polynomial-time inverter algorithm* Samp^{-1} *such that* $(\rho, \mathsf{Samp}(\rho))$ *and* $(\mathsf{Samp}^{-1}(x), x) | x = \mathsf{Samp}(\rho)$ *are indistinguishable*

Public-Key Encryption Schemes. A public-key encryption scheme consists of three polynomial-time probabilistic algorithms denoted KeyGen, Enc, and Dec such that for all $k \in \mathbb{N}$, $\Pr[\mathsf{Dec}_{sk}(\mathsf{Enc}_{pk}(m)) = m | \mathsf{KeyGen}(1^k) \to (sk, pk)] = 1$. The decryption algorithm may output $\perp$ if it could not decrypt a ciphertext c. We use the standard notions of IND-CPA and IND-CCA security. In the security game, the advantage of the adversary is denoted $\mathsf{Adv}^{\mathsf{IND\text{-}CPA}}$ resp. $\mathsf{Adv}^{\mathsf{IND\text{-}CCA}}$.

3 A Model for RFID Security and Privacy

Throughout this section, we recall the definitions in the Vaudenay model and our proposed updates. Most of what follows is taken from [34].

An RFID system is defined by a pair of two probabilistic polynomial-time algorithms and one two-party protocol to be executed between the reader and a tag. A first algorithm SetupReader is used to initialize the reader. It creates a pair of secret/public key (sk, pk) (typically, no public-key cryptography is used and $pk = \perp$). The second algorithm is for the creation of the tags and is $\mathsf{SetupTag}_{pk}(\mathsf{ID}) \rightarrow (K_{\mathsf{ID}}, S_{\mathsf{ID}})$, where ID refers to the identifier of the new tag. When the tag is *legitimate*, the tag secret K_{ID} is stored along with ID in the database; while the tag's initial state S_{ID} is always put inside the tag. An *illegitimate* tag has no entry in the database. Finally, a polynomial-time interactive protocol between the reader and a tag ID in which the reader ends up with a tape Output and the tag ends up with a tape $\mathsf{Output}_{\mathsf{ID}}$ completes the definition of an RFID system. By convention, if the protocol fails from the reader's perspective, we set $\mathsf{Output} = \perp$. When the protocol does not feature reader authentication, $\mathsf{Output}_{\mathsf{ID}}$ is void.

Simple RFID Protocols. We focus on a relevant class of RFID schemes called *simple*. These are 2-path protocols in which the reader sends a challenge and receives an answer. Then, it looks for a $(\mathsf{ID}, K_{\mathsf{ID}})$ database entry satisfying a predicate Ψ. The found pair identifies the tag and may be updated.

Definition 2 (Simple RFID Scheme). *An RFID scheme is said to be simple if the following conditions are fulfilled:*

- *The reader sends a query to the database with its secret key sk and the (possibly partial) transcript τ_p obtained from a protocol session.*
- *There exists a predicate Ψ, i.e., a deterministic polynomial-time algorithm that outputs a single bit, that takes as input sk, τ_p, and a database entry $(\mathsf{ID}, K_{\mathsf{ID}})$ such that the response from the database is computed by returning a database entry, picked uniformly, that satisfies the predicate.*
- *Once a tag ID has been identified in the database, its corresponding secret in the database, K_{ID}, may be updated to a new value. When it takes place, this procedure is carried out by an algorithm* Update *taking as input sk,* ID*, K_{ID}, and the full transcript of the protocol instance τ. This algorithm outputs a new K_{ID} and the database entry $(\mathsf{ID}, K_{\mathsf{ID}})$ is updated.*

It is straightforward to check that simple RFID schemes following Def. 2 satisfy the more general definition from [34].

Fig. 1 represents a simple RFID scheme from [34] which is based on a public-key cryptosystem. In what follows we call it the PKC protocol. In this scheme, the state of the tags is composed of their ID and a uniformly distributed κ-bit string K_{ID}. Upon reception of an α-bit string challenge a, a tag sends the encryption of $\mathsf{ID} \| K_{\mathsf{ID}} \| a$ under the public key pk to the reader. The latter decrypts the received ciphertext using its secret key sk and checks that it is well formed, that a is correctly recovered and that (ID, K) exists in the database.

Adversaries. Adversaries can request the creation of legitimate and illegitimate RFID tags. Furthermore, adversaries have the ability to draw one or more anonymous RFID tags, according to a chosen probability distribution. All interactions with the reader

Tag		**System**
State: $pk, \mathsf{ID}, K_{\mathsf{ID}}$		**Secret key:** sk **DB:** $\{\ldots, (\mathsf{ID}, K_{\mathsf{ID}}), \ldots\}$
	$\xleftarrow{a}$	Choose $a \in_R \{0,1\}^{\alpha}$
$c = \mathsf{Enc}_{pk}(\mathsf{ID} \| K_{\mathsf{ID}} \| a)$	$\xrightarrow{c}$	Search $(\mathsf{ID}, K_{\mathsf{ID}}) \in \mathsf{DB} : \mathsf{Dec}_{sk}(c) = \mathsf{ID} \| K_{\mathsf{ID}} \| a$ **Output:** ID or $\perp$

Fig. 1. PKC Protocol: an RFID scheme based on a public-key cryptosystem

and the drawn tags is controlled by the adversary. Moreover, the adversary has also the ability to tamper with any drawn tag and to retrieve its internal state. After a while, the adversary has also the possibility to release the tag so that it can be drawn again.

Definition 3 (Adversary [34]). *An adversary is a probabilistic polynomial-time algorithm. It takes a public key pk as input and has access to the following interfaces:*

- CREATETAG$^b(\mathsf{ID})$*: create a tag with unique identifier* ID*. Depending on the bit b submitted by the adversary, the tag may be legitimate, when* $b = 1$*, or illegitimate, when* $b = 0$*. After calling upon* $\mathsf{SetupTag}_{pk}(\mathsf{ID}) \to (K_{\mathsf{ID}}, S_{\mathsf{ID}})$ *for both type of tags, the pair* $(\mathsf{ID}, K_{\mathsf{ID}})$ *is inserted into the database if the adversary queried for a legitimate tag.*
- DRAWTAG$(\mathsf{Samp}) \to ((\mathsf{vtag}_1, b_1), \ldots, (\mathsf{vtag}_n, b_n))$*: select a vector of tags following a polynomial-time sampling algorithm* Samp*. During the period in which a tag is drawn, the adversary has complete control over its interactions. Along* vtag*, a bit b, set to 1 whenever the drawn tag is legitimate and to 0 otherwise, is returned. When a tag is drawn, it is designated by a unique virtual fresh identifier* vtag*. Drawing a tag that was already drawn makes the oracle output* $\perp$*.*[1] *Additionally, this interface keeps a private table* $\mathcal{T}$ *that keeps track of the real identifier of each drawn tag, i.e., it is such that* $\mathcal{T}(\mathsf{vtag})$ *is the real identifier of the virtual tag* vtag*.*
- FREE(vtag)*: release the RFID tag* vtag*. Once* vtag *is released, the adversary can no longer communicate with it (except under another pseudonym if it may be drawn again). Furthermore, its temporary memory is cleared to prevent a protocol session to span under several* vtag *pseudonyms.*[2]
- LAUNCH $\to \pi$*: make the reader launch a new protocol instance* π*. The returned* π *is a session identifier which can be assumed to be based on a counter.*
- SENDREADER$(m, \pi) \to m'$*: send a message m to a protocol instance* π *for the reader.*
- SENDTAG$(m, \mathsf{vtag}) \to m'$*: send a message m for the drawn tag* vtag *and receive the answer* m'*.*
- RESULT$(\pi) \to x$*: return the result of the completed protocol instance* π*. Namely, it yields* 0 *when* $\mathsf{Output} = \perp$ *and* 1 *otherwise.*

[1] Definition 3 only differs from the original one in [34] in the introduction of the sampling algorithm Samp in DRAWTAG queries. Vaudenay [34] uses the term of "distribution" for the input of DRAWTAG although its representation may have exponential length.

[2] The clearance of the temporary memory upon a FREE call was introduced in Paise-Vaudenay [32]. It also meets the notion of "clean tag" by Deng et al. [17].

– CORRUPT(vtag) → *S: return the current state S of the tag* $\mathcal{T}(\mathsf{vtag})$. *It does not return the content of the temporary memory of the tag.*

We consider several classes of adversaries. Weak adversaries do not use the CORRUPT interface. Forward adversaries can only use the CORRUPT interface at the end. Namely, no other interface can be used after the CORRUPT one. All other adversaries are Strong adversaries. We clearly have Weak $\subseteq$ Forward $\subseteq$ Strong. Adversaries that do not have access to the side channel information on the output of the protocol, i.e. to the RESULT oracle, are called NARROW. For any class P of adversaries, we define its Narrow counterpart for which we clearly have Narrow-P $\subseteq$ P.

In the sequel, we restrict to adversaries who use distributions to the DRAWTAG such that, at any step, the table $\mathcal{T}$ can be successfully simulated by an algorithm that is only given the view of the adversary as input. That is, we require adversaries to only submit sampling algorithms that are inverse-samplable and allow them to compute a plausible guess for the identity of drawn tags in polynomial-time. For this we introduce a new notion: simulatable adversaries.

Definition 4 (Simulatable adversary). *Let $\mathcal{A}$ be an adversary interacting with an RFID system. Let $\mathsf{view}^t_{\mathcal{A}}$ be the view of $\mathcal{A}$ at its t-th step and let $\mathcal{T}^t$ denote the table $\mathcal{T}$ of the* DRAWTAG *oracle at step t of $\mathcal{A}$. We say that the adversary $\mathcal{A}$ is simulatable if all her sampling algorithms submitted to* DRAWTAG *are inverse-samplable and, for all t, there exists a polynomial-time algorithm $\mathcal{A}'$, such that $(view^t_{\mathcal{A}}, \mathcal{T}^t)$ and $(view^t_{\mathcal{A}}, \mathcal{A}'(view^t_{\mathcal{A}}))$ are indistinguishable.*

We note that when the adversary only draws one tag at the time (or in general, a vector of logarithmic length), then our restrictions do not affect the original definition as any sampling algorithm over such a set is inverse-samplable. So, we believe that our restriction to simulatable adversaries is harmless.

Correctness. Basically, correctness formalizes the fact that whenever the reader and a tag ID participate in an undisturbed protocol session, the reader authenticates the tag, that is, it ends up with Output = ID, except with a small negligible probability. We include all malicious behaviors as it was done in [17].

Definition 5 (Correctness). *A scheme is correct if for any Strong adversary $\mathcal{A}$, whenever there is a matching conversation between a tag of identity* ID *produced by* DRAWTAG → (vtag, b) *and a reader instance π, except with negligible probability, π ends up with output $\perp$ if $b = 0$ and* ID *if $b = 1$.*

Clearly, the PKC protocol is correct.

Security. Security formalizes the fact that no adversary should be able to make the reader accept on a protocol session in which the adversary has been actively involved. Roughly, an RFID scheme is said to be secure if no adversary is able to make a reader protocol instance recognize an uncorrupted tag ID even if she corrupts all the other tags, unless π and the tag have a matching conversation.

It has been shown in [34] that, for the case of a simple RFID scheme, the notion of security reduces to an adversary playing the following game: create (and draw) a single

tag ID; use LAUNCH, SENDREADER, SENDTAG; use an oracle who checks the predicate $\Psi(sk,\cdot,\cdot,\cdot)$ on inputs different from ID; end on a final SEND command to complete the instance for the reader and the tag. The adversary wins the simple security game if one protocol instance on the reader identified tag ID but had no matching conversation. If the success probability of any adversary in wining the security experiment is negligible, then the scheme is simply secure. For simple schemes, simple security implies security.

It was shown in [34] that the PKC protocol is secure when the cryptosystem is IND-CCA secure.

When the protocol includes reader authentication, a security notion for the reader, in which the adversary's goal is to make the tag accept the reader, also needs to be defined. This was done in [32].

Privacy. An RFID scheme is private if no adversary can learn any information about the identity of drawn tags which is non-trivial. The information is trivial if the protocol messages could be simulated without interacting with the tags or reader and without affecting the output of the adversary. The simulation is performed by a process called a blinder.

Definition 6 (Blinder). *A blinder B for an adversary $\mathcal{A}$ is a polynomial-time algorithm which sees the same view as $\mathcal{A}$ (i.e, all the incoming messages and used coins), records all the adversary's Oracle queries and simulates all the* LAUNCH, SENDREADER, SENDTAG, RESULT *oracles to $\mathcal{A}$. A blinded adversary $\mathcal{A}^B$ is an adversary who does not produce any* LAUNCH, SENDREADER, SENDTAG, RESULT *oracles query but have them simulated by B.*

This definition changes from [34] by letting the blinder see the random tape of the adversary. This is a crucial change as the impossibility result in the Vaudenay model came from that adversaries could ask questions to the system for which they knew the answer but such that it could not be simulated. Providing used coins to the blinder allows it to "read the adversary's mind" and simulate the answer from the system.

Definition 7 (Privacy). *We consider simulatable adversaries who start with an attack phase consisting of* only *oracle queries and some computations then pursuing an analysis phase with no oracle query. In between phases, the adversary receives the hidden table $\mathcal{T}$ of the* DRAWTAG *oracle then outputs* true *or* false*. The adversary wins if the output is* true*. We say that the RFID scheme is P-private if for such adversary $\mathcal{A}$ which belongs to class P there exists a blinder B for which we have* $|\Pr[\mathcal{A}\ \mathsf{wins}] - \Pr[\mathcal{A}^B\ \mathsf{wins}]|$ *is negligible.*

Again, we only introduced that adversaries must be simulatable in this definition. Clearly, all positive results from [34] hold with these new definitions since privacy is defined by some property of form $\forall\mathcal{A}\ \exists\mathcal{B}$, our new adversaries are compatible with the old definition, and old blinders are compatible with the new definition. Namely:

- Weak privacy can be achieved using pseudo-random functions;
- The PKC protocol with IND-CCA encryption is Forward private;
- The randomized OSK protocol is Narrow-Forward private in ROM.

However, the impossibility of strong privacy may not hold anymore since it is a property of form $\exists\mathcal{A}\ \forall\mathcal{B}$. Actually, we will show that it no longer holds. This is similarly the case for the impossibility result by Armknecht et al. [1] for Narrow-Strong privacy and reader authentication, when mutual authentication is considered.

4 Plaintext-Awareness

Plaintext-awareness states that if an adversary is able to produce a valid ciphertext, then she should know the corresponding plaintext. Formalizing this notion has proven to be a non-trivial task [4,5,6,8,18]. In the end, several notions of plaintext-awareness were defined, such as, PA1, PA2, PA1+, and PA2+.

The difference between PA1 and PA2 lies in the attacker's ability to get ciphertexts from external sources. In the settings of PA2, there is an oracle $\mathcal{P}(\mathsf{aux})$, called plaintext creator and such that, on each query, it picks a message at random (or possibly according to a distribution partially defined by its input aux). The adversary can query $\mathsf{Enc}_{pk}(\mathcal{P}(\mathsf{aux}))$. Any ciphertext obtained through this oracle is added to a list CList, the list of ciphertexts for which the adversary does not know the corresponding plaintexts. The essence of plaintext-awareness is the existence of a polynomial-time algorithm $\mathcal{A}^\star$ (which construction may depend on $\mathcal{A}$), called plaintext extractor that successfully decrypts any ciphertext given by the adversary that was not returned by $\mathsf{Enc}_{pk}(\mathcal{P}(\mathsf{aux}))$. To carry out the extraction, $\mathcal{A}^\star$ is given the view of $\mathcal{A}$ (which includes CList and the random coins of $\mathcal{A}$) and the target ciphertext c to be decrypted for $c \notin \mathsf{CList}$.

To formalize information coming from external sources, Dent [18] extended PA1 to PA1+ for adversaries who can get hold of uniformly distributed bits from an external source. Later, Birkett and Dent [8] introduced the analog notion of PA2+ for PA2 plaintext-awareness. These last two notions were proven to be equivalent under the condition that the encryption scheme is IND-CPA [8]. PA1+ was also shown to imply PA2+ for simulatable encryption schemes [7]. We extend them to PA1++ and PA2++ when using any inverse-simulatable source.

Definition 8 (PA encryption). *Let O_1 denote an oracle that returns a single uniformly distributed bit. Let O_S be an oracle who takes as input an inverse-sampling algorithm and executes it using his own random tape. Given $* \in \{\mathsf{PA1}, \mathsf{PA1+}, \mathsf{PA1++}, \mathsf{PA2}, \mathsf{PA2+}, \mathsf{PA2++}\}$, we say that a public key cryptosystem* (KeyGen, Enc, Dec) *is $*$-plaintext-aware if $\forall\mathcal{A}$, $\exists\mathcal{A}^\star$, $\forall\mathcal{P}$, all probabilistic polynomial-time, $(pk, \mathcal{A}^{O_*(\mathsf{Dec}_{sk}(\mathcal{P}(\cdot)))}(pk))$ and $(pk, \mathcal{A}^{O_*(\mathcal{A}^\star(pk,\cdot,\mathsf{view}_\mathcal{A}))}(pk))$ are indistinguishable, where*

$$\begin{aligned} O_{\mathsf{PA1}}(o) &= (o) & O_{\mathsf{PA2}}(o) &= (\mathsf{Enc}_{pk}(\mathcal{P}(\cdot)), o) \\ O_{\mathsf{PA1+}}(o) &= (O_1, o) & O_{\mathsf{PA2+}}(o) &= (\mathsf{Enc}_{pk}(\mathcal{P}(\cdot)), O_1, o) \\ O_{\mathsf{PA1++}}(o) &= (O_S, o) & O_{\mathsf{PA2++}}(o) &= (\mathsf{Enc}_{pk}(\mathcal{P}(\cdot)), O_S, o) \end{aligned}$$

Note that PA1++ (resp. PA2++) plaintext-awareness trivially implies PA1+ (resp. PA2+). Actually, we can even show equivalence.

Theorem 9. *PA1+ and PA1++ (resp. PA2+ and PA2++) are equivalent.*

Proof. We prove the theorem for the case of PA1++. It can be easily modified so that it applies to PA2++. We thus assume PA1+ plaintext awareness. Let $\mathcal{A}$ be a PA1++ ciphertext creator. We want to construct a plaintext extractor $\mathcal{A}^\star$.

We construct a PA1+ ciphertext creator $\mathcal{B}$ as follows: $\mathcal{B}$ takes input pk and simulates $\mathcal{A}$, forwarding all its decryption queries to the decryption oracle. In order to answer $\mathcal{A}$'s queries to the randomness oracle, $\mathcal{B}$ runs the provided sampling algorithm and query its randomness oracle, that we denote O_1, every time a new random bit is asked for. Clearly, $\mathcal{B}$ terminates in polynomial-time if all samplings can be performed in polynomial-time. Remark that $\mathcal{B}$ does not use any internal randomness besides the one used to initialize $\mathcal{A}$.

Since $\mathcal{B}$ is a valid PA1+ ciphertext creator, we can assert the existence of a plaintext extractor $\mathcal{B}^\star$ indistinguishable from a decryption oracle. We use $\mathcal{B}^\star$ to construct a plaintext extractor $\mathcal{A}^\star$ for $\mathcal{A}$. In the following, we assume that $\mathcal{A}^\star$ maintains a state view' initialized to $\mathsf{view}_{\mathcal{A}}$ that will be used to simulate $\mathcal{B}$'s view. To answer $\mathcal{A}$'s decryption queries, $\mathcal{A}^\star$ proceeds as follow:

1. If $\mathcal{A}$ queried the randomness oracle with an inverse-sampling algorithm Samp and received x since the last invocation of $\mathcal{A}^\star$, then $\mathcal{A}^\star$ computes $\rho_S \leftarrow \mathsf{Samp}^{-1}(x)$. After that, $\mathcal{A}^\star$ updates the simulated view of $\mathcal{B}$ to include the random bits ρ_S, i.e., it sets $\mathsf{view}' \leftarrow \mathsf{view}' \| \rho_S$. Due to the property of inverse-sampling algorithms, $(\rho_S, \mathsf{view}_{\mathcal{A}})$ is indistinguishable from $(\rho, \mathsf{view}_{\mathcal{A}})$, where ρ is the random string returned by O_S for the sampling request. Thus, by induction we show that $\mathsf{view}_{\mathcal{B}}$ and view' are indistinguishable. This procedure is repeated for every new sampling query.
2. $\mathcal{A}^\star$ then calls upon $\mathcal{B}^\star(pk, c, \mathsf{view}')$ and forwards its output to $\mathcal{A}$.

Since $\mathsf{view}_{\mathcal{A}}$ is included in view' which is indistinguishable from $\mathsf{view}_{\mathcal{B}}$,

$$\left|\Pr[\mathcal{D}^{\mathcal{A}^{\mathcal{B}^\star(pk,\cdot,\mathsf{view}'),O_S(pk)}}(1^k) \to 1] - \Pr[\mathcal{D}^{\mathcal{A}^{\mathcal{B}^\star(pk,\cdot,\mathsf{view}_{\mathcal{B}}),O_S(pk)}}(1^k) \to]\right| = \mathsf{negl}(k).$$

Recalling that $\mathcal{B}^\star(pk, \cdot, \mathsf{view}_{\mathcal{B}})$ is indistinguishable from a decryption oracle to $\mathcal{A}$, we deduce that $\mathcal{A}^\star$ is a valid plaintext extractor. In other words,

$$\left|\Pr[\mathcal{D}^{\mathcal{A}^{\mathcal{B}^\star(pk,\cdot,\mathsf{view}'),O_S(pk)}}(1^k) \to 1] - \Pr[\mathcal{D}^{\mathcal{A}^{\mathsf{Dec}_{sk}(\cdot),O_S(pk)}}(1^k) \to 1]\right| = \mathsf{negl}(k).$$

This concludes the proof. □

The following corollary results from the combination of Theorem 9 with the equivalence result between PA2+ and PA2 [7], under the assumption that the scheme is IND-CPA secure.

Corollary 10. *If an encryption scheme is IND-CPA and PA2 plaintext-aware, then it is PA1++ plaintext-aware.*

5 Strong Privacy Is Possible

In this section, we show that using the new definition of blinders, we can achieve Strong privacy using plaintext-aware encryption schemes.

We consider the PKC protocol in Fig. 1. It has already been used by Vaudenay [34] to achieve Narrow-Strong privacy under the assumption that the underlying encryption scheme is IND-CPA secure, our result requires PA1+ plaintext-awareness from the encryption scheme. Naturally, since our definition of security is unchanged from the original model, IND-CCA security for the encryption scheme is sufficient to prove security [34].

Theorem 11 (PKC protocol achieves strong Privacy). *Assume a cryptosystem which is correct, PA1+ plaintext-aware, and IND-CCA secure. If* $2^{-\kappa}$ *and* $2^{-\alpha}$ *are negligible, then the PKC protocol is correct, secure, and Strong private.*

In Section A, we have shown that IND-CCA security alone is insufficient to prove this kind of result.

Note that in light of Corollary 10, we can implement the encryption scheme by a simulatable, PA1+ plaintext-aware, and IND-CPA secure public-key encryption scheme. Since the Cramer-Shoup [14] and Kurosawa-Desmedt [26] encryption schemes satisfy all these notions [7,24] (under certain extractor-based assumptions), any of these two schemes can be used.

Proof. First note that by Theorem 9, the encryption scheme is PA1++ plaintext-aware. Correctness is trivially induced by the correctness of the encryption scheme while security follows from IND-CCA security and [34, Theorem 19].

Therefore, we only need to prove privacy. To conduct the proof, we consider a Strong adversary $\mathcal{A}$ and construct a blinder iteratively. That is, we construct a sequence of partial blinders $B_1, \ldots, B_5$ and let $\mathcal{A}_i = \mathcal{A}_{i-1}^{B_i}$ with $\mathcal{A}_0 = \mathcal{A}$. The final blinder for $\mathcal{A}$ is $B = B_1 \circ \cdots \circ B_5$. By showing that the outcome of $\mathcal{A}_i$ and $\mathcal{A}_{i+1}$ are computationally indistinguishable, we deduce that B is indeed a full blinder for $\mathcal{A}$. So, the scheme is Strong private.

Game 0. Let Game 0 be the privacy game played by the adversary $\mathcal{A}_0$.

Game 1. We let Game 1 denote the privacy game performed by an adversary who simulates every RESULT on a session π with a transcript (a,c), such that c that has been obtained by a previous $c' = \text{SENDTAG}(\mathsf{vtag}, a')$ query. If $a \neq a'$, for sure c does not decrypt to something containing a, so the answer to $\text{RESULT}(\pi)$ must be 0. The simulation is easy and perfect. In the other case, that is, if $a = a'$, the decryption of c will be parsed to a matching challenge a and some entry $\mathsf{ID}\|K_{\mathsf{ID}}$ which is in the database if and only if vtag is legitimate. Fortunately, the blinder has access to this latter information as it is returned in the response of the DRAWTAG oracle query drawing vtag. Again, the simulation is easy and perfect. This fully defines B_1 and we deduce that $\Pr[\mathcal{A}_0 \text{ wins}] = \Pr[\mathcal{A}_0^{B_1} \text{ wins}]$. We can thus define the adversary $\mathcal{A}_1$ that never queries RESULT on an instance π in which the response c was produced by a previous SENDTAG query.

Game 2. In this game, we make all SENDTAG queries being simulated by a partial blinder B_2. To achieve this, we let r be number of SENDTAG queries and make a sequence of hybrid blinders $B_2^0, \ldots, B_2^r$ in which B_2^i simulates the i first SENDTAG queries. Note that B_2^0 does not make any simulation so $\mathcal{A}_1^{B_2^0}$ is exactly $\mathcal{A}_1$ and that B_2^r is a partial blinder

for all SENDTAG queries. We define the hybrid B_2^i by simulating the i first encountered SENDTAG queries by encrypting random strings of same length as $\mathsf{ID}\|K_{\mathsf{ID}}\|a$.

To prove that $\mathcal{A}_1^{B_2^{i-1}}$ and $\mathcal{A}_1^{B_2^i}$ have computationally indistinguishable distributions, we construct an adversary $\mathcal{C}$ playing the IND-CCA game. Adversary $\mathcal{C}$ receives the public key and runs $\mathcal{A}_1^{B_2^{i-1}}$ or $\mathcal{A}_1^{B_2^i}$, depending on the bit of the indistinguishability game, while simulating the RFID system, except the i-th SENDTAG query. For that, $\mathcal{C}$ must simulate the environment for $\mathcal{A}_1^{B_2^{i-1}}/\mathcal{A}_1^{B_2^i}$. Since all algorithms and oracles of the scheme, except for RESULT, do not require the secret key, $\mathcal{C}$ can easily perform the simulation by itself. Regarding the RESULT interface, $\mathcal{C}$ just queries a decryption oracle and checks whether the decrypted message matches.

The first $i-1$ SENDTAG queries are made to the IND-CCA challenger in a real-or-random version. The challenge ciphertext c in the IND-CCA game is the answer from the challenger. It is either a real answer (as in the $\mathcal{A}_1^{B_2^{i-1}}$ simulation) or a simulated one (as in the $\mathcal{A}_1^{B_2^i}$ simulation). Note that no RESULT query is made on the session in which the adversary sent c (this case has been taken care of in Game 1). So, $\mathcal{C}$ perfectly simulates either the game for $\mathcal{A}_1^{B_2^{i-1}}$ or the game for $\mathcal{A}_1^{B_2^i}$ and is an IND-CCA adversary. Since $\mathcal{C}$ produces the output of $\mathcal{A}_1^{B_2^{i-1}}/\mathcal{A}_1^{B_2^i}$, we obtain that $\left|\Pr[\mathcal{A}_1^{B_2^i} \text{ wins}] - \Pr[\mathcal{A}_1^{B_2^{i+1}} \text{ wins}]\right| \leq \mathsf{Adv}^{\mathsf{IND\text{-}CCA}}(k)$, and it results that $\left|\Pr[\mathcal{A}_1 \text{ wins}] - \Pr[\mathcal{A}_1^{B_2} \text{ wins}]\right| \leq r \cdot \mathsf{Adv}^{\mathsf{IND\text{-}CCA}}(k)$, which is negligible as r is polynomially bounded and the scheme is IND-CCA secure.

At this point, we can legitimately consider an adversary $\mathcal{A}_2$ who makes no SENDTAG queries.

Game 3. We now simulate all remaining RESULT queries. To do so, we construct an adversary $\mathcal{E}$ playing the PA1++ game. The way $\mathcal{B}_3$ simulates RESULT will come from the $\mathcal{E}$ construction.

$\mathcal{E}$ takes the public key then simulates $\mathcal{A}_2$ interacting with the RFID system. Recall that, like in Game 2, the algorithms and oracles of the scheme do not depend on the secret key, except for the RESULT queries that will be treated hereafter. We let $\mathcal{E}$ simulate the RFID system to $\mathcal{A}_2$, handling her queries as follow:

- Assuming w.l.o.g. that session identifiers are based on a counter, LAUNCH is deterministically computed by $\mathcal{E}$.
- Upon a CREATETAG(ID) query from $\mathcal{A}_2$, $\mathcal{E}$ inserts $(\mathsf{ID}, -)$ in a table DB_1 if the query asks for a legitimate tag. Otherwise, it inserts $(\mathsf{ID}, -)$ in a table DB_0. This is deterministic.
- $\mathcal{E}$ simulates SENDREADER $\rightarrow a$ queries by asking the oracle O_S to sample from the uniform distribution over $\{0,1\}^\alpha$. It then forwards the received answer a to $\mathcal{A}_2$. This is non-deterministic but only requires uniformly distributed independent bits.
- DRAWTAG(Samp) queries are handled by asking the randomness oracle O_S to sample from the distribution specified by Samp to get one or more random ID. If any of the returned identifiers corresponds to a drawn tag, $\mathcal{E}$ outputs $\perp$. Otherwise, it generates, deterministically and for each returned ID_i, a fresh vtag_i and inserts

the pair $(\mathsf{vtag}_i, \mathsf{ID}_i)$ in the table $\mathcal{T}$. After that, it sets the bit b_i to 1 if ID_i is legitimate, or to 0 otherwise. At last, it returns $(\mathsf{vtag}_1, b_1, \ldots, \mathsf{vtag}_n, b_n)$ to $\mathcal{A}_2$. This is non-deterministic but requires inverse samplable distributions.

- CORRUPT(vtag) makes $\mathcal{E}$ reveal $\mathsf{ID} = \mathcal{T}(\mathsf{vtag})$. Moreover, $\mathcal{E}$ looks for the entry $(\mathsf{ID}, K_{\mathsf{ID}})$ in DB_0 and DB_1. If that corresponding entry contains a K_{ID} different from $'-'$, then it returns it. Otherwise, it queries O_S to sample from the uniform distribution over $\{0,1\}^\kappa$ and assigns the answer to K_{ID}. It subsequently updates the entry $(\mathsf{ID}, -)$ to $(\mathsf{ID}, K_{\mathsf{ID}})$ and returns this last pair as its answer. We further assume that whenever the tag ID is a legitimate one, $\mathcal{E}$ inserts the entry $(\mathsf{ID}, K_{\mathcal{T}(\mathsf{vtag})})$ in a table $\mathcal{T}_{\mathcal{E}}$. This is non-deterministic but only requires uniformly distributed independent bits. Note that non-corrupted tags have no preset K_{ID} key.
- To simulate the RESULT(π) oracle for a reader instance π with transcript (a,c), $\mathcal{E}$ sends c to the decryption oracle, checks that the recovered plaintext is of the form $\mathsf{ID}\|K_{\mathsf{ID}}\|a$, that it matches a $\mathsf{ID}\|K_{\mathsf{ID}} \in \mathsf{DB}_1$. (Note that this implies that tag ID, has been corrupted, and has key K_{ID}.) If this is the case, the answer to RESULT must be 1, otherwise, the simulated answer is 0. Note that when the output of the $\mathcal{E}$ regarding a RESULT query is 1, the genuine RESULT query would also have answered 1. So, this simulation is correct. Errors in the simulation only occur when $\mathcal{E}$ predicts 0 and the genuine RESULT query would also have outputted 1 in a session without matching conversation. Clearly, the failure of one of $\mathcal{E}$'s simulations corresponds to the happening of the event that there is a legitimate and uncorrupted tag which was identified by a session π which received a c which was not produced by any SENDTAG query. This implies that the event E that $\mathcal{A}_2$ wins the security game holds. In other words, $\left|\Pr[\mathcal{A}_2 \text{ wins}] - \Pr\left[\mathcal{A}_2^{\mathcal{E}} \text{ wins}\right]\right| \leq \Pr[E]$. Since it was shown that the PKC protocol is secure, this is negligible.

Since we assumed the encryption scheme to be PA1++ plaintext-aware, we can use the plaintext extractor $\mathcal{E}^\star$ of $\mathcal{E}$ to replace the decryption oracle without significantly altering the outcome distribution. However, $\mathcal{E}^\star$ requires the view of $\mathcal{E}$ instead of the view of $\mathcal{A}_2$, so we cannot use it as an extractor for $\mathcal{A}_2$. Fortunately, it is possible to reconstruct that view given the adversary's random tape and its queries. At first, we note that $\mathcal{E}$'s random coins are only used to initialize $\mathcal{A}_2$. Furthermore, all the randomness $\mathcal{E}$ obtains from O_S to process CORRUPT queries is revealed to $\mathcal{A}_2$. Moreover, since $\mathcal{A}_2$ is simulatable, we can use the algorithm $\mathcal{A}_2'$, induced by Definition 4, to reconstruct, from $\mathcal{A}_2$'s view, a table $\mathcal{T}'$ indistinguishable from $\mathcal{T}$. Since this table lists all the mappings between real and virtual identifiers, it is straightforward to reconstruct a randomness for $\mathcal{E}$ that she received to process the DRAWTAG queries using the Samp^{-1} algorithms corresponding to the sampling queries of $\mathcal{A}_2$. We let this whole operation be carried by a polynomial-time algorithm $\mathcal{V}$ that takes as input the view of $\mathcal{A}_2$ and uses $\mathcal{A}_2'$ to reconstruct a view of $\mathcal{E}$, i.e., it is such that $\mathcal{V}(\mathsf{view}_{\mathcal{A}_2})$ and $\mathsf{view}_{\mathcal{E}}$ are indistinguishable. It follows that $\mathcal{E}^\star(pk, \cdot, \mathcal{V}(\mathsf{view}_{\mathcal{A}_2}))$ and $\mathcal{E}^\star(pk, \cdot, \mathsf{view}_{\mathcal{E}})$ are indistinguishable.

At this point, we are able to define B_3, the partial blinder for RESULT queries. Similarly to $\mathcal{E}$, we assume that B_3 maintains a table $\mathcal{T}_{B_3}$ containing a list of pairs $(\mathsf{ID}, K_{\mathsf{ID}})$ for corrupted legitimate tags. In order to simulate a RESULT query on an instance π of transcript (a,c), the blinder proceed as follow.

1. First, the blinder calls $\mathcal{E}^{\star}(pk, c, \mathcal{V}(\mathsf{view}_{\mathcal{A}_2}))$ to get $\mathsf{Dec}_{sk}(c) = \mathsf{ID}\|K_{\mathsf{ID}}\|a'$.
2. Then it verifies that $a = a'$ and outputs 0 in case of failure. Otherwise, it continues.
3. At last, it outputs 1 if the pair $\mathsf{ID}\|K_{\mathsf{ID}}$ is listed in $\mathcal{T}_{B_3}$, and 0 otherwise.

The probability that Step 1 fails can be expressed as a distinguisher advantage of the PA1++ game or between $\mathcal{V}(\mathsf{view}_{\mathcal{A}_2})$ and $\mathsf{view}_{\mathcal{E}}$, so

$$\left|\Pr\left[\mathcal{A}_2^{B_3} \text{ wins}\right] - \Pr\left[\mathcal{A}_2^{\mathcal{E}} \text{ wins}\right]\right| \leq \mathsf{Adv}^{\mathsf{PA1++}}(k) + \mathsf{negl}(k).$$

At the same time, Step 3 fails when the event E occurs, so using triangle inequalities we conclude that

$$\left|\Pr[\mathcal{A}_2 \text{ wins}] - \Pr\left[\mathcal{A}_2^{B_3} \text{ wins}\right]\right| \leq \mathsf{Adv}^{\mathsf{PA1++}}(k) + \Pr[E] + \mathsf{negl}(k).$$

Recalling that E occurs with negligible probability and that the scheme is PA1++ plaintext-aware, the quantity above becomes negligible. Hence, B_3 describes a successful blinder for the RESULT oracle.

Game 4. In this game, we get rid of $\text{SENDREADER}(\pi) \to a$ queries. This can easily be achieved by constructing a blinder B_4 that returns uniformly distributed values from the set $\{0,1\}^{\alpha}$. We further get rid of the $\text{SENDREADER}(\pi, c)$ queries in a trivial way as they return nothing and are not followed by any $\text{RESULT}(\pi)$ query. Clearly, simulation is perfect as both distributions are perfectly indistinguishable. Hence, $\Pr[\mathcal{A}_3 \text{ wins}] = \Pr[\mathcal{A}_3^{B_4} \text{ wins}]$.

Game 5. Finally, we have an adversary $\mathcal{A}_4$ who only produces LAUNCH queries. We can trivially simulate the Them. It follows that $\Pr[\mathcal{A}_4 \text{ wins}] = \Pr[\mathcal{A}_4^{B_5} \text{ wins}]$. In the end, we have obtained an adversary $\mathcal{A}_5 = \mathcal{A}^B$, with $B = B_1 \circ \cdots \circ B_5$, who does not produce any oracle query that is such that $\left|\Pr[\mathcal{A} \text{ wins}] - \Pr[\mathcal{A}^B \text{ wins}]\right| = \mathsf{negl}(k)$. The scheme is thus Strong private. □

6 Conclusion

We updated the Vaudenay model for RFID privacy. Our model now makes it possible to achieve strong privacy. Actually, we proved that the regular PKC protocol with an IND-CCA and PA1+ secure cryptosystem achieves it. We have further shown that IND-CCA security alone could fail to reach this level of privacy. This shows a separation between our privacy model and the one from [22]. However, the question whether this separation is significant remains open.

Acknowledgement. The authors would like to than Sherman Chow for his valuable help in the final version of this paper.

References

1. Armknecht, F., Sadeghi, A.-R., Scafuro, A., Visconti, I., Wachsmann, C.: Impossibility Results for RFID Privacy Notions. In: Gavrilova, M.L., Tan, C.J.K., Moreno, E.D. (eds.) Transactions on Computational Science XI, Part II. LNCS, vol. 6480, pp. 39–63. Springer, Heidelberg (2010)

2. Avoine, G.: Cryptography in radio frequency identification and fair exchange protocols. PhD thesis, EPFL, Lausanne, Switzerland. Thesis N° 3407 (2005)
3. Avoine, G., Dysli, E., Oechslin, P.: Reducing Time Complexity in RFID Systems. In: Preneel, B., Tavares, S. (eds.) SAC 2005. LNCS, vol. 3897, pp. 291–306. Springer, Heidelberg (2006)
4. Bellare, M., Desai, A., Pointcheval, D., Rogaway, P.: Relations among Notions of Security for Public-Key Encryption Schemes. In: Krawczyk, H. (ed.) CRYPTO 1998. LNCS, vol. 1462, pp. 26–45. Springer, Heidelberg (1998)
5. Bellare, M., Palacio, A.: Towards Plaintext-Aware Public-Key Encryption Without Random Oracles. In: Lee, P.J. (ed.) ASIACRYPT 2004. LNCS, vol. 3329, pp. 48–62. Springer, Heidelberg (2004)
6. Bellare, M., Rogaway, P.: Optimal Asymmetric Encryption. In: De Santis, A. (ed.) EUROCRYPT 1994. LNCS, vol. 950, pp. 92–111. Springer, Heidelberg (1995)
7. Birkett, J.: On Plaintext-Aware Public-Key Encryption Schemes. PhD thesis, Royal Holloway, University of London (2010)
8. Birkett, J., Dent, A.W.: Relations Among Notions of Plaintext Awareness. In: Cramer, R. (ed.) PKC 2008. LNCS, vol. 4939, pp. 47–64. Springer, Heidelberg (2008)
9. Bogdanov, A., Knudsen, L.R., Leander, G., Paar, C., Poschmann, A., Robshaw, M., Seurin, Y., Vikkelsoe, C.: PRESENT: An Ultra-Lightweight Block Cipher. In: Paillier, P., Verbauwhede, I. (eds.) CHES 2007. LNCS, vol. 4727, pp. 450–466. Springer, Heidelberg (2007)
10. Bogdanov, A., Leander, G., Paar, C., Poschmann, 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)
11. Burmester, M., van Le, T., de Medeiros, B.: Provably Secure Ubiquitous Systems: Universally Composable RFID Authentication Protocols. In: SecureComm 2006, Baltimore, Maryland, USA. IEEE Press (2006)
12. Canard, S., Coisel, I., Etrog, J., Girault, M.: Privacy-preserving RFID systems: Model and constructions. Cryptology ePrint Archive, Report 2010/405 (2010), http://eprint.iacr.org/
13. Canetti, R.: Universally composable security: A new paradigm for cryptographic protocols. Cryptology ePrint Archive, Report 2000/067 (2000), http://eprint.iacr.org/
14. Cramer, R., Shoup, V.: A Practical Public Key Cryptosystem Provably Secure against Adaptive Chosen Ciphertext Attack. In: Krawczyk, H. (ed.) CRYPTO 1998. LNCS, vol. 1462, pp. 13–25. Springer, Heidelberg (1998)
15. Damgård, I., Pedersen, M.Ø.: RFID Security: Tradeoffs between Security and Efficiency. In: Malkin, T. (ed.) CT-RSA 2008. LNCS, vol. 4964, pp. 318–332. Springer, Heidelberg (2008)
16. Deng, R.H., Li, Y., Yao, A.C., Yung, M., Zhao, Y.: A new framework for RFID privacy. Cryptology ePrint Archive, Report 2010/059 (2010), http://eprint.iacr.org/
17. Deng, R.H., Li, Y., Yung, M., Zhao, Y.: A New Framework for RFID Privacy. In: Gritzalis, D., Preneel, B., Theoharidou, M. (eds.) ESORICS 2010. LNCS, vol. 6345, pp. 1–18. Springer, Heidelberg (2010)
18. Dent, A.W.: The Cramer-Shoup Encryption Scheme Is Plaintext Aware in the Standard Model. In: Vaudenay, S. (ed.) EUROCRYPT 2006. LNCS, vol. 4004, pp. 289–307. Springer, Heidelberg (2006)
19. Feldhofer, M., Dominikus, S., Wolkerstorfer, J.: Strong Authentication for RFID Systems Using the AES Algorithm. In: Joye, M., Quisquater, J.-J. (eds.) CHES 2004. LNCS, vol. 3156, pp. 357–370. Springer, Heidelberg (2004)
20. Goldwasser, S., Micali, S.: Probabilistic encryption. Journal of Computer and System Sciences 28(2), 270–299 (1984)
21. Hein, D., Wolkerstorfer, J., Felber, N.: ECC Is Ready for RFID – A Proof in Silicon. In: Avanzi, R.M., Keliher, L., Sica, F. (eds.) SAC 2008. LNCS, vol. 5381, pp. 401–413. Springer, Heidelberg (2009)

22. Hermans, J., Pashalidis, A., Vercauteren, F., Preneel, B.: A New RFID Privacy Model. In: Atluri, V., Diaz, C. (eds.) ESORICS 2011. LNCS, vol. 6879, pp. 568–587. Springer, Heidelberg (2011)
23. Ishai, Y., Kumarasubramanian, A., Orlandi, C., Sahai, A.: On Invertible Sampling and Adaptive Security. In: Abe, M. (ed.) ASIACRYPT 2010. LNCS, vol. 6477, pp. 466–482. Springer, Heidelberg (2010)
24. Jiang, S., Wang, H.: Plaintext-Awareness of Hybrid Encryption. In: Pieprzyk, J. (ed.) CT-RSA 2010. LNCS, vol. 5985, pp. 57–72. Springer, Heidelberg (2010)
25. Juels, A., Weis, S.A.: Defining strong privacy for RFID. In: PerCom Workshops 2007, pp. 342–347. IEEE Computer Society (2007)
26. Kurosawa, K., Desmedt, Y.: A New Paradigm of Hybrid Encryption Scheme. In: Franklin, M. (ed.) CRYPTO 2004. LNCS, vol. 3152, pp. 426–442. Springer, Heidelberg (2004)
27. van Le, T., Burmester, M., de Medeiros, B.: Universally composable and forward-secure RFID authentication and authenticated key exchange. In: ASIACCS 2007, pp. 242–252. ACM (2007)
28. Molnar, D., Wagner, D.: Privacy and security in library RFID: issues, practices, and architectures. In: CCS 2004, pp. 210–219. ACM (2004)
29. 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)
30. Ohkubo, M., Suzuki, K., Kinoshita, S.: RFID privacy issues and technical challenges. Commun. ACM 48(9), 66–71 (2005)
31. Ouafi, K., Phan, R.C.-W.: Traceable Privacy of Recent Provably-Secure RFID Protocols. In: Bellovin, S.M., Gennaro, R., Keromytis, A.D., Yung, M. (eds.) ACNS 2008. LNCS, vol. 5037, pp. 479–489. Springer, Heidelberg (2008)
32. Paise, R.-I., Vaudenay, S.: Mutual authentication in RFID: security and privacy. In: Proceedings of the ASIACCS 2008, pp. 292–299. ACM (2008)
33. Shamir, A.: SQUASH – A New MAC with Provable Security Properties for Highly Constrained Devices Such as RFID Tags. In: Nyberg, K. (ed.) FSE 2008. LNCS, vol. 5086, pp. 144–157. Springer, Heidelberg (2008)
34. Vaudenay, S.: On Privacy Models for RFID. In: Kurosawa, K. (ed.) ASIACRYPT 2007. LNCS, vol. 4833, pp. 68–87. Springer, Heidelberg (2007)

A IND-CCA Security Is Not Sufficient for Strong Privacy

We define $(\mathsf{KeyGen}^1, \mathsf{Enc}^1, \mathsf{Dec}^1)$, a variant of the Goldwasser-Micali cryptosystem [20] as follows.

- $\mathsf{KeyGen}^1(1^k)$. Pick an RSA modulus $N = pq$, i.e, s.t. p and q are primes, and $y, z \in \mathbb{Z}_N^\star$ such that $\left(\frac{y}{N}\right) = +1$, $\left(\frac{y}{p}\right) = -1$, and $\left(\frac{z}{N}\right) = +1$. The scheme's key pair is $pk^1 = (N, y, z)$ and $sk^1 = p$.
- $\mathsf{Enc}^1_{pk^1}(b) = y^b r^2 \mod N$ where $b \in \{0,1\}$ and $r \in_R \mathbb{Z}_N^\star$.
- $\mathsf{Dec}^1_{sk^1}(c) = b$ such that $(-1)^b = \left(\frac{c}{p}\right)$.

Note that z in the public key is unused. Further note that $z \cdot \mathsf{Enc}^1{}_{pk^1}(b) \bmod N$ is a valid encryption of either b or $\bar{b} = 1 - b$ depending on $\left(\frac{z}{p}\right)$ which is unknown for someone holding the public key. Let $(\mathsf{KeyGen}^0, \mathsf{Enc}^0, \mathsf{Dec}^0)$ denote an IND-CCA secure encryption scheme, we define $(\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ as follows.

- **KeyGen.** Run $(sk^0, pk^0) \leftarrow \mathsf{KeyGen}^0(1^k)$ and $(sk^1, pk^1) \leftarrow \mathsf{KeyGen}^1(1^k)$. The scheme's key pair is $pk = (pk^0, pk^1)$ and $sk = (sk^0, sk^1)$.
- **Encrypt.** To encrypt, set $\mathsf{Enc}_{pk}(x) = \mathsf{Enc}^0_{pk^0}\left(\mathsf{Enc}^1_{pk^1}(x_1), \ldots, \mathsf{Enc}^1_{pk^1}(x_n)\right)$ where $x_1, \ldots, x_n$ is the binary decomposition of x.
- **Decrypt.** To decrypt, compute $\mathsf{Dec}_{sk}(c) = \mathsf{Dec}^1_{sk^1}(t_1), \ldots, \mathsf{Dec}^1_{sk^1}(t_n)$ where $t_1, \ldots, t_n = \mathsf{Dec}^0_{sk^0}(c)$.

We can easily see that (KeyGen, Enc, Dec) is IND-CCA secure. It is not plaintext-aware since $\mathsf{Enc}^0_{pk^0}\left(z \cdot \mathsf{Enc}^1_{pk^1}(x_0) \bmod N, \ldots, z \cdot \mathsf{Enc}^1_{pk^1}(x_{n-1}) \bmod N\right)$ is a valid encryption of either $x_0, \ldots, x_{n-1}$ or $\overline{x_0}, \ldots, \overline{x_{n-1}}$ depending on $\left(\frac{z}{p}\right)$. To figure out whether this encrypts x or $\bar{x}$ would require to solve the quadratic residuosity problem, which is supposedly a hard problem.

Consider the PKC protocol of Fig. 1 using the above IND-CCA secure public-key encryption scheme.

Finally, the following Strong adversary defeats privacy.

1: CREATETAG(ID)
2: $\mathsf{v} \leftarrow$ DRAWTAG(ID)
3: $S_{\mathsf{ID}} \leftarrow$ CORRUPT(v)
4: $\pi \leftarrow$ LAUNCH
5: $a \leftarrow$ SENDREADER($\emptyset, \pi$)
6: Set $x = S_{\mathsf{ID}} \| a = x_1, \ldots, x_n$
7: $s = z \cdot \mathsf{Enc}^1_{pk^1}(x_1), \ldots, z \cdot \mathsf{Enc}^1_{pk^1}(x_n)$
8: $c \leftarrow \mathsf{Enc}^0_{pk}(s)$
9: SENDREADER(c, π)
10: $b \leftarrow$ RESULT(π)
11: Output b

Clearly, an adversary outputs 1 if and only if $\left(\frac{z}{p}\right) = +1$. Therefore, a blinder that follows the same distribution would break the quadratic residuosity problem, i.e., the problem of distinguishing quadratic residues from non-quadratic residues. So, the PKC protocol based on this cryptosystem is strong-private in the model from [22] but is not in our model, assuming that quadratic residuosity is a hard problem. This proves the separation between privacy from [22] and our strong privacy. Indeed, we have shown that the PKC protocol based on an IND-CCA cryptosystem may still leak some non-simulatable information although it is strong private in the sense of [22].