On Privacy for RFID

Serge Vaudenay

EPFL
CH-1015 Lausanne, Switzerland
http://lasec.epfl.ch

Abstract. Many wearable devices identify themselves in a pervasive way. But at the same time, people want to remain anonymous. Modeling anonymity and unlinkability in identification protocols is a delicate issue. In this paper, we revisit the privacy model from Asiacrypt 2007. We show how to achieve forward-privacy (in the V07 sense) using an IND-CCA secure cryptosystem with the PKC protocol. We review the impossibility result of strong privacy and the model extension from CANS 2012 to reach strong privacy (in the OV12 sense) using the PKC protocol with plaintext awareness. We also discuss on the simplified model from ESORICS 2011 and achieve strong-privacy (in the HPVP11 sense) using IND-CCA security only. Finally, we apply these results to add privacy protection in distance bounding protocols.

1 Introduction

People wear more and more passive RFID devices, from identity documents or credit cards to smart socks[1]. These devices typically identify with a traceable ID number to whichever device trying to scan them. Clearly, this opens opportunities for malicious people to tracing people based on their ID or to check how frequently they changed their socks.

Concretely, an RFID system defines a set of legitimate tags, readers, and a communication protocol between a tag and a reader. Sometimes, the protocol may also require the reader to communicate with a centralized (authority) server. The input of the tag consists of an internal state (which may contain a certificate and a tag-specific secret key). The private output of the tag may be a new state (for stateful protocols). The input of the reader may contain a root certificate or a database of the secret keys of legitimate tags. The private output of the reader is the ID of the tag. So, the purpose of the RFID protocol is to *identify* the tag to the reader. At the same time, the identification must be secure (i.e., it must *authenticate* the tag). A typical secondary issue is that the protocol must keep *privacy*. I.e., no adversary could infer any non-trivial information about the ID of the tag from the protocol.

A typical example is the GSM protocol: a GSM phone holding a SIM card identifies for the first time to the network cell in clear using its IMSI number. This

[1] which tell when they stinks so that we can wash them and also how to pair them again after washing.

tells the cell to which home network the SIM card belongs and how to get means to open a secure communication channel with it. Once this is done, the cell gives a pseudonym TMSI to the phone which will be used for the next identification. The TMSI is renewed through the secure channel as often as required. Security is based on symmetric cryptography. Privacy is clearly ineffective for the first connection. Furthermore, active attacks can break the synchronization between a phone and the cell, forcing the phone to identify in clear again. So, privacy protection is very weak in this case.

One difficult task when defining privacy is to model the capabilities of the adversary and his goal. In the early days of secure RFID protocols, some simple protocols were proposed with privacy protection [19,30,45]. These protocols assumed the adversary could not *corrupt* legitimate tags to get their internal state. A step further was made by the Ohkubo-Suzuki-Kinoshita protocol (OSK) [32,33] (see also [17,34]) to model *forward privacy*, i.e., such that uncorrupted tags running the protocol could not be identified in the future after they become corrupted. An early model for RFID privacy was proposed by Avoine-Dysli-Oechslin [3,2]. Their, the adversary chooses two tags; one of them is drawn at random and they must guess which one after interacting with this tag and the reader concurrently. The model was later refined by Juels and Weis [25] by telling the adversary when the reader succeeds to identify a legitimate tag. This information, which is the *result* of the protocol, models a side-channel information that the adversary could exploit.

The most complete privacy model (called the V07 model herein) appeared at Asiacrypt 2007 [39]. It is based on simulation. Essentially, the adversary plays with tags and readers concurrently. He specifies the distribution following which tags are drawn. His goal is to infer some information about identities, but the information must be non-trivial in the sense that it cannot be inferred by simulating the protocol messages. The V07 model defines a 2×4-matrix of privacy levels, depending on whether the adversary has access to the result of the protocol (which are called *narrow* and *wide* adversaries), and depending on how corruption is feasible. With no corruption, we have a *weak* adversary. With corruption which can only happen at the end of the game, we have a *forward* adversary (to address forward privacy). With corruption which destroy tags (i.e., the adversary can no longer play with a corrupted tag), we have a *destructive* adversary. With corruption which happens with no such restriction, we have a *strong* adversary. In [39], a secure RFID protocol protecting both narrow-strong and wide-forward privacy was constructed based on a chosen-ciphertext-secure (IND-CCA) cryptosystem. This protocol based on a cryptosystem is called PKC herein. It was further shown that wide-weak privacy was achievable with just a pseudorandom function (PRF).

In [39], it was proven that an RFID protocol could not offer at the same time wide-destructive and narrow-strong privacy. In particular, wide-strong privacy is impossible. The impossibility result was however quite technical, more showing that the privacy definition was overly restrictive than showing a concrete impossibility result. This made Ng *et al.* [31] propose the notion of *wise* adversary, i.e.,

an adversary who does not ask questions for which he knows the answer. The definition from [31] was not formal enough to be usable, but this made Ouafi and Vaudenay [35] to refine the V07 model by letting the simulator know the input of the adversary (so, know what answer the adversary expects). This model is called the OV12 model herein. Then, they have shown that the PKC protocol is wide-strong-private when the cryptosystem is further plaintext-aware (PA).

Finally, Hermans *et al.* [23] proposed another privacy model (called the HPVP11 model herein) in which the game uses a left-or-right oracle and corruption is not made on anonymous tags. Surprisingly, this makes the notion of "trivial information" obtained by adversaries easy to specify and allows to get rid of the simulation. In addition, the PKC model is shown to be wide-strong private in the HPVP11 sense with only IND-CCA (and not PA) security as an assumption. This makes the model much more easy to use. However, it was shown in [35] that the OV12 model is strictly stronger that the HPVP11 model in the sense that we can construct a protocol being HPVP11-wide-strong-private but not OV12-wide-strong-private. However, the proof that the protocol is not OV12-private does not yield any convincing real privacy threat. So, the definitions of the OV12 model may be too restrictive and the HPVP11 notion of wide-strong privacy may certainly be enough.

With distance bounding (DB) protocols, the tag wants to prove its proximity to the reader. There are symmetric DB protocols in which the tag and the reader share a secret and public-key DB protocols in which the tag (and sometimes the reader) has a public/private key pair to authenticate. Modeling security for DB protocols is not easy.

The first complete security models and provably secure protocols were independently proposed by Boureanu *et al.* [8,9,10,11,12] and by Fischlin and Onete [18,20]. None of these protocols were optimal but by combining both ideas we obtain the DBopt protocols [7,26]. In these protocols, the tag and the reader must share a symmetric secret.

Regarding public-key DB protocols, the DBopt model was adapted for public-key DB in [41,42,43]. Not many public-key DB protocols exist. We list them in Table 1 with the known proven security/insecurity results (see [42,43] for details). The table includes *Man-in-the-Middle* security (MiM), *Distance Fraud* (DF), *Distance Hijacking* (DH), *Collusion Fraud* (CF), *wide-Privacy* (Privacy), and *wide-Strong Privacy* (Strong Privacy). Note that DBPK-Log [15] is broken [4]. As we can see, only the HPO protocol and privDB [24,42] provide some form of privacy. HPO [24] relies on ad-hoc assumptions. Futhermore, it does not provide wide-strong privacy (as shown in [44]). So far, only privDB [42] provides wide-strong privacy. We added in the table the eProProx protocol which is proposed in this paper. It extends ProProx by providing wide-strong privacy.

2 The V07 Model and the OV12 Extension

We describe here the V07 model [39] and the OV12 extension [35], as presented in [40]. The V07 model [39] from Asiacrypt 2007 follows up some joint work dur-

Table 1. Existing Public-Key Distance Bounding Protocols

protocol	MiM	DF	DH	CF	Privacy	Strong privacy
Brands-Chaum [14]	secure	secure	insecure	insecure	insecure	insecure
DBPK-Log [15]		insecure		insecure	insecure	insecure
HPO [24]	secure	secure		insecure	secure	insecure
GOR [21]	secure	secure	insecure	insecure	insecure	insecure
privDB [42]	secure	secure	secure	insecure	secure	secure
ProProx [43]	secure	secure	secure	secure	insecure	insecure
eProProx (this paper)	secure	secure	secure	secure	secure	secure

ing the MSc Thesis of Bocchetti [6]. The results were also announced in [38]. For completeness, we also indicate that some extension with reader authentication was proposed in [36] ... but with a few incorrect results as shown by Armknecht *et al.* [1].

The V07 model considers a multiparty setting with a malicious adversary and several concurrent honest tags and honest readers which can be activated by the adversary. All readers are assumed to be front ends of a secure server which contains a database. The communication from readers to the central database is assumed to be secure. Although all tags are honest, some belong to the system (these tags are sometimes called *legitimate*) and some do not. The adversary can initiate the creation of new tags (in the system or not). He controls the communications to every participants. Furthermore, the access to random tags in practice is modelled by having the adversary being able to draw anonymous tags with a chosen probability distribution.

RFID system. More concretely, there is an algorithm

$$\mathsf{SetupReader} \rightarrow (K_S, K_P)$$

producing a key pair. The key K_S is secret. It can be used by readers. The key K_P is public and used to create tags. Indeed, there is an algorithm

$$\mathsf{SetupTag}_{K_P}(\mathsf{ID}) \rightarrow (\mathsf{data}, S)$$

producing an initial state S for the tag and some data so that the entry $(\mathsf{ID}, \mathsf{data})$ is inserted into the database when the tag is meant to belong to the system. In addition to SetupReader and SetupTag, an RFID system specifies an interactive protocol between a tag and a reader. The tag has as input its current state S and as output a value S' which becomes the new state of the tag. The reader has as input K_S and as output some value out. If $\mathsf{out} = \perp$, we say that the identification failed. Otherwise, out shall corresponds to the ID of the tag.

Game. In a game, after SetupReader was run, the adversary receives K_P and can access to an oracle

$$\mathsf{CreateTag}(\mathsf{ID}, b)$$

which runs $\mathsf{SetupTag}_{K_P}(\mathsf{ID}) \rightarrow (\mathsf{data}, S)$. Additionally, if $b = 1$, the oracle inserts $(\mathsf{ID}, \mathsf{data})$ into the database. So, $b = 1$ means that the tag will be recognized as belonging to the system but $b = 0$ can be used to create "foreign tags".

The adversary can also access to the

$$\mathsf{DrawTag}(D) \rightarrow (\mathsf{vtag}_1, b_1, \ldots, \mathsf{vtag}_n, b_n)$$

with a chosen distribution. This oracle draws a vector $(\mathsf{ID}_1, \ldots, \mathsf{ID}_n)$ following the chosen distribution D. If any tag ID_i is already drawn or was not created, the oracle returns $\perp$. Otherwise, it defines some fresh random identifiers vtag_i and sets b_i to 1 if and only if ID_i belongs to the system. Additionally, the oracle adds the matching $\mathsf{vtag}_i \leftrightarrow \mathsf{ID}_i$ in a private table $\mathcal{T}$. So, the adversary can draw anonymous tags with a chosen distribution and can see which tag belongs to the system. This assumption is realistic as practical tags often leak their version, manufacturer, and other information from which we can deduce what type of tag it is. Clearly, the drawing oracle is such that a drawn tag cannot be drawn again. However, the adversary can call a

$$\mathsf{Free}(\mathsf{vtag})$$

oracle to free the anonymous tag vtag so that it can be drawn again.

As discussed in [36,35], the oracle Free must reset the temporary memory of the anonymous tag before releasing it. This is in order to present protocol sessions to span through several anonymous tag instances.

The adversary can call a

$$\mathsf{Launch} \rightarrow \pi$$

oracle which initiates a new reader session which can be called by the identifier π.

The adversary can send messages to a launched reader π or to a drawn tag vtag as long as it has not be freed. He can call

$$\mathsf{SendReader}(m, \pi) \rightarrow m'$$

to send m to π and obtain the response m' (if any). If the reader initiates the interactive protocol and π did not start yet, m is empty. If π was not launched or if the protocol terminated on the session π, nothing is returned. He can call

$$\mathsf{SendTag}(m, \mathsf{vtag}) \rightarrow m'$$

to send m to vtag and obtain the response m' (if any). If the tag initiates the interactive protocol and vtag did not start yet, m is empty. If vtag was not drawn, or was freed, or if the protocol terminated on vtag, nothing is returned. A new session may start with vtag by calling $\mathsf{SendTag}$ again.

The adversary may use a

$$\mathsf{Result}(\pi) \rightarrow x$$

oracle which tells whether the reader protocol succeeded to identify a tag on session π. (So, $x = 0$ or 1.) If the adversary is *narrow*, this oracle cannot be used. If the adversary is *wide*, no restriction applies on using this oracle.

Finally, the adversary may use a

$$\mathsf{Corrupt}(\mathsf{vtag}) \to S$$

oracle which returns the current state of the anonymous tag vtag. As vtag can only be accessed between the time it is drawn and the time it is freed, the oracle returns nothing at any other time. If the adversary is *weak*, this oracle cannot be used. If the adversary is *forward*, only further Corrupt queries can be made after this oracle call but no other oracle can be used. If the adversary is *strong*, no restriction applies on corruption.

Matching conversation. We say that two participants have a *matching conversation* at a given time if the sequence of incoming/outgoing messages that they have seen match and are well interleaved. I.e., if the protocol transcript seen by one participant is of form

$$(t_1, \mathsf{in}_1, \mathsf{out}_1), (t_2, \mathsf{in}_2, \mathsf{out}_2), \ldots (t_n, \mathsf{in}_n, \mathsf{out}_n),$$

or (when the participant initiates the protocol)

$$(t_1, \perp, \mathsf{out}_1), (t_2, \mathsf{in}_2, \mathsf{out}_2), \ldots (t_n, \mathsf{in}_n, \mathsf{out}_n),$$

with $t_1 < \cdots < t_n$ (meaning that at time t_i, the participant received in_i and sent out_i), then the protocol transcript seen by the other participant must be

$$(t'_1, \perp, \mathsf{in}_1), (t'_2, \mathsf{out}_1, \mathsf{in}_2), \ldots (t'_n, \mathsf{out}_{n-1}, \mathsf{in}_n),$$

or

$$(t'_2, \mathsf{out}_1, \mathsf{in}_2), \ldots (t'_n, \mathsf{out}_{n-1}, \mathsf{in}_n),$$

respectively, for some $t'_1, \ldots, t'_n$ such that $t'_1 < t_1 < t'_2 < \cdots < t_{n-1} < t'_n < t_n$ (meaning that at time t'_i, the participant received out_{i-1} and sent in_i).

Correct system. The protocol is *correct* if for any game, whenever there is a matching conversation between some vtag and some π, if vtag was drawn by DrawTag with the bit b, then, except with negligible probability, the output of π is $\mathsf{out} = \perp$ if $b = 0$ and $\mathsf{out} = \mathcal{T}(\mathsf{vtag})$ if $b = 1$.

Secure system. An RFID system is *secure* if for any game, except with negligible probability, for all π which produced $\mathsf{out} = \mathsf{ID} \neq \perp$, there must exist vtag such that $\mathcal{T}(\mathsf{vtag}) = \mathsf{ID}$ and either vtag has a matching conversation with π or vtag was corrupted.

Privacy. In the privacy game, the adversary $\mathcal{A}$ plays with the oracle. When done, he receives the table $\mathcal{T}$ and produces a binary output. To identify the trivial ways to output 1, we use a simulator based on a *blinder* B. A blinder sees all oracle queries of the adversary (but cannot see the table $\mathcal{T}$) and simulates the responses of the Launch, SendReader, SendTag, and Result oracles to $\mathcal{A}$. When $\mathcal{A}$ interacts with the blinded oracles (instead of the oracles directly), we denote it by $\mathcal{A}^B$. A trivial way for $\mathcal{A}$ to output 1 is such that there exists B such that $\mathcal{A}^B$ outputs 1 with nearly the same probability as $\mathcal{A}$. Intuitively, it means that $\mathcal{A}$ learns nothing new from the protocol messages, as he could simulate them by himself. A protocol is P-private if for any adversary $\mathcal{A}$ in the class P, there exists a blinder B such that $\mathcal{A}$ and $\mathcal{A}^B$ produce the same output except with negligible probability. As an example of class P, we can consider all wide-strong adversaries.

Impossibility of wide-strong privacy in the V07 model. To prove the impossibility of wide-strong privacy by contradiction, we essentially have to make the adversary play against the blinder. Let us assume that the protocol provides wide-strong privacy. We consider a first game in which the adversary creates a legitimate tag ID_1, draws ID_1, and corrupts it to get its state S_1. Then, he runs on its own $\mathsf{SetupTag}_{K_P}(\mathsf{ID}_0) \to S_0$. Now, the adversary can simulate either tag ID_0 or tag ID_1 using their state. So, he can flip a coin b, launch a reader session π and simulate ID_b to π using S_b. Finally, the adversary calls $\mathsf{Result}(\pi)$ and gives it as an output. Clearly, correctness imposes that the result of π is b. Due to privacy, there must exist a blinder B such that from the states of the two tags S_0 and S_1 and the messages from the tag ID_b, then B can guess b. This means that we can make a second game in which we create two tags in the system, corrupt both of them to get their states S_0 and S_1, then draw one at random and play with, and use B to infer which tag was drawn. This would identify the tag, but there is no blinder to do so. So, there is a contradiction. The crucial point in this argument is that the adversary in the first game knows which tag he simulates and makes the Result guess it. So, a blinder must simulate this guess.

V07 vs OV12. A big difference between the V07 and OV12 models is that in OV12, the blinder can use the view of the adversary as input. So, he can simulate the internal computations of the adversary and somehow "read his thoughts". This is an essential technique used with plaintext awareness (PA). Essentially, whenever the adversary issues a ciphertext, we can use a plaintext extractor on the view of the adversary to see what was encrypted. With the previous impossibility result in the V07 model, we can see that now, in the first game, the blinder could now read the bit b from the thoughts of the adversary and no longer need to guess it from the states and messages. So, the argument of impossibility does not hold in the OV12 model.

There are also tricky issues about the DrawTag oracle when the number n of tags to be drawn is not logarithmic. For instance, if n is linear, the vector spans in a set of exponential size. So, the representation of the input distribution D can be large. In [35], it is specified that D is submitted in the form of an efficient

sampling algorithm Samp. It is required that D must additionally be *inverse-samplable*, i.e., there exists an efficient algorithm Samp^{-1} such that $(\rho, \mathsf{Samp}(\rho))$ and $(\mathsf{Samp}^{-1}(x), x)$ are indistinguishable. (This is always the case when n is logarithmic.) Furthermore, [35] requires that there exists a simulator S such that the pair $(\mathsf{View}_{\mathcal{A}}, \mathcal{T})$ consisting of the view of the adversary and the table $\mathcal{T}$ is indistinguishable from the pair $(\mathsf{View}_{\mathcal{A}}, S(\mathsf{View}_{\mathcal{A}}))$. This is used to reconstruct some possible random coins which are used in the privacy game so that we can feed the plaintext extractor of the PA game (see [35]).

PKC protocol. The PKC RFID system is pretty simple. First, $\mathsf{SetupReader}$ sets up a key pair $(\mathsf{sk}, \mathsf{pk})$ using Gen for a public-key cryptosystem $(\mathsf{Gen}, \mathsf{Enc}_{\mathsf{pk}}, \mathsf{Dec}_{\mathsf{sk}})$. We have $K_S = \mathsf{sk}$ and $K_P = \mathsf{pk}$. Then, $\mathsf{SetupTag}$ picks a random K_{ID} and sets up the state $S = (\mathsf{pk}, \mathsf{ID}, K_{\mathsf{ID}})$ and $\mathsf{data} = K_{\mathsf{ID}}$ to be inserted in the database. Then, in the identification protocol, the reader selects a nonce N, sends it to the tag. The tag then encrypts his ID, his key K_{ID} and the nonce N and sends the ciphertext to the reader. The reader can then decrypt, check that the nonce is correct, and that $(\mathsf{ID}, K_{\mathsf{ID}})$ is in the database.

A variant based on a PRF avoids using a database: we add a generation of a secret K_M for a PRF by $\mathsf{SetupReader}$ (so $K_S = (\mathsf{sk}, K_M)$) and use $K_{\mathsf{ID}} = \mathsf{PRF}_{K_M}(\mathsf{ID})$.

In [39], it was proven that if the cryptosystem is IND-CCA secure then the PKC protocol is correct, secure, wide-forward private, and narrow-strong private in the V07 model. In [35], it was proven that if the cryptosystem is further PA (in the PA1+ sense [16] or the PA2 sense [5]), then the PKC protocol is wide-strong private in the OV12 model.

IND-CCA security is necessary for the security of PKC. Clearly, it is essential that the cryptosystem is IND-CCA secure: without non-malleability, we could loose security by forging the ciphertext of a legitimate tag. For instance, given a secure cryptosystem (Gen, E, D), defining a malleable yet IND-CPA secure cryptosystem $\mathsf{Enc}_{\mathsf{pk}}(\mathsf{ID} \| K_{\mathsf{ID}} \| N) = E_{\mathsf{pk}}(\mathsf{ID} \| K_{\mathsf{ID}}) \| E_{\mathsf{pk}}(N)$ would allow to take $\mathsf{Enc}_{\mathsf{pk}}(\mathsf{ID} \| K_{\mathsf{ID}})$ as a reusable credential to be use with any fresh nonce. Hence, we could impersonate a legitimate tag.

IND-CCA security is insufficient for the wide-strong privacy of PKC in the OV12 sense. In [35], it was further proven that IND-CCA security was not sufficient to achieve wide-strong privacy. To prove this, the authors essentially construct a cryptosystem which is IND-CCA secure but not PA. More concretely, if (G^0, E^0, D^0) is an IND-CCA cryptosystem and if (G^1, E^1, D^1) is a homomorphic IND-CPA cryptosystem over the message space $\{0, 1\}$ such as the Goldwasser-Micali cryptosystem [22], we define

$$\mathsf{Gen} \to ((\mathsf{sk}_0, \mathsf{sk}_1), (\mathsf{pk}_0, \mathsf{pk}_1, z)) \quad \text{for} \quad \begin{cases} G^0 \to (\mathsf{sk}_0, \mathsf{pk}_0) \\ G^1 \to (\mathsf{sk}_1, \mathsf{pk}_1) \\ \xi \in_U \{0, 1\} \\ z = E^1_{\mathsf{pk}_1}(\xi) \end{cases}$$

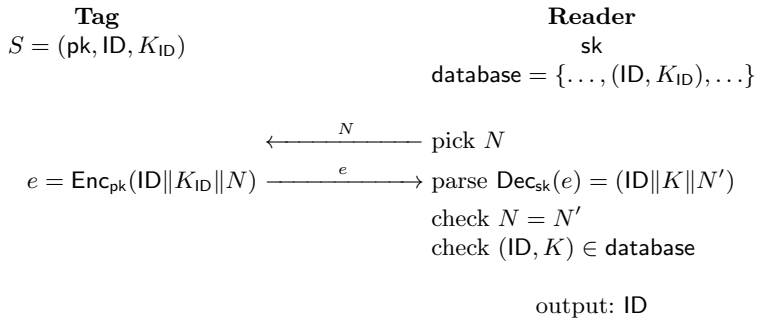

Fig. 1. The PKC Protocol based on a Cryptosystem $\mathsf{Enc}/\mathsf{Dec}$.

(note that ξ is discarded and never used again) and

$$\mathsf{Enc}_{(\mathsf{pk}_0,\mathsf{pk}_1),z}(m_1 \cdots m_n) = E^0_{\mathsf{pk}_0}(E^1_{\mathsf{pk}_1}(m_1)\| \cdots \|E^1_{\mathsf{pk}_1}(m_n))$$

where the m_i are bits. We can show that $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is an IND-CCA-secure cryptosystem. Then, we mount a wide-strong adversary who creates a legitimate tag, corrupts it, then simulate it to the reader, except that the encryption of $(\mathsf{ID}, K_{\mathsf{ID}}, N) = m_1 \cdots m_n$ is modified as follows: after computing $E^1(m_i)$ to encrypt each bit of the plaintext, he multiplies them by z. Finally,

$$e = E^0_{\mathsf{pk}_0}(z \cdot E^1_{\mathsf{pk}_1}(m_1)\| \cdots \|z \cdot E^1_{\mathsf{pk}_1}(m_n))$$

Clearly, the decryption of e by the reader is unchanged if and only if $\xi = 0$. Otherwise, all bits are flipped and lead to an incorrect nonce, so the protocol fails. As the adversary gets $\mathsf{Result}(\pi)$ from the reader, this bit is thus equal to $1 - \xi$. Although the blinder knows how the ciphertext was forged, he cannot compute ξ when (G^1, E^1, D^1) is secure. So, no blinder can simulate the $\mathsf{Result}(\pi)$ oracle and we do not have wide-strong privacy.

Public-key cryptography is necessary. We can similarly show that a wide-strong private RFID system can define a public-key cryptosystem. So, it is unlikely that we could construct one based on symmetric cryptography only. More concretely, if we create two tags ID_0 and ID_1 then corrupt both of them, their state is equivalent to a public key. Alice could send a bit b to Bob by simulating ID_b using the public key while Bob would simulate the reader with the secret key. We can show that if the scheme is wide-strong private, then we have a public cryptographic scheme in the sense of [37]. Hence, public-key cryptography is necessary.

3 The HPVP11 Model

In [23], Hermans *et al.* proposed a quite simpler privacy model (the HPVP11 model herein).

To define the HPVP11 model, we revisit the oracle calls of the adversary. All oracles work the same except CreateTag, DrawTag, and Corrupt. Namely,

$$\mathsf{CreateTag}(\mathsf{ID})$$

always create a legitimate tag.

$$\mathsf{DrawTag}(\mathsf{ID}_0, \mathsf{ID}_1) \rightarrow \mathsf{vtag}$$

draws either the tag ID_0 (in the left world) or the tag ID_1 (in the right world), and returns a fresh identifier vtag. It is not allowed to use as input an ID_b which was input of a previous $\mathsf{DrawTag} \rightarrow \mathsf{vtag}$ such that vtag was not freed. In addition to this,

$$\mathsf{Corrupt}(\mathsf{ID}) \rightarrow S$$

now works on the true identity ID of the tag instead of the one of an anonymous tag, and it is not allowed if the corresponding tag was input of a previous DrawTag and was not freed since then.[2]

The main difference is that the game first flips a coin b and uses the left world for $b = 0$ and the right world for $b = 1$. The goal of the adversary is to guess b. We have P-privacy if for any adversary in the class P, the probability to correctly guess b is lower than $\frac{1}{2}$ plus some negligible advantage.

Surprisingly, they even proved that based on IND-CCA security, the PKC RFID system is wide-strong private in their model. Hence, our proof that IND-CCA security is not sufficient shows that the PKC protocol can be wide-strong private in the HPVP11 sense but not in the OV12 sense. So, HPVP11 privacy does not imply OV12 privacy. However, looking closer at what it means in practice, we can wonder to what extend the proof that IND-CCA security is not enough for OV12 privacy implies any privacy threat. Indeed, the inability to simulate the Result oracle in our counterexample does not seem to imply any leakage in identifying information. So, HPVP11 privacy may be enough in practice.

4 Strong Privacy in Distance Bounding

In distance bounding (DB) protocols, the tag wants to prove its proximity to the reader. There are several threat models. With honest tags, we have to face to *man-in-the-middle* attacks (MiM) trying to make the reader accept a proof of proximity although no tag is actually close. MiM-security is also called HP-security (as for *Honest Prover*) in [41,43]. With malicious tags, we consider *distance fraud* (DF), where no tag is close to the reader, *distance hijacking* (DH), where a honest tag is close to the reader but the malicious one far away tries to pass the protocol, and *collusion fraud* (CF), where the malicious tag can be

[2] Some variants allow this but do not disclose states depending on possible ongoing sessions (typically: the volatile memory). So, extra care must be taken with stateful protocols.

helped by a close-by malicious adversary. CF-resistance is formalized in [41,43] in terms of *soundness* of the proximity proof: essentially, we show that if the protocol succeeds, then we can extract the secret of the identified tag from the view of participants which are close to the reader. So, there is no better CF than the trivial one consisting of giving the secret of the tag to the close-by adversary. In symmetric DB, the tag and the reader are assumed to share a secret. In public-key DB, the tag holds a key pair but shares no secret with the reader. In this paper, we concentrate on strong privacy. Since this requires public-key cryptography (as already mentioned), there is no need for limiting ourselves to symmetric DB. So, we only consider public-key DB.

Privacy in public-key DB. The first public-key DB protocol to offer privacy is the HPO protocol [24]. However, it does not offer strong privacy [44]. In [13], it was suggested to transform a symmetric DB protocol into a public-key DB protocol using a key agreement protocol. We can wonder how privacy can be preserved. The first concrete example is the privDB protocol [42]. It is depicted on Fig. 2 (taken from [42]). There, symDB denotes a one-time symmetric DB protocol (such as OTDB on Fig 3). We use a signature scheme Sign/Verify and a cryptosystem Enc/Dec. The function Validate(pk) is used to "validate" a public key, i.e. either to check that it belongs to a database, or to check a certificate (which could be pk itself).

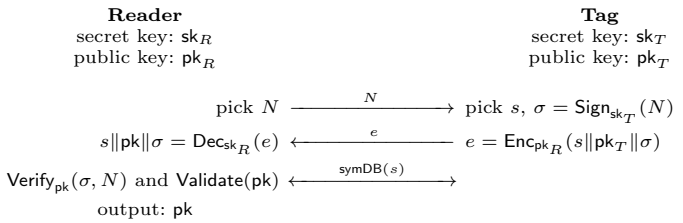

Fig. 2. privDB: Private Public-Key DB [42].

In public-key DB, the tag has a key pair $(\mathsf{sk}, \mathsf{pk})$ and the public key K_P of the system. We modify the PKC protocol as follows: instead of encrypting $\mathsf{ID}\|K_{\mathsf{ID}}\|N$, we now encrypt $s\|\mathsf{pk}\|\mathsf{Sign}_{\mathsf{sk}}(N)$ where s is a random key. Then, the reader no longer needs the secret of the tag. The identity is obtained by pk and it is enough to authenticate the tag using the signature on N. The value s can further be used as the result of a key agreement. Hence, the tag and the reader can now use s to run a symmetric DB protocol. This is the principle of the privDB protocol [42] which is wide-strong private (in the HPVP11 sense) and secure DB[3].

[3] More precisely, it defeats distance fraud, man-in-the-middle attacks, and distance hijacking, but not collusion fraud, as shown on Table 1.

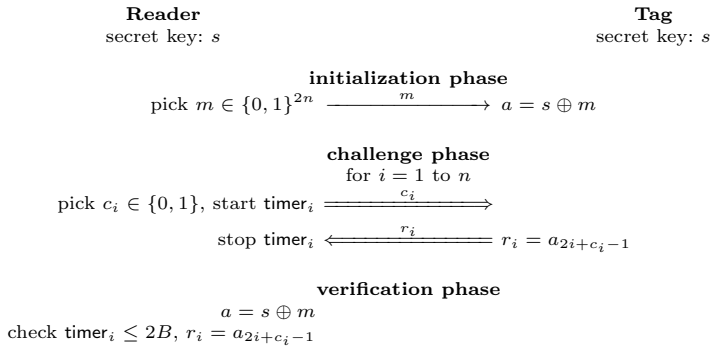

Fig. 3. OTDB: One-Time Symmetric DB [42].

Strengthening ProProx. We recall on Fig. 4 [43] a (simplified) version of ProProx. There, we use a homomorphic bit commitment scheme Com such that

$$\mathsf{Com}(b;\rho) = \theta^b \rho^2$$

in a group such that $\theta^2 = 1$ and θ has no square root, and a deterministic vector commitment scheme

$$\mathsf{Com}_H(\mathsf{sk}) = (\mathsf{Com}(\mathsf{sk}_1; H(\mathsf{sk}, 1)), \ldots, \mathsf{Com}(\mathsf{sk}_s; H(\mathsf{sk}, s)))$$

There is no required assumption on the hash function H except that Com_H must be one-way. This is the case when H is a random oracle (and sk is not too small), but H does not necessarily need to be a random oracle in this construction. We also use a zero-knowledge proof $\mathsf{ZKP}_\kappa(z_{i,j}; \zeta_{i,j})$ that there exists some $\zeta_{i,j}$ such that $z_{i,j} = \zeta_{i,j}^2$ for all i, j. We can use parallel instances of the protocol from Fig. 5 [43] with enough challenges to that the soundness probability is κ. There, we use a trapdoor commitment scheme $(\mathsf{Gen}, \mathsf{Commit}, \mathsf{Equiv})$.

As shown on Table 1, ProProx is the only public-key DB protocol with full security. However, it does not protect privacy. We can extend ProProx into a protocol eProProx as shown on Fig. 6 to add privacy protection. If $\mathsf{ProProx}_H(\mathsf{pk})$ denotes the protocol from Fig. 4, we just change the function H (as used by the tag) into a function H' and the public key pk (as used by the reader) into pk'. Essentially, we blind the public key of the tag so that it does not leak from ZKP. Interestingly, the encryption step in this extension is similar to the PKC protocol from Fig. 1. We can indeed use this encrypted channel to identify by transmitting pk: now, pk is given as an output of the reader instead as an input.

We state the security results for eProProx as they are stated for ProProx in [43]. We however simplified it (in more details, we took $p_{\mathsf{noise}} = 0$ and $\tau = n$).

Theorem 1. *The* eProProx *protocol is a sound, MiM-secure, DF-resistant, and DH-resistant proof of proximity under the assumption that*

- Com *is a perfectly binding, computationally hiding, and homomorphic bit commitment;*

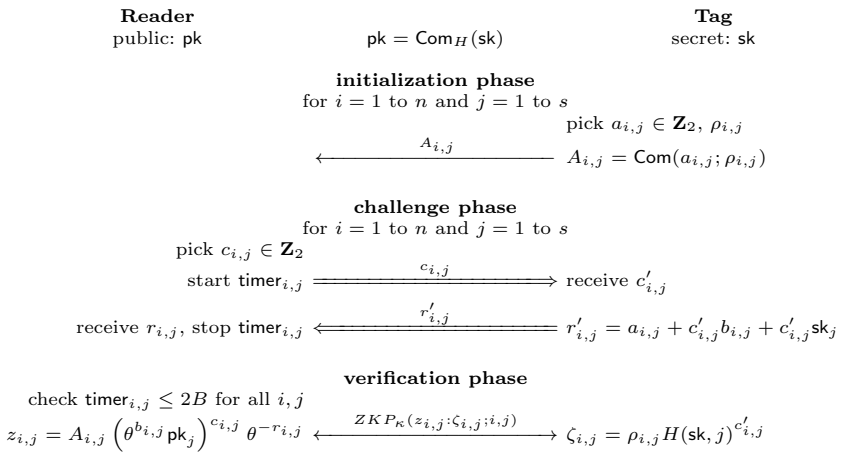

Fig. 4. ProProx: Sound Public-Key DB [43].

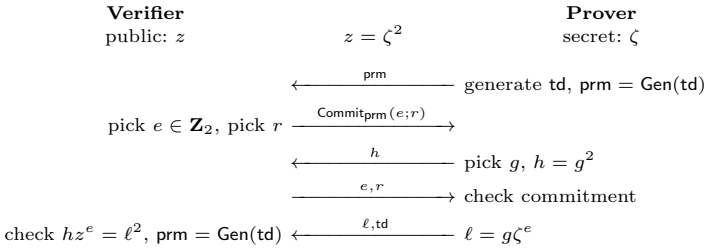

Fig. 5. ZKP($z : \zeta$): a Zero-Knowledge Proof of Existence of ζ such that $z = \zeta^2$ [43].

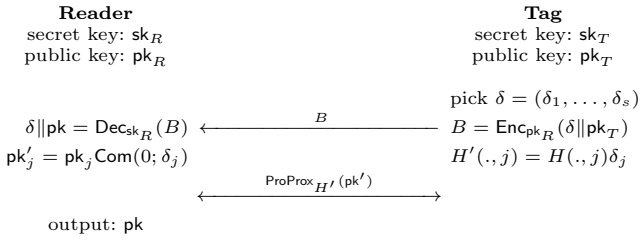

Fig. 6. eProProx: a Privacy Extension for ProProx.

– Com_H *is one-way;*
– ZKP_κ *is a complete and* κ*-sound computationally zero-knowledge proof of membership for* $\kappa = \mathsf{negl}$.

Proof. Since the cryptosystem plays no role in the security, we assume without loss of generality that $\delta \| \mathsf{pk}_T$ is sent in clear in B. We let Γ_0 be a security game. We make a new game Γ_1 which first picks one tag pk at random and succeeds in this is the tag which is identified in the attack. So, the target pk is given first. If this game succeeds with negligible probability, then Γ_0 succeeds with negligible probability as well. So, we can concentrate on Γ_1.

Reader sessions who do not receive pk in B can just be simulated by the adversary without affecting the success probability. So, we obtain a new game Γ_2 in which all readers are dedicated to pk.

Since the adversary knows δ, he knows the multiplicative factors to change pk_j into pk'_j in each $z_{i,j}$ and $H(\mathsf{sk}, j)$ into $H'(\mathsf{sk}, j)$ in each $\zeta_{i,j}$. So, we can construct an adversary against $\mathsf{ProProx}_H(\mathsf{pk})$. Then, we apply the security results from [41,43]. □

Theorem 2. *The* eProProx *protocol is wide-strong private in the HPVP11 sense under the assumption that*

– Enc/Dec *is an IND-CCA-secure cryptosystem;*
– Com *is a computationally hiding homomorphic bit commitment;*
– ZKP_κ *is a computationally zero-knowledge proof of membership.*

Proof. We consider the HPVP11 game Γ_0. Without loss of generality, we assume that drawn tags run a single session of the protocol (indeed, they can be freed and drawn again to run more sessions).

We first reduce to a game in which no different tag sessions pick the same δ vector. So, they never produce the same B, due to the correctness of the cryptosystem. We obtain a game Γ_1 producing the same output as Γ_0, except with negligible probability.

We observe that in the privacy game, the output pk on the reader side plays no role and that the reader only needs pk' to run $\mathsf{ProProx}_{H'}(\mathsf{pk}')$. Namely, δ and H' are of no use to the reader. So, we can change the protocol by having the reader saying to the adversary whether pk is a valid key, returning pk', and stopping. Then, the reader messages as in Γ_1 can be fully simulated by the adversary. We obtain a game Γ_2 in which the reader is only decrypting B, checking pk, computing pk', and releasing it.

Next, we change the game by making sure that if a session π receives B which was previously issued by one vtag after encrypting some $\delta \| \mathsf{pk}_T$, then π does not use the decryption algorithm but rather continues directly with δ and $\mathsf{pk} = \mathsf{pk}_T$. Clearly, the simulation is perfect. We let Γ_3 denote the new game.

Then, by using hybrids, we replace in Γ_3 every B issued by vtag by the encryption of some random junk string. Thanks to the IND-CCA security, we obtain a new game Γ_4 which produces the same output, except with negligible probability. So, in Γ_4, it is as if we had a protocol in which vtag has some perfectly

secure channel to transmit δ and pk_T to π but the adversary can plug or unplug this channel.

Then, we change again the protocol by having the tag to release pk'. It is already known that pk_T is valid. So, the adversary need not ask for it to the reader. Hence, we can now suppress the private channel to the reader, then even give sk_R to the adversary and completely get rid of the reader. We obtain a privacy game Γ_5 against a protocol in which there are only tags who first pick δ, compute pk', release pk', and run $\mathsf{ProProx}_{H'}(\mathsf{pk}')$.

We now construct hybrid games of Γ_5. The ith hybrid is using the right world for the first $i-1$ $\mathsf{DrawTag}$ queries and the left world for all queries starting from the $(i+1)$th one. Let vtag be the ith drawn tag. Note that vtag runs a single session of the protocol. By giving the secret of all tags to the adversary, we can get rid of all tags except vtag. We can then apply the zero-knowledge property of $\mathsf{ProProx}_{H'}(\mathsf{pk}')$ [41,43] to simulate the view of the adversary and produce the same output in the left or right world of the hybrid, except with negligible probability, by only getting pk' from vtag. Clearly, pk' is just a random commitment of sk. So, we can use the hiding property of Com to deduce that the left and right worlds of the hybrid produce the same output except with negligible probability.

So, the left and the right worlds of Γ_5 are producing the same output except with negligible probability. □

5 Conclusion

As we have seen, we now have pretty mature privacy models for RFID and protocols reaching their stronger flavors. So far, these models fully cover *unilateral authenticated identification* protocols, in which a tag identifies to a reader. These models could be enriched to cover other protocols: we could consider DB protocols in which the tag is already identified, we could consider *bilateral identification* protocols, we could consider protocols with *mutual authentication* (such as in [1,36]). In general, even if we do have a general model for elementary protocol, it is not clear that we could compose private elementary protocols for free. More research must be done for *composable privacy*.

For completeness, we mention that we only discussed privacy related to identifying information. Some other forms of privacy are discussed in the literature, such as the *location privacy* [27,28,29].

Acknowledgements. This work was partly sponsored by the ICT COST Action IC1403 Cryptacus in the EU Framework Horizon 2020.

References

1. F. Armknecht, A.-R. Sadeghi, I. Visconti, C. Wachsmann. On RFID Privacy with Mutual Authentication and Tag Corruption. In *Applied Cryptography and Network Security (ACNS'10)*, Beijing, China, Lecture Notes in Computer Science 6123, pp. 493–510, Springer-Verlag, 2010.

2. G. Avoine. Cryptography in Radio Frequency Identification and Fair Exchange Protocols. PhD Thesis no. 3407, EPFL, 2005. http://library.epfl.ch/theses/?nr=3407
3. G. Avoine, E. Dysli, P. Oechslin. Reducing Time Complexity in RFID Systems. In *Selected Areas in Cryptography'05*, Kingston, Ontario, Canada, Lecture Notes in Computer Science 3897, pp. 291–306, Springer-Verlag, 2006.
4. A. Bay, I. Boureanu, A. Mitrokotsa, I. Spulber, S. Vaudenay. The Bussard-Bagga and Other Distance-Bounding Protocols under Attacks. In *Information Security and Cryptology INSCRYPT'12*, Beijing, China, Lecture Notes in Computer Science 7763, pp. 371–391, Springer-Verlag, 2012.
5. M. Bellare, A. Palacio. Towards Plaintext-Aware Public-Key Encryption without Random Oracles. In *Advances in Cryptology ASIACRYPT'04*, Jeju Island, Korea, Lecture Notes in Computer Science 3329, pp. 48–62, Springer-Verlag, 2004.
6. S. Bocchetti. Security and Privacy in RFID Protocols. Master Thesis, 2006.
7. I. Boureanu, S. Vaudenay. Optimal Proximity Proofs. In *Information Security and Cryptology Inscrypt'14*, Beijing, China, Lecture Notes in Computer Science 8957, pp. 170–190, Springer-Verlag, 2014. Eprint 2014/693. http://eprint.iacr.org/2014/693.pdf
8. I. Boureanu, A. Mitrokotsa, S. Vaudenay. Towards Secure Distance Bounding. In *Fast Software Encryption'13*, Singapore, Lecture Notes in Computer Science 8424, pp. 55–67, Springer-Verlag, 2013. Eprint 2015/208. http://eprint.iacr.org/2015/208.pdf
9. I. Boureanu, A. Mitrokotsa, S. Vaudenay. Secure & Lightweight Distance-Bounding. In *Lightweight Cryptography for Security and Privacy LightSec'13*, Gebze, Turkey, Lecture Notes in Computer Science 8162, pp. 97–113, Springer-Verlag, 2013.
10. I. Boureanu, A. Mitrokotsa, S. Vaudenay. Practical & Provably Secure Distance-Bounding. To appear in the proceedings of ISC'13. Also [12,11].
11. I. Boureanu, A. Mitrokotsa, S. Vaudenay. Practical & Provably Secure Distance-Bounding. IACR Eprint 2013/465 report, 2013. http://eprint.iacr.org/2013/465.pdf Also [12,10].
12. I. Boureanu, A. Mitrokotsa, S. Vaudenay. Practical and Provably Secure Distance-Bounding. *Journal of Computer Security (JCS)*, vol. 23 (2), pp. 229–257, 2015. Also [10,11]
13. I. Boureanu, S. Vaudenay. Challenges in Distance Bounding. *Security & Privacy*, vol. 13 (1), pp. 41–47, 2015.
14. S. Brands, D. Chaum. Distance-Bounding Protocols (Extended Abstract). In *Advances in Cryptology EUROCRYPT'93*, Lofthus, Norway, Lecture Notes in Computer Science 765, pp. 344–359, Springer-Verlag, 1994.
15. L. Bussard, W. Bagga. Distance-Bounding Proof of Knowledge to Avoid Real-Time Attacks. In *IFIP TC11 International Conference on Information Security SEC'05*, Chiba, Japan, pp. 223–238, Springer, 2005.
16. A. Dent. The Cramer-Shoup Encryption Scheme is Plaintext-Aware in the Standard Model. In *Advances in Cryptology EUROCRYPT'06*, St. Petersburg, Russia, Lecture Notes in Computer Science 4004, pp. 289–307, Springer-Verlag, 2006.
17. T. Dimitriou. A Lightweight RFID Protocol to Protect against Traceability and Cloning Attacks. In *Conference on Security and Privacy for Emerging Areas in Communication Networks (SecureComm'05)*, Athens, Greece, IEEE, 2005. http://ieeexplore.ieee.org/iel5/10695/33755/01607559.pdf?arnumber=1607559
18. U. Dürholz, M. Fischlin, M. Kasper, C. Onete. A Formal Approach to Distance-Bounding RFID Protocols. In *Information Security ISC'11*, Xi'an, China, Lecture Notes in Computer Science 7001, pp. 47–62, Springer-Verlag, 2011.

19. M. Feldhofer, S. Dominikus, J. Wolkerstorfer. Strong Authentication for RFID Systems using the AES Algorithm. In *Workshop on Cryptographic Hardware and Embedded Systems (CHES'04)*, Boston, MA, USA, Lecture Notes in Computer Science 3156, pp. 357–370, Springer-Verlag, 2004.
20. M. Fischlin, C. Onete. Terrorism in Distance Bounding: Modelling Terrorist-Fraud Resistance. In *Applied Cryptography and Network Security ACNS'13*, Banff AB, Canada, Lecture Notes in Computer Science 7954, pp. 414–431, Springer-Verlag, 2013.
21. S. Gambs, C. Onete, J.-M. Robert. Prover Anonymous and Deniable Distance-Bounding Authentication. In *ACM Symposium on Information, Computer and Communications Security (ASIACCS'14)*, Kyoto, Japan, pp. 501–506, ACM Press, 2014.
22. S. Goldwasser, S. Micali. Probabilistic Encryption. *Journal of Computer and System Sciences*, vol. 28(2), pp. 270–299, 1984.
23. J. Hermans, A. Pashalidis, F. Vercauteren, B. Preneel. A New RFID Privacy Model. In *Computer Security - ESORICS'11*, Leuven, Belgium, Lecture Notes in Computer Science 6879, pp. 568–587, Springer-Verlag, 2011.
24. J. Hermans, R. Peeters, C. Onete. Efficient, Secure, Private Distance Bounding without Key Updates. In *ACM Conference on Security and Privacy in Wireless and Mobile Networks WISEC'13*, Budapest, Hungary, pp. 195–206, ACM, 2013.
25. A. Juels, S. Weis. Defining Strong Privacy for RFID. Technical report 2006/137, IACR, 2006. http://eprint.iacr.org/2006/137
26. H. Kılınç, S. Vaudenay. Optimal Proximity Proofs Revisited. To appear in ACNS'15.
27. A. Mitrokotsa, C. Onete, S. Vaudenay. Mafia Fraud Attack against the RC Distance-Bounding Protocol. In *Proceedings of the RFID-TA'12*, Nice, France, pp. 74–79, IEEE, 2012.
28. A. Mitrokotsa, C. Onete, S. Vaudenay. Location Leakage in Distance Bounding: Why Location Privacy does not Work. IACR Eprint 2013/776 report, 2013. http://eprint.iacr.org/2013/776.pdf
29. A. Mitrokotsa, C. Onete, S. Vaudenay. Location Leakage in Distance Bounding: Why Location Privacy does not Work. *Computers & Security*, vol. 45, pp. 199–209, 2014.
30. D. Molnar, D. Wagner. Privacy and Security in Library RFID: Issues, Practices, and Architectures. In *11th ACM Conference on Computer and Communications Security*, Washington DC, USA, pp. 210–219, ACM Press, 2004.
31. C.Y. Ng, W. Susilo, Y. Mu, R. Safavi-Naini. RFID Privacy Models Revisited. In *Computer Security ESORICS'08*, Málaga, Spain, Lecture Notes in Computer Science 5283, pp. 251–266, Springer-Verlag, Springer. 2008
32. M. Ohkubo, K. Suzuki, S. Kinoshita. Cryptographic Approach to a Privacy Friendly Tag. Presented at the *RFID Privacy Workshop*, MIT, USA, 2003.
33. M. Ohkubo, K. Suzuki, S. Kinoshita. Efficient Hash-Chain based RFID Privacy Protection Scheme. Presented at the *International Conference on Ubiquitous Computing (Ubicomp'04), Workshop Privacy: Current Status and Future Directions*, Nottingham, UK, 2004.
34. M. Ohkubo, K. Suzuki. RFID Privacy Issues and Technical Challenges. *Communications of the ACM*, vol. 48, pp. 66–71, 2005.
35. K. Ouafi, S. Vaudenay. Strong Privacy for RFID Systems from Plaintext-Aware Encryption. In *Cryptology and Network Security, 8th International Conference CANS'12*, Darmstadt, Germany, Lecture Notes in Computer Science 7712, pp. 247–262, Springer-Verlag, 2012.

36. R.-I. Paise, S. Vaudenay. Mutual Authentication in RFID. In *ACM Symposium on Information, Computer and Communications Security (ASIACCS'08)*, Tokyo, Japan, pp. 292–299, ACM Press, 2008.
37. S. Rudich. The Use of Interaction in Public Cryptosystems. In *Advances in Cryptology CRYPTO'91*, Santa Barbara, California, U.S.A., Lecture Notes in Computer Science 576, pp. 242–251, Springer-Verlag, 1992.
38. S. Vaudenay. RFID Privacy based on Public-Key Cryptography. (Invited Talk.) In *International Conference on Information Security and Cryptography ICISC'06*, Busan, Korea, Lecture Notes in Computer Science 4296, pp. 1–6, Springer-Verlag, 2006.
39. S. Vaudenay. On Privacy Models for RFID. In *Advances in Cryptology ASIACRYPT'07*, Kuching, Malaysia, Lecture Notes in Computer Science 4833, pp. 68–87, Springer-Verlag, 2007.
40. S. Vaudenay. Strong Privacy for RFID Systems from Plaintext-Aware Encryption. Presented at the Early Symmetric Crypto seminar ESC'13, Mondorf-les-Bains, Luxembourg, 2013.
41. S. Vaudenay. Proof of Proximity of Knowledge. IACR Eprint 2014/695 report, 2014. http://eprint.iacr.org/2014/695.pdf Also [43].
42. S. Vaudenay. Private and Secure Public-Key Distance Bounding: Application to NFC Payment. In *Financial Cryptography and Data Security (FC'15)*, San Juan, Puerto Rico, Lecture Notes in Computer Science 8975, pp. 207–216, Springer-Verlag, 2015.
43. S. Vaudenay. Sound Proof of Proximity of Knowledge. In these proceedings. Also [41].
44. S. Vaudenay. Privacy Failure in Public-Key Distance-Bounding Protocols. Under submission.
45. S. Weis, S. Sarma, R. Rivest, D. Engels. Security and Privacy Aspects of Low-Cost Radio Frequency Identification Systems. In *International Conference on Security in Pervasive Computing (SPC'03)*, Boppard, Germany, Lecture Notes in Computer Science 2802, pp. 454–469, Springer-Verlag, 2003.