Impossibility Results for RFID Privacy Notions

Frederik Armknecht[1], Ahmad-Reza Sadeghi[2], Alessandra Scafuro[3],
Ivan Visconti[3], and Christian Wachsmann[2]

[1] University of Mannheim, Germany
armknecht@informatik.uni-mannheim.de
[2] Horst Görtz Institute for IT-Security (HGI), Ruhr-University Bochum, Germany
{ahmad.sadeghi,christian.wachsmann}@trust.rub.de
[3] Dipartimento di Informatica ed Applicazioni, University of Salerno, Italy
{scafuro,visconti}@dia.unisa.it

Abstract. RFID systems have become increasingly popular and are already used in many real-life applications. Although very useful, RFIDs introduce privacy risks since they carry identifying information that can be traced. Hence, several RFID privacy models have been proposed. However, they are often incomparable and in part do not reflect the capabilities of real-world adversaries. Recently, Paise and Vaudenay presented a general RFID security and privacy model that abstracts and unifies most previous approaches. This model defines mutual authentication (between RFID tags and readers) and several privacy notions that capture adversaries with different tag corruption behavior and capabilities.

In this paper, we revisit the model proposed by Paise and Vaudenay and investigate some subtle issues such as tag corruption aspects. We show that in their formal definitions tag corruption discloses the temporary memory of tags and leads to the impossibility of achieving both mutual authentication and any reasonable notion of RFID privacy in their model. Moreover, we show that the strongest privacy notion (narrow-strong privacy) cannot be achieved simultaneously with reader authentication even under the strong assumption that tag corruption does not disclose temporary tag states. Further, we show other impossibility results that hold if the adversary can manipulate an RFID tag such that it resets its state or when tags are stateless.

Although our results are shown on the privacy definition by Paise and Vaudenay, they give insight to the difficulties of setting up a mature security and privacy model for RFID systems that aims at fulfilling the sophisticated requirements of real-life applications.

Keywords: RFID, Privacy, Authentication, Security, Resettability.

1 Introduction

Radio Frequency Identification (RFID) enables RFID *readers* to perform fully automatic wireless identification of objects that are labeled with RFID *tags*, and is widely deployed to many applications (e.g., access control [2,29], electronic tickets [31,29], and e-passports [19]). As pointed out in previous publications

M.L. Gavrilova et al. (Eds.): Trans. on Comput. Sci. XI, LNCS 6480, pp. 39–63, 2010.
© Springer-Verlag Berlin Heidelberg 2010

(see, e.g., [38,20,34]), this prevalence of RFID technology introduces various risks, in particular concerning the privacy of its users and holders. The most deterrent privacy risk concerns the tracking of users, which allows the creation and misuse of detailed user profiles. Thus, it is desired that an RFID system provides *anonymity* (confidentiality of the tag identity) as well as *untraceability* (unlinkability of the communication of a tag), even in case the state (e.g., the secret) of a tag has been disclosed.

The design of a secure privacy-preserving RFID scheme requires a careful analysis in an appropriate formal model. There is a large body of literature on security and privacy models for RFID (see, e.g., [3,21,8,37,30,12]). Existing solutions often do not consider important aspects like adversaries with access to auxiliary information, e.g., on whether the identification of a tag was successful, or the privacy of corrupted tags whose state has been disclosed. In particular, tag corruption is usually considered to happen only *before* and/or *after* but *not during* a protocol-run. However, in practice there are a variety of side-channel attacks (see., e.g., [24,18,22]) that extract the state of a tag based on the observation of, e.g., the power consumption of the tag *while* it is executing a protocol with the reader. Since RFID tags are usually cost-effective devices without expensive tamper-proof mechanisms [2,29], tag corruption is an important aspect to be covered by the underlying (formal) security model. Though in literature, tag corruption during protocol execution is rarely considered. To the best of our knowledge, the security and privacy model in [8] is the only one that considers corruption of tags during protocol executions and proposes a protocol in this model. However, this model does not consider issues like the privacy of tags after they have been corrupted and privacy against adversaries with access to auxiliary information. Moreover, [8] only provides an informal security analysis of the proposed protocol. Recently, tag corruption during protocol-runs has been informally discussed in [12]. However, the formal RFID security and privacy model proposed in [12] assumes that such attacks cannot occur. Moreover, [12] indicates informally without giving formal proofs that tag corruption during protocol execution may have an impact on the formal definitions of [37,30], which are basis for many subsequent works (see, e.g., [26,25,7,33,32,11,10,36,35]). The first papers addressing tag corruption during protocol-runs in the model of [37] are [11,10], where it is shown that privacy can be achieved under the assumption that tag corruption during protocol execution can be detected by the tag.

In this paper, we focus on the security and privacy model by Paise and Vaudenay [30] (that is based on [37]), which we call the *PV-Model* (Paise-Vaudenay Model) in the following. The PV-Model is one of the most comprehensive RFID security and privacy models up to date since it captures many aspects of real world RFID systems and aims at abstracting most previous works in a single concise framework. It defines mutual authentication between RFID tags and readers and several privacy notions that correspond to adversaries with different tag corruption abilities. However, as we show in this paper, the PV-Model suffers from subtle deficiencies and weaknesses that are mainly caused by tag corruption aspects: in the PV-Model, each tag maintains a state that can be divided into a persistent and a temporary part.[1] The *persistent state* subsumes all information that must be available to the tag in more than one interaction with the reader (e.g., the authentication secret of the tag) and can be updated during the interaction with the reader. The *temporary state* consists of all ephemeral information that is discarded by the tag after each interaction with the reader (e.g., the randomness used by the tag). As discussed in [30], in the PV-Model it is impossible to achieve any notion of privacy that allows tag corruption if the adversary can obtain *both* the persistent *and* the temporary tag state by tag corruption. This issue is addressed by the PV-Model by the assumption that each tag erases its temporary state each time it gets out of the reading range of the adversary. However, this assumption leaves open the possibility to corrupt a tag *while* it is in the reading range of the adversary, i.e., *before* its temporary state is erased. In particular, the PV-Model allows the adversary to corrupt a tag *while* it is executing the authentication protocol with the reader.

Moreover, an adversary in practice could physically tamper with a tag such that the tag resets its state and randomness to a previous value. This form of physical attack is not considered in the PV-Model and thus, the study of privacy notions done in [30] does not address these attacks.

Contribution. In this paper, we point out subtle weaknesses and deficiencies in the PV-Model. First, we show that the assumption of erasing temporary tag states whenever a tag gets out of the reading range of the adversary made by the PV-Model is not strong enough. We prove that, even under this assumption, it is *impossible* to achieve reader authentication and simultaneously *any* notion of privacy that allows tag corruption. This implies that the PV-Model cannot provide privacy along with mutual authentication without relying on tamper-proof hardware, which is unrealistic for low-cost RFID tags. Consequently, two of the three schemes presented in [30] do not satisfy their claimed properties.

Our second contribution is to show that even under the strong assumption that the temporary tag state is not subject to tag corruption attacks, some privacy notions still remain impossible in the PV-Model. This implies that the third protocol of [30] has another conceptually different weakness.

Finally, we show that by extending the model of [30] to capture reset attacks on tag states and randomness, no privacy can be achieved, and, more interestingly, when tags are stateless (i.e., when tags cannot update their persistent state), then destructive privacy is impossible. Although our results are shown on the privacy model by Paise and Vaudenay, we believe that our work is helpful for developing a mature security and privacy model for RFID systems that fulfills the sophisticated requirements of real-life applications.

Outline. We first informally discuss the general RFID scenario on a high level in Section 2. Then we focus on the formalization of the relevant aspects by revisiting the RFID security and privacy model by Paise and Vaudenay (PV-Model) [30]

[1] During a protocol execution tags could store some temporary information that allows them to verify the response of the reader.

in Section 3. In Section 4 we present our first result while our second result is shown in Section 5. In Section 6 we show our third impossibility result based on resettable and stateless tags. Finally, we conclude in Section 7.

2 RFID System and Requirement Analysis

System model. An RFID system consists of at least an operator $\mathcal{I}$, a reader $\mathcal{R}$ and a tag $\mathcal{T}$. $\mathcal{I}$ is the entity that enrolls and maintains the RFID system. Hence, $\mathcal{I}$ initializes $\mathcal{T}$ and $\mathcal{R}$ before they are deployed in the system. $\mathcal{T}$ and $\mathcal{R}$ are called *legitimate* if they have been initialized by $\mathcal{I}$. In many applications $\mathcal{T}$ is a hardware token with constrained computing and memory capabilities that is equipped with a radio interface [2,29]. All information, e.g., secrets and data that is stored on $\mathcal{T}$ is denoted as the *state* of $\mathcal{T}$. Usually $\mathcal{T}$ is attached to some object or carried by a user of the RFID system [14,28]. $\mathcal{R}$ is a stationary or mobile computing device that interacts with $\mathcal{T}$ when $\mathcal{T}$ gets into the reading range of $\mathcal{R}$. The main purpose of this interaction usually is the authentication of $\mathcal{T}$ to $\mathcal{R}$. Depending on the use case, $\mathcal{R}$ may also authenticate to $\mathcal{T}$ and/or obtain additional information like the identity of $\mathcal{T}$. $\mathcal{R}$ can have a sporadic or permanent online connection to some backend system $\mathcal{D}$, which typically is a database maintaining detailed information on all tags in the system. $\mathcal{D}$ is initialized and maintained by $\mathcal{I}$ and can be read and updated by $\mathcal{R}$.

Trust and adversary model. The operator $\mathcal{I}$ maintains the RFID system and is considered to behave correctly. However, $\mathcal{I}$ may be curious and collect user information. Since $\mathcal{T}$ and $\mathcal{R}$ communicate over a radio link, any entity can eavesdrop and manipulate this communication, even from outside the nominal reading range of $\mathcal{R}$ and $\mathcal{T}$ [23]. Thus, the adversary $\mathcal{A}$ can be every (potentially unknown) entity. Besides the communication between $\mathcal{T}$ and $\mathcal{R}$, $\mathcal{A}$ can also obtain useful auxiliary information (e.g., by visual observation) on whether $\mathcal{R}$ accepted $\mathcal{T}$ as a legitimate tag [21,37]. Most commercial RFID tags are cost-efficient devices without expensive protection mechanisms against physical tampering [2,29]. Hence, $\mathcal{A}$ can physically attack (*corrupt*) $\mathcal{T}$ and obtain its state, e.g., its secrets. In practice, RFID readers are embedded devices that can be integrated into mobile devices (e.g., mobile phones or PDAs) or computers. The resulting complexity exposes readers to sophisticated hard- and software attacks, e.g., viruses and Trojans. This problem aggravates for mobile readers that can easily be lost or stolen. Hence, $\mathcal{A}$ can get full control over $\mathcal{R}$ [4,16,27].

Security and privacy objectives. The most deterrent privacy risk concerns the *tracking* of tag users, which allows the creation and misuse of detailed user profiles in an RFID system [20]. For instance, detailed movement profiles can leak sensitive information on the personal habits and interests of the tag user. The major security threats are to create illegitimate (*forge*) tags that are accepted by honest readers, to simulate (*impersonate*) or to copy (*clone*) legitimate tags, and to permanently prevent users from using the RFID system (*denial-of-service*) [8].

Thus, an RFID system should provide *anonymity* as well as *untraceability* of a tag $\mathcal{T}$ even when the state of $\mathcal{T}$ has been disclosed. Anonymity means the confidentiality of the identity of $\mathcal{T}$ whereas untraceability refers to the unlinkability of the communication of $\mathcal{T}$. The main security objective is to ensure that only legitimate tags are accepted by honest readers (*tag authentication*). Most use cases (like access control systems) additionally require $\mathcal{R}$ to determine the authentic tag identity (*tag identification*). Moreover, there are several applications (e.g., electronic tickets) where reader authentication is a fundamental security property. However, there are also use cases (e.g., electronic product labels) that do not require reader authentication.

3 The PV-Model

In this section, we recall the RFID security and privacy model by Paise and Vaudenay (PV-Model) [30] that refines the model in [37]. We give a more formal specification of this model, which is one of the most comprehensive RFID privacy and security models up to date. We start by specifying our notation.

General notation. For a finite set S, $|S|$ denotes the size of S whereas for an integer (or a bit-string) n the term $|n|$ means the bit-length of n. The term $s \in_R S$ means the assignment of a uniformly chosen element of S to variable s. Let A be a probabilistic algorithm. Then $y \leftarrow \mathsf{A}(x)$ means that on input x, algorithm A assigns its output to variable y. The term $[\mathsf{A}(x)]$ denotes the set of all possible outputs of A on input x. $\mathsf{A}_K(x)$ means that the output of A depends on x and some additional parameter K (e.g., a secret key). The term $\mathsf{Prot}[\mathsf{A}{:}x_\mathsf{A};\ \mathsf{B}{:}x_\mathsf{B};\ *{:}x_{pub}] \rightarrow [\mathsf{A}{:}y_\mathsf{A};\ \mathsf{B}{:}y_\mathsf{B}]$ denotes an interactive protocol Prot between two probabilistic algorithms A and B. Hereby, A (resp. B) gets a private input x_A (resp. x_B) and a public input x_{pub}. While A (resp. B) is operating, it can interact with B (resp. A). After the protocol terminates, A (resp. B) returns y_A (resp. y_B). Let E be some event (e.g., the result of a security experiment), then $\Pr[E]$ denotes the probability that E occurs. Probability $\epsilon(l)$ is called *negligible* if for all polynomials f it holds that $\epsilon(l) \leq 1/f(l)$ for all sufficiently large l. Probability $1 - \epsilon(l)$ is called *overwhelming* if $\epsilon(l)$ is negligible.

3.1 System Model

The PV-Model considers RFID systems that consist of a single operator $\mathcal{I}$, a single reader $\mathcal{R}$ and a polynomial number of tags $\mathcal{T}$. Note that the PV-Model does not explicitly define an entity that corresponds to the operator $\mathcal{I}$ but implies the existence of such an entity. $\mathcal{R}$ is assumed to be capable of performing public-key cryptography and of handling multiple instances of the mutual authentication protocol with different tags in parallel. Each tag $\mathcal{T}$ is a passive device, i.e., it does not have its own power supply but is powered by the electromagnetic field of $\mathcal{R}$. Hence, $\mathcal{T}$ cannot initiate communication, has a narrow communication range (i.e., a few centimeters to meters) and erases its temporary state (i.e., all session-specific information and randomness) after it gets out of the reading range of $\mathcal{R}$. Each $\mathcal{T}$ is assumed to be capable of computing basic cryptographic functions like hashing, random number generation and symmetric-key encryption. The authors of [37,30] also use public-key encryption, although it exceeds the capabilities of most currently available RFID tags [2,29].

Security and privacy objectives. The main security objective of the PV-Model is mutual authentication. More precisely, $\mathcal{R}$ should only accept legitimate tags and must be able to identify them, while each legitimate tag $\mathcal{T}$ should only accept $\mathcal{R}$. Availability and protection against cloning are not captured by the PV-Model. The privacy objectives are anonymity and unlinkability.

Definitions. The operator $\mathcal{I}$ sets up $\mathcal{R}$ and all tags $\mathcal{T}$. Hence, there are two setup algorithms where $\mathcal{R}$ and $\mathcal{T}$ are initialized and their system parameters (e.g., keys) are generated and defined. A protocol between $\mathcal{T}$ and $\mathcal{R}$ covers mutual authentication.

Definition 1 (RFID System [30]). *An RFID system is a tuple of probabilistic polynomial time (p.p.t.) algorithms* $(\mathcal{R}, \mathcal{T}, \mathsf{SetupReader}, \mathsf{SetupTag}, \mathsf{Ident})$ *that are defined as follows:*

$\mathsf{SetupReader}(1^l) \to (sk_{\mathcal{R}}, pk_{\mathcal{R}}, \mathtt{DB})$ *On input of a security parameter l, this algorithm creates the public parameters $pk_{\mathcal{R}}$ that are known to all entities. Moreover, it creates the secret parameters $sk_{\mathcal{R}}$ and a database* $\mathtt{DB}$ *that can only be accessed by $\mathcal{R}$.*

$\mathsf{SetupTag}_{pk_{\mathcal{R}}}(\mathtt{ID}) \to (K, S)$ *uses $pk_{\mathcal{R}}$ to generate a tag secret K and tag state S, initializes $\mathcal{T}_{\mathtt{ID}}$ with S, and stores* $(\mathtt{ID}, K)$ *in* $\mathtt{DB}$.

$\mathsf{Ident}[\mathcal{T}_{\mathtt{ID}} : S;\ \mathcal{R} : sk_{\mathcal{R}}, \mathtt{DB};\ * : pk_{\mathcal{R}}] \to [\mathcal{T}_{\mathtt{ID}} : out_{\mathcal{T}_{\mathtt{ID}}};\ \mathcal{R} : out_{\mathcal{R}}]$ *is an interactive protocol between $\mathcal{T}_{\mathtt{ID}}$ and $\mathcal{R}$. $\mathcal{T}_{\mathtt{ID}}$ takes as input its current state S while $\mathcal{R}$ has input $sk_{\mathcal{R}}$ and* $\mathtt{DB}$. *The common input to all parties is $pk_{\mathcal{R}}$. After the protocol terminates, $\mathcal{R}$ returns either the identity* $\mathtt{ID}$ *of $\mathcal{T}_{\mathtt{ID}}$ or $\perp$ to indicate that $\mathcal{T}_{\mathtt{ID}}$ is not a legitimate tag. $\mathcal{T}_{\mathtt{ID}}$ returns either* $\mathtt{ok}$ *to indicate that $\mathcal{R}$ is legitimate or $\perp$ otherwise.*

Definition 2 (Correctness [30]). *An RFID system (Definition 1) is correct if* $\forall\ l$, $\forall\ (sk_{\mathcal{R}}, pk_{\mathcal{R}}, \mathtt{DB}) \in [\mathsf{SetupReader}(1^l)]$, *and* $\forall\ (K, S) \in [\mathsf{SetupTag}_{pk_{\mathcal{R}}}(\mathtt{ID})]$ $\mathsf{Ident}[\mathcal{T}_{\mathtt{ID}} : S;\ \mathcal{R} : sk_{\mathcal{R}}, \mathtt{DB};\ * : pk_{\mathcal{R}}] \to [\mathcal{T}_{\mathtt{ID}} : \mathtt{ok};\ \mathcal{R} : \mathtt{ID}]$ *holds with overwhelming probability.*

3.2 Trust and Adversary Model

In the PV-Model, the issuer $\mathcal{I}$, the backend database $\mathcal{D}$ and the readers are assumed to be trusted whereas a tag $\mathcal{T}$ can be compromised. All readers and $\mathcal{D}$ are subsumed to *one single* reader entity $\mathcal{R}$ that cannot be corrupted. This implies that all readers are assumed to be tamper-resistant devices that have a permanent online connection to $\mathcal{D}$.[2] The PV-Model defines privacy and security

[2] Depending on the use case, this assumption can be problematic in practice, e.g., for mobile readers that usually have only a sporadic or no online connection and that are subject to a variety of soft- and hardware attacks.

as security experiments, where a p.p.t. adversary $\mathcal{A}$ can interact with a set of oracles that model the capabilities of $\mathcal{A}$. These oracles are:

$\mathsf{CreateTag}^b(\mathtt{ID})$ Allows $\mathcal{A}$ to set up a tag $\mathcal{T}_{\mathtt{ID}}$ with identifier $\mathtt{ID}$ by internally calling $\mathsf{SetupTag}_{pk_{\mathcal{R}}}(\mathtt{ID})$ to create (K, S) for $\mathcal{T}_{\mathtt{ID}}$. If input $b = 1$, then $(\mathtt{ID}, K)$ is added to $\mathtt{DB}$. If $b = 0$, then $(\mathtt{ID}, K)$ is *not* added to $\mathtt{DB}$.

$\mathsf{Draw}(\delta) \rightarrow (vtag_1, b_1, \ldots, vtag_n, b_n)$ Initially, $\mathcal{A}$ cannot interact with any tag but must query Draw to get access to a set of tags chosen according to a probability distribution δ. $\mathcal{A}$ knows the tags it can interact with by some temporary tag identifiers $vtag_1, \ldots, vtag_n$. Draw manages a secret look-up table Γ that keeps track of the real tag identifier $\mathtt{ID}_i$ associated with each temporary tag identifier $vtag_i$, i.e., $\Gamma[vtag_i] = \mathtt{ID}_i$. Moreover, Draw also provides $\mathcal{A}$ with information on whether the tags are legitimate $(b_i = 1)$ or not $(b_i = 0)$.

$\mathsf{Free}(vtag)$ Makes tag $vtag$ inaccessible to $\mathcal{A}$ such that $\mathcal{A}$ cannot interact with $vtag$ until it is made accessible again under a new temporary identifier $vtag'$ by another Draw query.

$\mathsf{Launch}() \rightarrow \pi$ Makes $\mathcal{R}$ to start a new instance π of the Ident protocol.

$\mathsf{SendReader}(m, \pi) \rightarrow m'$ Sends a message m to instance π of the Ident protocol that is running on $\mathcal{R}$. $\mathcal{R}$ interprets m as a protocol message of instance π of the Ident protocol and responds with a message m'.

$\mathsf{SendTag}(m, vtag) \rightarrow m'$ Sends a message m to the tag $vtag$, which interprets m as a protocol message of the Ident protocol and responds with a message m'.

$\mathsf{Result}(\pi)$ Returns 1 if instance π of the Ident protocol has been completed and the tag $\mathcal{T}_{\mathtt{ID}}$ that participated in instance π has been accepted by $\mathcal{R}$. Otherwise Result returns 0.

$\mathsf{Corrupt}(vtag) \rightarrow S$ Returns the current state S (i.e., all information stored in the memory) of the tag $vtag$ to $\mathcal{A}$.

The PV-Model distinguishes eight adversary classes, which differ in (i) their ability to corrupt tags and (ii) the availability of auxiliary information, i.e., the ability to access the $\mathsf{Corrupt}$ and Result oracle, respectively.

Definition 3 (Adversary Classes [30]). *An adversary is a p.p.t. algorithm that has arbitrary access to all oracles described in Section 3.2. Weak adversaries cannot access the* $\mathsf{Corrupt}$ *oracle. Forward adversaries cannot query any other oracle than* $\mathsf{Corrupt}$ *after they made the first* $\mathsf{Corrupt}$ *query. Destructive adversaries cannot query any oracle for vtag again after they made a* $\mathsf{Corrupt}(vtag)$ *query. Strong adversaries have no restrictions on the use of the* $\mathsf{Corrupt}$ *oracle. Narrow adversaries cannot access the* Result *oracle.*

Tag corruption aspects. Depending on the concrete scenario, the temporary tag state is disclosed under tag corruption. In general, any concrete scenario will range between the following two extremes: (i) corruption discloses the full temporary tag state, or (ii) corruption does not disclose any information on the temporary tag state. In Section 4 and 5, we will prove that in both cases some privacy notions are impossible to achieve in the PV-Model. Thus, *independently* of any possible interpretation of tag corruption, impossibility results exist that contradict the claims of [30].

3.3 Security Definition

The security definition of the PV-Model focuses on attacks where the adversary aims to impersonate or forge a legitimate tag $\mathcal{T}$ or the reader $\mathcal{R}$. It does *not* capture availability and security against cloning.

Tag authentication. The definition of tag authentication is based on a security experiment $\mathbf{Exp}_{\mathcal{A}_{\text{sec}}}^{\mathcal{T}\text{-aut}}$ where a strong adversary $\mathcal{A}_{\text{sec}}$ (Definition 3) must make $\mathcal{R}$ to identify some tag $\mathcal{T}_{\texttt{ID}}$ in some instance π of the Ident protocol. To exclude trivial attacks (e.g., relay attacks), $\mathcal{A}_{\text{sec}}$ is not allowed to simply forward all the messages from $\mathcal{T}_{\texttt{ID}}$ to $\mathcal{R}$ in instance π nor to corrupt $\mathcal{T}_{\texttt{ID}}$. This means that at least some of the protocol messages that made $\mathcal{R}$ to return $\texttt{ID}$ must have been computed by $\mathcal{A}_{\text{sec}}$ without knowing the secrets of $\mathcal{T}_{\texttt{ID}}$. With $\mathbf{Exp}_{\mathcal{A}_{\text{sec}}}^{\mathcal{T}\text{-aut}} = 1$ we denote the case where $\mathcal{A}_{\text{sec}}$ wins the security experiment.

Definition 4 (Tag Authentication [30]). *An RFID system (Definition 1) achieves tag authentication if for every strong adversary $\mathcal{A}_{\text{sec}}$ (Definition 3) $\Pr[\mathbf{Exp}_{\mathcal{A}_{\text{sec}}}^{\mathcal{T}\text{-aut}} = 1]$ is negligible.*

Reader Authentication. The definition of reader authentication is based on a security experiment $\mathbf{Exp}_{\mathcal{A}_{\text{sec}}}^{\mathcal{R}\text{-aut}}$ where a strong adversary $\mathcal{A}_{\text{sec}}$ (Definition 3) must successfully impersonate $\mathcal{R}$ to a legitimate tag $\mathcal{T}_{\texttt{ID}}$. Also here, to exclude trivial attacks, $\mathcal{A}_{\text{sec}}$ must achieve this without simply forwarding the protocol messages from $\mathcal{R}$ to $\mathcal{T}_{\texttt{ID}}$. This means that $\mathcal{A}_{\text{sec}}$ must have computed at least some of the protocol messages that made $\mathcal{T}_{\texttt{ID}}$ to return $\texttt{ok}$. With $\mathbf{Exp}_{\mathcal{A}_{\text{sec}}}^{\mathcal{R}\text{-aut}} = 1$ we denote the case where $\mathcal{A}_{\text{sec}}$ wins the security experiment.

Definition 5 (Reader Authentication [30]). *An RFID system (Definition 1) achieves reader authentication if for every strong adversary $\mathcal{A}_{\text{sec}}$ (Definition 3) $\Pr[\mathbf{Exp}_{\mathcal{A}_{\text{sec}}}^{\mathcal{R}\text{-aut}} = 1]$ is negligible.*

Note that both tag and reader authentication are critical properties that must be preserved even against strong adversaries.

3.4 Privacy Definition

The privacy definition of the PV-Model is very flexible and, dependent on the adversary class (see Definition 3), it covers different notions of privacy. It captures anonymity and unlinkability and focuses on the privacy leakage of the communication of tags with the reader. It is based on the existence of a simulator $\mathcal{B}$, called *blinder*, that can simulate $\mathcal{R}$ and any tag $\mathcal{T}$ without knowing their secrets such that an adversary $\mathcal{A}_{\text{prv}}$ cannot distinguish whether it is interacting with the real or the simulated RFID system. The rationale behind this simulation-based definition is that the communication of $\mathcal{T}$ and $\mathcal{R}$ does not leak any information about $\mathcal{T}$. Hence, everything $\mathcal{A}_{\text{prv}}$ observes from the interaction with $\mathcal{T}$ and $\mathcal{R}$ appears to be independent of $\mathcal{T}$ and consequently, $\mathcal{A}_{\text{prv}}$ cannot distinguish different tags based on their communication.

This privacy definition can be formalized by the following privacy experiment $\mathbf{Exp}_{\mathcal{A}_{\text{prv}}}^{\text{prv-}b} = b'$: let $\mathcal{A}_{\text{prv}}$ be an adversary according to Definition 3, l be a given security parameter and $b \in_R \{0,1\}$. In the first phase of the experiment, $\mathcal{R}$ is initialized with $(sk_{\mathcal{R}}, pk_{\mathcal{R}}, \mathtt{DB}) \leftarrow \mathsf{SetupReader}(1^l)$. The public key $pk_{\mathcal{R}}$ is given to $\mathcal{A}_{\text{prv}}$ and $\mathcal{B}$. Now, $\mathcal{A}_{\text{prv}}$ is allowed to arbitrarily interact with all oracles defined in Section 3.2. Hereby, $\mathcal{A}_{\text{prv}}$ is subject to the restrictions of its corresponding adversary class (see Definition 3). If $b = 1$, all queries to the Launch, $\mathsf{SendReader}$, $\mathsf{SendTag}$ and Result oracles are redirected to and answered by $\mathcal{B}$. Hereby, $\mathcal{B}$ can observe all queries $\mathcal{A}_{\text{prv}}$ makes to all other oracles that are not simulated by $\mathcal{B}$ and the corresponding responses ("$\mathcal{B}$ sees what $\mathcal{A}_{\text{prv}}$ sees"). After a polynomial number of oracle queries, the second phase of the experiment starts. In this second stage, $\mathcal{A}_{\text{prv}}$ cannot interact with the oracles but is given the secret table Γ of the Draw oracle. Finally, $\mathcal{A}_{\text{prv}}$ returns a bit b'.

Definition 6 (Privacy [37]). *Let C be one of the adversary classes according to Definition 3. An RFID system (Definition 1) is C-private if for every adversary $\mathcal{A}_{\text{prv}}$ of C there exists a p.p.t. algorithm $\mathcal{B}$ (blinder) such that the advantage $\mathbf{Adv}_{\mathcal{A}_{\text{prv}}}^{\text{prv}} = \left| \Pr\left[\mathbf{Exp}_{\mathcal{A}_{\text{prv}}}^{\text{prv-0}} = 1\right] - \Pr\left[\mathbf{Exp}_{\mathcal{A}_{\text{prv}}}^{\text{prv-1}} = 1\right] \right|$ of $\mathcal{A}_{\text{prv}}$ is negligible. $\mathcal{B}$ simulates the* Launch*,* $\mathsf{SendReader}$*,* $\mathsf{SendTag}$ *and* Result *oracles to $\mathcal{A}_{\text{prv}}$ without having access to $sk_{\mathcal{R}}$ and* $\mathtt{DB}$*. Hereby, all oracle queries $\mathcal{A}_{\text{prv}}$ makes and their corresponding responses are also sent to $\mathcal{B}$.*

All privacy notions defined in the PV-Model are summarized in Figure 1, which also shows their relations. It has been shown that strong privacy is impossible [37] while the technical feasibility of destructive privacy currently is an open problem.

$$\begin{tikzcd}
\text{Strong} \arrow[r, Rightarrow] \arrow[d, Rightarrow] & \text{Destructive} \arrow[r, Rightarrow] \arrow[d, Rightarrow] & \text{Forward} \arrow[r, Rightarrow] \arrow[d, Rightarrow] & \text{Weak} \arrow[d, Rightarrow] \\
\text{Narrow-Strong} \arrow[r, Rightarrow] & \text{Narrow-Destructive} \arrow[r, Rightarrow] & \text{Narrow-Forward} \arrow[r, Rightarrow] & \text{Narrow-Weak}
\end{tikzcd}$$

Fig. 1. Privacy notions defined in the PV-Model and their relations

4 Corruption with Temporary State Disclosure

We now point out a subtle weakness of the PV-Model. We show that in the PV-Model it is *impossible* to achieve *any* notion of privacy simultaneously with reader authentication (under temporary state disclosure) except for the weak and narrow-weak privacy notions. As a consequence, two of the protocols given in [30] do not achieve their claimed privacy properties.

We stress that this impossibility result is due to the fact that, according to the formal definitions of the PV-Model, the adversary can obtain the *full* state including the temporary memory of a tag by corrupting the tag *while* it is executing a protocol with the reader. Such attacks are a serious threat in practice, in particular to low-cost RFID tags, and hence must be formally considered.

Although [30] informally discusses an issue related to tag corruption during protocol execution, we show that such attacks are *not* adequately captured by the formal definitions of the PV-Model. Hence, the only achievable privacy notions are those where the adversary is not allowed to corrupt tags at all. Since in practice tag corruption is realistic, this implies that using the PV-Model is not helpful when reader authentication and a reasonable notion of privacy are needed.

Impossibility of narrow-forward privacy. To prove our first impossibility result, we need the following lemma, which we will prove in detail further below:

Lemma 1. *If there is a blinder $\mathcal{B}$ for every narrow-forward adversary $\mathcal{A}_{\text{prv}}$ such that $\mathbf{Adv}^{\text{prv}}_{\mathcal{A}_{\text{prv}}}$ is negligible (Definition 6), then $\mathcal{B}$ can be used to construct an adversary $\mathcal{A}^{\mathcal{B}}_{\text{sec}}$ such that* $\Pr[\mathbf{Exp}^{\mathcal{R}\text{-aut}}_{\mathcal{A}^{\mathcal{B}}_{\text{sec}}} = 1]$ *is non-negligible (Definition 5).*

Based on this lemma, we set up the following theorem, which we need later to prove our main impossibility result:

Theorem 1. *There is no RFID system (Definition 1) that achieves both reader authentication (Definition 5) and narrow-forward privacy (Definition 6) under temporary tag state disclosure.*

Proof (Theorem 1). Let $\mathcal{A}_{\text{prv}}$ be a narrow-forward adversary (Definition 3). Definition 6 requires the existence of a blinder $\mathcal{B}$ such that $\mathcal{A}_{\text{prv}}$ cannot distinguish $\mathcal{B}$ from the real oracles. From Lemma 1 it follows that such a $\mathcal{B}$ can be used to impersonate $\mathcal{R}$ to any legitimate tag $\mathcal{T}_{\texttt{ID}}$ with non-negligible probability. Hence, the existence of $\mathcal{B}$ contradicts reader authentication (Definition 5). □

Proof (Lemma 1). First, we show how to construct $\mathcal{A}^{\mathcal{B}}_{\text{sec}}$ from $\mathcal{B}$. Second, we prove that $\mathcal{A}^{\mathcal{B}}_{\text{sec}}$ violates reader authentication (Definition 5) if $\mathcal{B}$ is such that $\mathbf{Adv}^{\text{prv}}_{\mathcal{A}_{\text{prv}}}$ is negligible for every narrow-forward $\mathcal{A}_{\text{prv}}$ (Definition 3).

Let $q_{\mathcal{R}} \in \mathbb{N}$ with $q_{\mathcal{R}} > 0$ be the (expected) number of $\mathsf{SendReader}$ queries as specified by the Ident protocol and let $S^{\mathcal{R}}_i$ be the state of $\mathcal{R}$ after processing the i-th $\mathsf{SendReader}$ query. The initial reader state $S^{\mathcal{R}}_0$ includes the public key $pk_{\mathcal{R}}$ and the secret key $sk_{\mathcal{R}}$ of $\mathcal{R}$ as well as a pointer to the credentials database $\texttt{DB}$. Note that during the processing of a $\mathsf{SendReader}$ query, $\mathcal{R}$ can update $\texttt{DB}$. $\mathcal{R}$ can be considered as a tuple of algorithms $(\mathcal{R}^{(1)}_{\pi}, \ldots, \mathcal{R}^{(q_{\mathcal{R}})}_{\pi})$, where $\mathcal{R}^{(i)}_{\pi}$ represents the computation done by $\mathcal{R}$ when processing the i-th $\mathsf{SendReader}$ query in instance π of the Ident protocol. More formally: $(S^{\mathcal{R}}_1, m_1) \leftarrow \mathcal{R}^{(0)}_{\pi}(S^{\mathcal{R}}_0)$ and $(S^{\mathcal{R}}_{i+1}, m_{2i+1}) \leftarrow \mathcal{R}^{(i)}_{\pi}(S^{\mathcal{R}}_i, m_{2i})$ for $1 \leq i < q_{\mathcal{R}}$. Since tags are passive devices that cannot initiate communication $\mathcal{R}$ must send the first protocol message. Thus, $\mathcal{R}$ generates all protocol messages with odd indices whereas the tag $\mathcal{T}$ generates all messages with even indices. In case the Ident protocol specifies that $\mathcal{T}$ sends the last protocol message, then $m_{2q_{\mathcal{R}}-1}$ is the empty string.

Let $q_{\mathcal{T}} \in \mathbb{N}$ with $q_{\mathcal{T}} > 0$ be the (expected) number of $\mathsf{SendTag}$ queries as specified by the Ident protocol and let $S^{\mathcal{T}}_i$ be the state of $\mathcal{T}$ after processing the i-th $\mathsf{SendTag}$ query. $\mathcal{T}$ can be represented as a tuple of algorithms $(\mathcal{T}^{(1)}, \ldots, \mathcal{T}^{(q_{\mathcal{T}})})$ where $\mathcal{T}^{(i)}$ means the computation done by $\mathcal{T}$ when processing the i-th $\mathsf{SendTag}$

Alg. 1. Adversary $\mathcal{A}_{\text{sec}}^{\mathcal{B}}$ violating reader authentication

```
1: $\mathsf{CreateTag}(\mathtt{ID})$
2: $vtag \leftarrow \mathsf{Draw}(\Pr[\mathtt{ID}] = 1)$
3: $\pi \leftarrow \mathsf{Launch}()$                                        ▷ simulated by $\mathcal{B}$
4: $m_1 \leftarrow \mathsf{SendReader}(-, \pi)$                              ▷ simulated by $\mathcal{B}$
5: $i \leftarrow 1$
6: while $i < q_{\mathcal{R}}$ do
7:     if $i \leq q_{\mathcal{T}}$ then $m_{2i} \leftarrow \mathsf{SendTag}(m_{2i-1}, vtag)$    ▷ simulated by $\mathcal{B}$
8:     end if
9:     $m_{2i+1} \leftarrow \mathsf{SendReader}(m_{2i}, \pi)$                ▷ simulated by $\mathcal{B}$
10:    $i \leftarrow i + 1$
11: end while
12: $out_{\mathcal{T}_{\mathtt{ID}}} \leftarrow \mathsf{SendTag}(m_{2q_{\mathcal{R}}-1}, vtag)$   ▷ computed by $\mathcal{T}_{\mathtt{ID}}$
```

query in an instance of the Ident protocol that involves $\mathcal{T}$. More formally: $(S_{i+1}^{\mathcal{T}}, m_{2i}) \leftarrow \mathcal{T}^{(i)}(S_i^{\mathcal{T}}, m_{2i-1})$ for $1 \leq i \leq q_{\mathcal{T}}$. Note that $m_{2q_{\mathcal{T}}}$ is the empty string if Ident specifies that $\mathcal{R}$ must send the last protocol message.

The idea of $\mathcal{A}_{\text{sec}}^{\mathcal{B}}$ is to internally use $\mathcal{B}$ as a black-box to simulate the final protocol message of $\mathcal{R}$ that makes each legitimate tag $\mathcal{T}_{\mathtt{ID}}$ to accept $\mathcal{A}_{\text{sec}}^{\mathcal{B}}$ as $\mathcal{R}$. The construction of $\mathcal{A}_{\text{sec}}^{\mathcal{B}}$ is shown in Algorithm 1. First, $\mathcal{A}_{\text{sec}}^{\mathcal{B}}$ creates a legitimate tag $\mathcal{T}_{\mathtt{ID}}$ (step 1) and makes it accessible (step 2). Both steps are also shown to $\mathcal{B}$, which expects to observe all oracle queries. Then, $\mathcal{A}_{\text{sec}}^{\mathcal{B}}$ makes $\mathcal{B}$ to start a new instance π of the Ident protocol with $\mathcal{T}_{\mathtt{ID}}$ (step 3) and obtains the first protocol message m_1 generated by $\mathcal{B}$ (step 4). Now, $\mathcal{A}_{\text{sec}}^{\mathcal{B}}$ internally runs $\mathcal{B}$ that simulates both $\mathcal{T}_{\mathtt{ID}}$ and $\mathcal{R}$ until $\mathcal{B}$ returns the final reader message $m_{2q_{\mathcal{R}}-1}$ (steps 5–11). Finally, $\mathcal{A}_{\text{sec}}^{\mathcal{B}}$ sends $m_{2q_{\mathcal{R}}-1}$ to the real tag $\mathcal{T}_{\mathtt{ID}}$ (step 12). $\mathcal{A}_{\text{sec}}^{\mathcal{B}}$ succeeds if $\mathcal{T}_{\mathtt{ID}}$ accepts $\mathcal{B}$ as $\mathcal{R}$. More formally, this means that:

$$\Pr\left[\mathbf{Exp}_{\mathcal{A}_{\text{sec}}^{\mathcal{B}}}^{\mathcal{R}\text{-aut}} = 1\right] = \Pr\left[\mathsf{Ident}\left[\mathcal{T}_{\mathtt{ID}} : S_0^{\mathcal{T}_{\mathtt{ID}}};\ \mathcal{A}_{\text{sec}}^{\mathcal{B}} : -;\ * : pk_{\mathcal{R}}\right] \rightarrow \left[\mathcal{T}_{\mathtt{ID}} : \mathtt{ok};\ \mathcal{A}_{\text{sec}}^{\mathcal{B}} : \cdot\right]\right] \quad (1)$$

We stress that this indeed is a valid attack w.r.t. Definition 5 since $\mathcal{A}_{\text{sec}}$ does not just forward the protocol messages between $\mathcal{R}$ and $\mathcal{T}_{\mathtt{ID}}$.

Next, we show that narrow-forward privacy (Definition 6) ensures that $\mathcal{A}_{\text{sec}}^{\mathcal{B}}$ succeeds with non-negligible probability, i.e., that Eq. 1 is non-negligible. Note that in case Eq. 1 is negligible, this implies that with non-negligible probability $p_{\perp}$ message $m_{2q_{\mathcal{R}}-1}$ generated by $\mathcal{B}$ makes $\mathcal{T}_{\mathtt{ID}}$ to return $out_{\mathcal{T}_{\mathtt{ID}}} = \perp$. In the following, we show that if $p_{\perp}$ is non-negligible, then there is a narrow-forward adversary $\mathcal{A}_{\text{prv}}$ that has non-negligible advantage $\mathbf{Adv}_{\mathcal{A}_{\text{prv}}}^{\text{prv}}$ to distinguish $\mathcal{B}$ form the real oracles, which contradicts narrow-forward privacy (Definition 6). The construction of $\mathcal{A}_{\text{prv}}$ is shown in Algorithm 2. First, $\mathcal{A}_{\text{prv}}$ creates a legitimate tag $\mathcal{T}_{\mathtt{ID}}$ (step 1) and makes it accessible (step 2). Then, $\mathcal{A}_{\text{prv}}$ makes $\mathcal{R}$ to start a new instance π of the Ident protocol with $\mathcal{T}_{\mathtt{ID}}$ (step 3) and obtains the first protocol message m_1 from $\mathcal{R}$ (step 4). Now, $\mathcal{A}_{\text{prv}}$ eavesdrops on the execution of the Ident protocol up to to the point *after* $\mathcal{R}$ has sent its last protocol message $m_{2q_{\mathcal{R}}-1}$ (steps 5–11) and corrupts $\mathcal{T}_{\mathtt{ID}}$ just *before* $\mathcal{T}_{\mathtt{ID}}$ received $m_{2q_{\mathcal{R}}-1}$ (step 12). Next, $\mathcal{A}_{\text{prv}}$ performs the computation $\mathcal{T}_{\mathtt{ID}}$ would have done on receipt of $m_{2q_{\mathcal{R}}-1}$

Alg. 2. Narrow-forward adversary $\mathcal{A}_{\text{prv}}$

```
 1: $\mathsf{CreateTag}(\mathtt{ID})$
 2: $vtag \leftarrow \mathsf{Draw}(\Pr[\mathtt{ID}] = 1)$
 3: $\pi \leftarrow \mathsf{Launch}(\,)$
 4: $m_1 \leftarrow \mathsf{SendReader}(-, \pi)$
 5: $i \leftarrow 1$
 6: while $i < q_{\mathcal{R}}$ do
 7:     if $i \leq q_{\mathcal{T}}$ then $m_{2i} \leftarrow \mathsf{SendTag}(m_{2i-1}, vtag)$
 8:     end if
 9:     $m_{2i+1} \leftarrow \mathsf{SendReader}(m_{2i}, \pi)$
10:     $i \leftarrow i + 1$
11: end while
12: $S_{q_{\mathcal{R}}}^{\mathcal{T}_{\mathtt{ID}}} \leftarrow \mathsf{Corrupt}(vtag)$
13: $out_{\mathcal{T}_{\mathtt{ID}}} \leftarrow \mathcal{T}_{\mathtt{ID}}{}^{(q_{\mathcal{R}})}(S_{q_{\mathcal{R}}}^{\mathcal{T}_{\mathtt{ID}}}, m_{2q_{\mathcal{R}}-1})$
14: if $out_{\mathcal{T}_{\mathtt{ID}}} = \mathsf{ok}$ then return 0
15: else return 1
16: end if
```

(step 13). If this computation results in $out_{\mathcal{T}_{\mathtt{ID}}} = \mathsf{ok}$, $\mathcal{A}_{\text{prv}}$ returns 0 to indicate that it interacted with the real oracles (step 14). Otherwise, $\mathcal{A}_{\text{prv}}$ indicates the presence of $\mathcal{B}$ by returning 1 (step 15). Note that $\mathcal{A}_{\text{prv}}$ indeed is a narrow-forward adversary (Definition 3) since $\mathcal{A}_{\text{prv}}$ never queries Result and none of the oracles defined in Section 3.2 after corrupting $\mathcal{T}_{\mathtt{ID}}$.

Next, we show that $\mathcal{A}_{\text{prv}}$ has non-negligible advantage $\mathbf{Adv}_{\mathcal{A}_{\text{prv}}}^{\text{prv}}$ if $p_{\perp}$ is non-negligible. Therefore, we first consider the case where $\mathcal{A}_{\text{prv}}$ interacts with the real oracles. Since $\mathcal{T}_{\mathtt{ID}}$ is legitimate, it follows from correctness (Definition 2) that $out_{\mathcal{T}_{\mathtt{ID}}} = \mathsf{ok}$ with overwhelming probability p_{ok}. Hence, $\Pr\left[\mathbf{Exp}_{\mathcal{A}_{\text{prv}}}^{\text{prv-0}} = 1\right] = 1 - p_{\mathsf{ok}}$ is negligible. Now, consider the case where $\mathcal{A}_{\text{prv}}$ interacts with $\mathcal{B}$. Note that by the contradicting hypothesis, $\mathcal{B}$ generates a protocol message $m_{2q_{\mathcal{R}}-1}$ that makes $\mathcal{T}_{\mathtt{ID}}$ to return $out_{\mathcal{T}_{\mathtt{ID}}} = \perp$ with non-negligible probability $p_{\perp}$. Thus, we have $\Pr\left[\mathbf{Exp}_{\mathcal{A}_{\text{prv}}}^{\text{prv-1}} = 1\right] = p_{\perp}$. Hence, it follows that $\mathbf{Adv}_{\mathcal{A}_{\text{prv}}}^{\text{prv}} = \left|1 - p_{\mathsf{ok}} - p_{\perp}\right|$. Note that due to correctness both p_{ok} is overwhelming and by assumption $p_{\perp}$ is non-negligible. Hence, $\mathbf{Adv}_{\mathcal{A}_{\text{prv}}}^{\text{prv}}$ is non-negligible, which contradicts narrow-forward privacy (Definition 6). In turn, this means that narrow-forward privacy ensures that Eq. 1 is non-negligible, which finishes the proof. □

Since the impossibility of narrow-forward privacy (Theorem 1), implies the impossibility of all other stronger privacy notions (see Figure 1), we have the following corollary, which corresponds to the first main claim of this paper:

Corollary 1. *In the PV-Model there is no RFID system (Definition 1) that achieves both reader authentication (Definition 5) and any privacy notion that is different from weak and narrow-weak privacy (Definition 6) under temporary state disclosure.*

5 Corruption without Temporary State Disclosure

Our first impossibility result shows that the PV-Model requires further assumptions to evaluate the privacy properties of RFID systems where tag corruption is of concern. A natural question therefore is, whether one can achieve mutual authentication along with some form of privacy, if the temporary tag state is *not* disclosed. Hence, in this section we consider the case where corruption *only* reveals the persistent tag state but *no* information on the temporary tag state.

The attack and the impossibility result shown in Section 4 critically use the fact that in the PV-Model an adversary $\mathcal{A}_{\text{prv}}$ can learn the temporary state of a tag during the Ident protocol. This allows $\mathcal{A}_{\text{prv}}$ to verify the response of $\mathcal{R}$ (that may have been simulated by $\mathcal{B}$) and hence, due to reader authentication (Definition 5), $\mathcal{A}_{\text{prv}}$ can distinguish with non-negligible advantage between the real oracles and $\mathcal{B}$. However, if $\mathcal{A}_{\text{prv}}$ cannot obtain temporary tag states, it cannot perform this verification. Hence, the impossibility result we proved in Section 4 does not necessarily hold if the temporary state is safe to corruption.

Impossibility of narrow-strong privacy. We now show our second impossibility result: in the PV-Model, it is *impossible* to achieve narrow-strong privacy along with reader authentication. This means that even in case the adversary cannot obtain the temporary tag state, the most challenging privacy notion defined in [30] (narrow-strong privacy) still remains unachievable. This implies a conceptually different weakness of the claimed narrow-strong private protocol in [30].

Theorem 2. *In the PV-Model there is no RFID system (Definition 1) that fulfills both reader authentication (Definition 5) and narrow-strong privacy (Definition 6).*

Proof (Theorem 2). Narrow-strong privacy (Definition 6) requires the existence of a blinder $\mathcal{B}$ that simulates the Launch, $\mathsf{SendReader}$ and $\mathsf{SendTag}$ oracles such that every narrow-strong adversary $\mathcal{A}_{\text{prv}}$ has negligible advantage $\mathbf{Adv}^{\text{prv}}_{\mathcal{A}_{\text{prv}}}$ to distinguish $\mathcal{B}$ from the real oracles. We now show that $\mathcal{B}$ can be used to construct an algorithm $\mathcal{A}^{\mathcal{B}}_{\text{sec}}$ that violates reader authentication (Definition 5).

The construction of $\mathcal{A}^{\mathcal{B}}_{\text{sec}}$ is as shown in Algorithm 3. First, $\mathcal{A}^{\mathcal{B}}_{\text{sec}}$ creates a legitimate tag $\mathcal{T}_{\mathtt{ID}}$ (step 1), makes it accessible (step 2), and corrupts it (step 3). These three steps are also shown to $\mathcal{B}$, which expects to observe all oracle queries. Then, $\mathcal{A}^{\mathcal{B}}_{\text{sec}}$ makes $\mathcal{B}$ to start a new instance π of the Ident protocol with $\mathcal{T}_{\mathtt{ID}}$ (step 4) and obtains the first protocol message m_1 generated by $\mathcal{B}$ (step 5). Now, $\mathcal{A}^{\mathcal{B}}_{\text{sec}}$ internally runs $\mathcal{B}$ that simulates *vtag* and $\mathcal{R}$ until $\mathcal{B}$ returns the final reader message $m_{2q_{\mathcal{R}}-1}$ (steps 6–12). Finally, $\mathcal{A}^{\mathcal{B}}_{\text{sec}}$ sends $m_{2q_{\mathcal{R}}-1}$ to the real tag $\mathcal{T}_{\mathtt{ID}}$ (step 13). $\mathcal{A}^{\mathcal{B}}_{\text{sec}}$ succeeds if $\mathcal{T}_{\mathtt{ID}}$ accepts $m_{2q_{\mathcal{R}}-1}$ and returns $out_{\mathcal{T}_{\mathtt{ID}}} = \mathsf{ok}$, which means that $\mathcal{T}_{\mathtt{ID}}$ accepts $\mathcal{B}$ as $\mathcal{R}$. More formally, this means that:

$$\Pr\left[\mathbf{Exp}^{\mathcal{R}\text{-aut}}_{\mathcal{A}^{\mathcal{B}}_{\text{sec}}} = 1\right] = \Pr\left[\mathsf{Ident}\left[\mathcal{T}_{\mathtt{ID}} : S_0^{\mathcal{T}_{\mathtt{ID}}};\ \mathcal{A}^{\mathcal{B}}_{\text{sec}} : -;\ * : pk_{\mathcal{R}}\right] \to \left[\mathcal{T}_{\mathtt{ID}} : \mathsf{ok};\ \mathcal{A}^{\mathcal{B}}_{\text{sec}} : \cdot\right]\right] \quad (2)$$

We stress that this indeed is a valid attack w.r.t. Definition 5 since $\mathcal{A}_{\text{sec}}$ does not just forward the protocol messages between $\mathcal{R}$ and $\mathcal{T}_{\mathtt{ID}}$.

Alg. 3. Adversary $\mathcal{A}_{\text{sec}}^{\mathcal{B}}$ violating reader authentication

```
$\mathsf{CreateTag}(\mathtt{ID})$
$vtag \leftarrow \mathsf{Draw}(\Pr[\mathtt{ID}] = 1)$
$S_0^{\mathcal{T}_{\mathtt{ID}}} \leftarrow \mathsf{Corrupt}(vtag)$
$\pi \leftarrow \mathsf{Launch}(\,)$                                          ▷ simulated by $\mathcal{B}$
$m_1 \leftarrow \mathsf{SendReader}(-, \pi)$                                  ▷ simulated by $\mathcal{B}$
$i \leftarrow 1$
while $i < q_{\mathcal{R}}$ do
    if $i \leq q_{\mathcal{T}}$ then $m_{2i} \leftarrow \mathsf{SendTag}(m_{2i-1}, vtag)$
    end if
    $m_{2i+1} \leftarrow \mathsf{SendReader}(m_{2i}, \pi)$                      ▷ simulated by $\mathcal{B}$
    $i \leftarrow i + 1$
end while
$out_{\mathcal{T}_{\mathtt{ID}}} \leftarrow \mathsf{SendTag}(m_{2q_{\mathcal{R}}-1}, vtag)$   ▷ computed by $\mathcal{T}_{\mathtt{ID}}$
```

From reader authentication (Definition 5) it follows that Eq. 2 must be negligible. However, this implies that with overwhelming probability $\mathcal{B}$ generates at least one protocol message that makes $\mathcal{T}_{\mathtt{ID}}$ to finally return $out_{\mathcal{T}_{\mathtt{ID}}} = \bot$. Let p_t be the probability that this is the case for message m_{2t-1} for some $t \in \{1, \ldots, q_{\mathcal{T}}\}$. We now show a narrow-strong adversary $\mathcal{A}_{\text{prv}}$ that succeeds with non-negligible advantage $\mathbf{Adv}_{\mathcal{A}_{\text{prv}}}^{\text{prv}}$ if p_t is non-negligible, which contradicts narrow-strong privacy (Definition 6). The construction of $\mathcal{A}_{\text{prv}}$ is shown in Algorithm 4. First, $\mathcal{A}_{\text{prv}}$ creates a legitimate tag $\mathcal{T}_{\mathtt{ID}}$ (step 1), makes it accessible (step 2), and corrupts it (step 3). Note that by a $\mathsf{Corrupt}$ query, $\mathcal{A}_{\text{prv}}$ only learns the persistent tag state $S_0^{\mathcal{T}_{\mathtt{ID}}}$ of $\mathcal{T}_{\mathtt{ID}}$. Then, $\mathcal{A}_{\text{prv}}$ makes $\mathcal{R}$ to start an instance π of the Ident protocol with $\mathcal{T}_{\mathtt{ID}}$ (step 4) and obtains the first protocol message m_1 from $\mathcal{R}$ (step 5). Now, $\mathcal{A}_{\text{prv}}$ guesses t (step 6) and simulates $\mathcal{T}_{\mathtt{ID}}$ (using $S_0^{\mathcal{T}_{\mathtt{ID}}}$) in the Ident protocol up to the point where $\mathsf{SendReader}$ returns message m_{2t-1} (steps 7–13). Next, $\mathcal{A}_{\text{prv}}$ performs the computation $\mathcal{T}_{\mathtt{ID}}$ would have done on receipt of message m_{2t-1} (step 14). Finally, $\mathcal{A}_{\text{prv}}$ returns either 0 to indicate that it interacted with the real oracles (step 15) or 1 to indicate the presence of $\mathcal{B}$ (step 16).

Next, we show that $\mathcal{A}_{\text{prv}}$ has non-negligible $\mathbf{Adv}_{\mathcal{A}_{\text{prv}}}^{\text{prv}}$ if $p_{\bot}$ is non-negligible. Therefore, we first consider the case where $\mathcal{A}_{\text{prv}}$ interacts with the real oracles. Since $\mathcal{T}_{\mathtt{ID}}$ is legitimate, it follows form correctness (Definition 2) that $out_{\mathcal{T}_{\mathtt{ID}}} = \mathsf{ok}$ holds with overwhelming probability p_{ok}. This means that $\Pr\left[\mathbf{Exp}_{\mathcal{A}_{\text{prv}}}^{\text{prv-0}} = 1\right] = 1 - p_{\mathsf{ok}}$ is negligible. Now, consider the case where $\mathcal{A}_{\text{prv}}$ interacts with $\mathcal{B}$. Note that by the contradicting hypothesis, with non-negligible probability p_t $\mathcal{B}$ generates a message m_{2t-1} that makes $\mathcal{T}_{\mathtt{ID}}$ to return $out_{\mathcal{T}_{\mathtt{ID}}} = \bot$. Moreover, $\mathcal{A}_{\text{prv}}$ guesses t with probability of at least $1/q_{\mathcal{T}}$. Thus, we have $\Pr\left[\mathbf{Exp}_{\mathcal{A}_{\text{prv}}}^{\text{prv-1}} = 1\right] \geq \frac{p_t}{q_{\mathcal{T}}}$. Hence, it follows that $\mathbf{Adv}_{\mathcal{A}_{\text{prv}}}^{\text{prv}} \geq |1 - p_{\mathsf{ok}} - \frac{p_t}{q_{\mathcal{T}}}|$. Note that due to correctness p_{ok} is overwhelming while p_t is non-negligible by assumption and $q_{\mathcal{T}}$ is polynomially bounded. Hence, $\mathbf{Adv}_{\mathcal{A}_{\text{prv}}}^{\text{prv}}$ is non-negligible, which contradicts narrow-strong privacy (Definition 6). □

Alg. 4. Narrow-strong adversary $\mathcal{A}_{\text{prv}}$

```
$\mathsf{CreateTag}(\mathtt{ID})$
$vtag \leftarrow \mathsf{Draw}(\Pr[\mathtt{ID}] = 1)$
$S_0^{\mathcal{T}_{\mathtt{ID}}} \leftarrow \mathsf{Corrupt}(vtag)$
$\pi \leftarrow \mathsf{Launch}(\,)$
$m_1 \leftarrow \mathsf{SendReader}(-, \pi)$
$t \in \{1, \ldots, q_{\mathcal{T}}\}$
$i \leftarrow 1$
while $i < t$ do
    $(S_{i+1}^{\mathcal{T}_{\mathtt{ID}}}, m_{2i}) \leftarrow \mathcal{T}_{\mathtt{ID}}{}^{(i)}(S_i^{\mathcal{T}_{\mathtt{ID}}}, m_{2i-1})$
    if $i < q_{\mathcal{R}}$ then $m_{2i+1} \leftarrow \mathsf{SendReader}(m_{2i}, \pi)$
    end if
    $i \leftarrow i + 1$
end while
$out_{\mathcal{T}_{\mathtt{ID}}} \leftarrow \mathcal{T}_{\mathtt{ID}}{}^{(t)}(S_t^{\mathcal{T}_{\mathtt{ID}}}, m_{2t-1})$
if $out_{\mathcal{T}_{\mathtt{ID}}} = \mathtt{ok}$ then return 0
else return 1
end if
```

6 Impossibility Results for Resettable and Stateless Tags

It is well known (see [9] for details and in particular [5] for identification schemes) that standard security notions do not work anymore when the adversary can manipulate the device that is running an honest party protocol, in particular when the adversary can reset the internal state of the device. To face this security issue, Canetti et al. [9] considered the concept of *resettability* for obtaining a security notion that is resilient to "reset attacks", e.g., attacks where the adversary can force a device to reuse the same randomness. The crucial importance of this notion is proved by several results (see, e.g., [5,9,13,6,17]) with the focus on obtaining feasibility results and efficient constructions for proof systems and identification schemes in such hostile settings. *Reset attacks* have been motivated in particular by the use of smart cards since some specific smart cards, when disconnected from power, go back to their initial state and perform their computations using the same randomness they already used before. However, the concept of a reset attacks can have a wider applicability. In particular reset attacks are always possible when the adversary controls the environment and can therefore force a stateless device to use the same randomness in different executions of a protocol.

As discussed in Section 2, most RFID tags in practice are low-cost devices that are usually not protected against physical tampering. Moreover, the randomness generator of a real-life RFID tag has already been successfully attacked [15]. Therefore, it is interesting to investigate the impact of reset attacks on the security and privacy of RFID systems.

In this section, we focus on the effect of reset attacks on privacy as defined in both the PV-Model [30] and the model it is based on [37]. Therefore, we first extend the formal adversary model in [37,30] to capture reset attacks. Then, we show that any privacy notion as defined in Definition 6 is spoiled when an adversary is able to launch reset attacks. We finally show that, when restricting the power of the adversary to the capability of resetting *only* the persistent state of a tag, i.e., the randomness of the tag is out of the control of the adversary, it is impossible to achieve destructive privacy.

6.1 Impossibility of Narrow-Weak Privacy under Reset Attacks

In order to extend the model in [37,30] to capture reset attacks, we add an additional oracle $\mathsf{Reset}(vtag)$ to the adversary model shown in Section 3.2. This oracle allows the adversary to reset the randomness and the state of a tag *vtag* to their initial values. We stress out that resetting a tag is a mere adversarial action and is never performed by honest parties. Thus we do not require that such an action must be carried out efficiently, instead according to the result showed in [5,9] we assume that it can be carried out in polynomial time. Note that, as for the Corrupt oracle, the Reset oracle is not simulated by the blinder $\mathcal{B}$ (see Definition 6) but is observed by it.

Now we are ready to formalize the impossibility of achieving any privacy notion in the extended model of [37,30] when the adversary can perform reset attacks against tags.

Theorem 3. *In the model of [37,30], no privacy notion (Definition 6) is achievable if the adversary is allowed to query the* Reset *oracle.*

Proof (Theorem 3). We show a narrow-weak adversary $\mathcal{A}_{\text{prv}}$ that can distinguish with non-negligible advantage $\mathbf{Adv}^{\text{prv}}_{\mathcal{A}_{\text{prv}}}$ whether it is interacting with the real oracles or a blinder $\mathcal{B}$. The construction of $\mathcal{A}_{\text{prv}}$ is shown in Algorithm 5. First, $\mathcal{A}_{\text{prv}}$ creates two legitimate tags $\mathcal{T}_{\text{ID}0}, \mathcal{T}_{\text{ID}1}$ (steps 1–2) and makes one of them accessible (step 3). Then $\mathcal{A}_{\text{prv}}$ eavesdrops a complete execution protocol of the Ident protocol between *vtag* and $\mathcal{R}$ (steps 4-11). We define τ as the complete transcript of the protocol execution. Note that τ contains the messages sent by both $\mathcal{R}$ and *vtag*. Now, $\mathcal{A}_{\text{prv}}$ resets the state of *vtag* by querying the Reset oracle (step 12) and makes *vtag* inaccessible again by querying the Free oracle (step 13). Next, $\mathcal{A}_{\text{prv}}$ makes a randomly chosen tag *vtag*$'$ accessible (step 14) and then executes a complete run of the Ident protocol with *vtag*$'$ simulating $\mathcal{R}$ (steps 15–18). To simulate $\mathcal{R}$, $\mathcal{A}_{\text{prv}}$ uses the messages that have been sent by $\mathcal{R}$ in the previous execution according to the transcript τ. Finally, $\mathcal{A}_{\text{prv}}$ obtains a new protocol transcript τ'. If the same tag has played both times, then $\mathcal{A}_{\text{prv}}$ expects that the transcripts τ and τ' are the same due to the Reset oracle. The idea is that $\mathcal{B}$ has no information about which tag has been drawn in step 14 (the resetted one or the other one). Thus, $\mathcal{B}$ can at most guess which tag has been chosen when answering the SendTag query in the second protocol execution.

In the following we show that $\mathcal{A}_{\text{prv}}$ has non-negligible advantage $\mathbf{Adv}^{\text{prv}}_{\mathcal{A}_{\text{prv}}}$ of distinguishing between $\mathcal{B}$ and real oracles, which violates narrow-weak privacy. First, we consider the case where $\mathcal{A}_{\text{prv}}$ interacts with the real oracles. It is easy to see that in this case the attack is always successful. Indeed, if $\mathcal{A}_{\text{prv}}$ interacts with the same tag in both executions of the Ident protocol, then, due to the

Alg. 5. Experiment with a narrow-weak adversary $\mathcal{A}_{\mathrm{prv}}$

```
CreateTag($\mathtt{ID}_0$)
CreateTag($\mathtt{ID}_1$)
$vtag \leftarrow$ Draw($\Pr[\mathtt{ID}_0] = \frac{1}{2}, \Pr[\mathtt{ID}_1] = \frac{1}{2}$)
$m_1 \leftarrow$ SendReader($-, \pi$)
$i \leftarrow 1$
while $i < q_{\mathcal{R}}$ do
    if $i \leq q_{\mathcal{T}}$ then $m_{2i} \leftarrow$ SendTag($m_{2i-1}, vtag$)
    end if
    $m_{2i+1} \leftarrow$ SendReader($m_{2i}, \pi$)
    $i \leftarrow i + 1$
end while
Reset($vtag$)
Free($vtag$)
$vtag' \leftarrow$ Draw($\Pr[\mathtt{ID}_0] = \frac{1}{2}, \Pr[\mathtt{ID}_1] = \frac{1}{2}$)
$i \leftarrow 1$
while $i \leq q_{\mathcal{T}}$ do $m_{2i} \leftarrow$ SendTag($m_{2i-1}, vtag'$)
    $i \leftarrow i + 1$
end while
if $\tau = \tau'$ then $out_{\mathcal{A}} \leftarrow 1$
else $out_{\mathcal{A}} \leftarrow 0$
end if
return $\big(\Gamma[vtag] = \Gamma[vtag'] \wedge out_{\mathcal{A}}\big) \vee \big(\Gamma[vtag] \neq \Gamma[vtag'] \wedge \overline{out_{\mathcal{A}}}\big)$
```

Reset query, challenging $vtag'$ with the *same* messages must generate the same protocol transcript. Thus, after $\mathcal{A}_{\mathrm{prv}}$ is given the hidden table Γ, one of the two conditions must hold: either $\mathcal{A}_{\mathrm{prv}}$ has (i) interacted with the same tag twice and the transcripts match (which is always true in case $\Gamma[vtag] = \Gamma[vtag']$), or (ii) the tag involved in the second execution of Ident is not the resetted tag and the protocol transcripts are different (which holds with overwhelming probability in case $\Gamma[vtag] \neq \Gamma[vtag']$ due to tag authentication, since otherwise $\mathcal{A}_{\mathrm{prv}}$ can create a faked tag state that can be used to generate the messages of a legitimate tag with non-negligible probability). Hence, $\mathcal{A}_{\mathrm{prv}}$ succeeds in $\mathbf{Exp}_{\mathcal{A}_{\mathrm{prv}}}^{\mathrm{prv}\text{-}0}$ with probability $1 - \epsilon(l)$ where ϵ is a negligible function in the security parameter l. Formally, $\Pr\big[\mathbf{Exp}_{\mathcal{A}_{\mathrm{prv}}}^{\mathrm{prv}\text{-}0} = 1\big] = \Pr\big[\big(\Gamma[vtag] = \Gamma[vtag']\big) \wedge out_{\mathcal{A}}\big] + \Pr\big[\big(\Gamma[vtag] \neq \Gamma[vtag']\big) \wedge \overline{out_{\mathcal{A}}}\big] = \frac{1}{2} \cdot 1 + \frac{1}{2} \cdot (1 - \epsilon(l)) = 1 - \epsilon(l)/2$. Next we consider the case where the SendTag oracle is simulated by $\mathcal{B}$. In this case any $\mathcal{B}$ can at most guess which tag has been selected by Draw. Hence, the probability that $\mathcal{A}_{\mathrm{prv}}$ wins the experiment $\mathbf{Exp}_{\mathcal{A}_{\mathrm{prv}}}^{\mathrm{prv}\text{-}1}$ is at most $\Pr\big[\mathbf{Exp}_{\mathcal{A}_{\mathrm{prv}}}^{\mathrm{prv}\text{-}1} = 1\big] = \Pr\big[\big(\Gamma[vtag] = \Gamma[vtag']\big) \wedge out_{\mathcal{A}}\big] + \Pr\big[\big(\Gamma[vtag] \neq \Gamma[vtag']\big) \wedge \overline{out_{\mathcal{A}}}\big] \leq \frac{1}{2} \cdot \frac{1}{2} + \frac{1}{2} \cdot \frac{1}{2} = \frac{1}{2}$. According to Definition 6, from the above probability it follows that $\mathcal{A}_{\mathrm{prv}}$ has non-negligible advantage $\mathbf{Adv}_{\mathcal{A}_{\mathrm{prv}}}^{\mathrm{prv}} \geq 1 - \epsilon(l)/2 - \frac{1}{2}$ to distinguish between $\mathcal{B}$ and the real oracles. □

Alg. 6. Narrow-forward adversary $\mathcal{A}_{\text{prv}}$

```
$\mathsf{CreateTag}(\mathtt{ID})$
$vtag \leftarrow \mathsf{Draw}(\Pr[\mathtt{ID}] = 1)$
$\mathsf{Free}(vtag)$
$vtag \leftarrow \mathsf{Draw}(\Pr[\mathtt{ID}] = 1)$
$t \in \{1, \ldots, q_{\mathcal{T}}\}$
$m_1 \leftarrow \mathsf{SendReader}(-, \pi)$
$i \leftarrow 1$
while $i \leq t$ do
    $m_{2i} \leftarrow \mathsf{SendTag}(m_{2i-1}, vtag)$
    $m_{2i+1} \leftarrow \mathsf{SendReader}(m_{2i}, \pi)$
    $i \leftarrow i + 1$
end while
$S \leftarrow \mathsf{Corrupt}(vtag)$
return 1 if and only if the temporary state in $S$ is empty
```

6.2 Impossibility of Destructive Privacy with Stateless Tags

In this section we show that destructive privacy is impossible to achieve in the model of [37,30] when tags are *stateless*, i.e., when their persistent state cannot be updated. This implies that destructive privacy is impossible when an adversary can reset the persistent state of a tag to its original value: by resetting a tag, the adversary can interact with a tag that uses the same state several times, which corresponds to an experiment with a stateless tag. We stress that in a stateless RFID scheme the Free oracle erases any temporary information stored on the tag. Otherwise there would be an updatable information that survives even when a tag is not powered, and thus the tag would be stateful.

We recall that in our previous notation we associate $S_i^{\mathcal{T}}$ to the full state (including both the persistent *and* temporary state) of a tag when playing the i-th message from the moment it has been drawn, i.e., powered on. We start by giving a useful preliminary lemma.

Lemma 2. *In any stateless narrow-forward RFID scheme the temporary tag state is always empty.*

Proof (Lemma 2). To prove the lemma we show in Algorithm 6 that if there exists a non-empty temporary tag state, then there exists a narrow-forward adversary $\mathcal{A}_{\text{prv}}$ that distinguishes between the real oracles and $\mathcal{B}$. We stress that for a stateless tag, due to the Free query, the output S returned by a Corrupt$(vtag)$ query played immediately after a Draw query corresponds to the persistent state generated by the CreateTag oracle. Clearly, when interacting with the real oracles the output of $\mathcal{A}_{\text{prv}}$ is different than 1 with non-negligible probability. Indeed, since stateless tags are allowed to have some non-empty temporary state, there exists at least one round, which can be guessed with non-negligible probability by the selection of t, that, when followed by the Corrupt query, reveals to $\mathcal{A}_{\text{prv}}$ that the temporary state of the tag is not empty.

During interaction with the blinder $\mathcal{B}$ the tag does not play any round, as all SendTag queries are simulated by $\mathcal{B}$. Therefore, the output of the above experiment is always equal to 1, which shows that $\mathcal{A}_{\mathrm{prv}}$ is successful and the claim holds. □

Due to Lemma 2 we can assume that $(S^{\mathcal{T}_{\mathtt{ID}}}, \cdot) \leftarrow {\mathcal{T}_{\mathtt{ID}}}^{(i)}(S^{\mathcal{T}_{\mathtt{ID}}}, \cdot)$, i.e., the new state after each round is always identical to the previous one. Recall that an RFID scheme is stateless if the persistent tag state is not allowed to change over time. In this section we show that when the tag state does not change, then achieving destructive privacy is impossible.

Theorem 4. *There is no stateless RFID system (Definition 1) that achieves destructive privacy (Definition 6).*

Proof (Theorem 4). Recall that destructive privacy implies forward privacy (see Figure 1). We prove that a stateless RFID system cannot achieve destructive and narrow-forward privacy at the same time. The proof is by contradiction. Note that a destructive private stateless RFID system implies the existence of a blinder $\mathcal{B}$ such that $\mathcal{A}_{\mathrm{prv}}$ fails in distinguishing the real oracles from their simulation by $\mathcal{B}$ with overwhelming probability. Thus, we first show a destructive adversary $\mathcal{A}_{\mathrm{prv}}$ for which there must exist a successful blinder, that we denote by $\mathcal{B}_D$. Then, we construct a narrow-forward adversary $\mathcal{A}_{\mathrm{prv}}^{\mathcal{B}_D}$ that internally uses $\mathcal{B}_D$ to violate forward privacy. Hence, we obtain a contradiction.

Since we are considering *stateless* tags, we assume that at each step of the tag algorithm the persistent state remains unchanged. Formally, this means that $\mathcal{T}$ can be represented as a tuple of algorithms $(\mathcal{T}^{(1)}, \ldots, \mathcal{T}^{(q_{\mathcal{T}})})$ where $\mathcal{T}^{(i)}$ means the computation done by $\mathcal{T}$ when processing the i-th SendTag query in an instance of the Ident protocol that involves $\mathcal{T}$. We have $m_{2i} \leftarrow \mathcal{T}^{(i)}(S^{\mathcal{T}}, m_{2i-1})$ for $1 \leq i \leq q_{\mathcal{T}}$ where $q_{\mathcal{T}}$ is an upper bound on the number of messages sent by $\mathcal{T}$ during the protocol.

Let $\mathcal{A}_{\mathrm{prv}}$ be the destructive adversary defined in Algorithm 7. Informally, the attack is the following: $\mathcal{A}_{\mathrm{prv}}$ faithfully forwards the messages generated by $\mathcal{R}$ and $\mathcal{T}$, up to a certain (randomly chosen) round t of the Ident protocol execution. Then $\mathcal{A}_{\mathrm{prv}}$ corrupts $\mathcal{T}$ and gets its state. Since $\mathcal{A}_{\mathrm{prv}}$ is destructive, it is not allowed to query any other oracle for $\mathcal{T}$ after corrupting $\mathcal{T}$ but $\mathcal{A}_{\mathrm{prv}}$ can still compute the remaining protocol messages of $\mathcal{T}$ by running the tag algorithm with the state obtained by corruption. Then $\mathcal{A}_{\mathrm{prv}}$ picks a state S with the same distribution used by CreateTag (i.e., SetupTag) with the purpose of distinguishing if it is interacting with the real oracles or $\mathcal{B}_D$. Then $\mathcal{A}_{\mathrm{prv}}$ randomly selects one of the two states and continues the protocol execution running the tag algorithm with the chosen state until the end of the protocol. The main idea is that when $\mathcal{A}_{\mathrm{prv}}$ runs the tag algorithm with the state obtained through the Corrupt query, then, due to correctness (Definition 2), $\mathcal{R}$ will accept, i.e., the Result query outputs 1 with overwhelming probability, while $\mathcal{R}$ will reject otherwise.

Formally, $\mathcal{A}_{\mathrm{prv}}$ behaves as follows: First, $\mathcal{A}_{\mathrm{prv}}$ creates two legitimate tags $\mathcal{T}_{\mathtt{ID}}$ (step 1 and step 2) and makes one of them accessible (step 3). Then, $\mathcal{A}_{\mathrm{prv}}$ asks

Alg. 7. Destructive adversary $\mathcal{A}_{\text{prv}}$

```
 1: $\mathsf{CreateTag}(\mathtt{ID})$
 2: $\mathsf{CreateTag}(\mathtt{ID}')$
 3: $vtag \leftarrow \mathsf{Draw}(\Pr[\mathtt{ID}] = \frac{1}{2}, \Pr[\mathtt{ID}'] = \frac{1}{2})$
 4: $\pi \leftarrow \mathsf{Launch}(\,)$
 5: $m_1 \leftarrow \mathsf{SendReader}(-, \pi)$
 6: $j_{\mathcal{R}} \in_R \{1, \ldots, q_{\mathcal{R}}\}$
 7: $i \leftarrow 1$
 8: while $i < j_{\mathcal{R}}$ do $m_{2i} \leftarrow \mathsf{SendTag}(m_{2i-1}, vtag)$
 9:     $m_{2i+1} \leftarrow \mathsf{SendReader}(m_{2i}, \pi)$
10:     $i \leftarrow i + 1$
11: end while
12: $S^{\mathcal{T}_{\text{ID}}} \leftarrow \mathsf{Corrupt}(vtag)$
13: $b \in_R \{0, 1\}$
14: if $b = 1$ then
15:     $m_{2j_{\mathcal{R}}} \leftarrow {\mathcal{T}_{\text{ID}}}^{(j_{\mathcal{R}})}(S^{\mathcal{T}_{\text{ID}}}, m_{2j_{\mathcal{R}}-1})$
16: else
17:     pick a state $S$ with the same distribution used by CreateTag (i.e., SetupTag)
18:     $S^{\mathcal{T}_{\text{ID}}} \leftarrow S$
19:     $m_{2j_{\mathcal{R}}} \leftarrow {\mathcal{T}_{\text{ID}}}^{(j_{\mathcal{R}})}(S^{\mathcal{T}_{\text{ID}}}, m_{2j_{\mathcal{R}}-1})$
20: end if
21: $m_{2j_{\mathcal{R}}+1} \leftarrow \mathsf{SendReader}(m_{2j_{\mathcal{R}}}, \pi)$
22: $i \leftarrow j_{\mathcal{R}} + 1$
23: while $i < q_{\mathcal{R}}$ do
24:     if $i \leq q_{\mathcal{T}}$ then $m_{2i} \leftarrow {\mathcal{T}_{\text{ID}}}^{(i)}(S^{\mathcal{T}_{\text{ID}}}, m_{2i-1})$
25:     end if
26:     $m_{2i+1} \leftarrow \mathsf{SendReader}(m_{2i}, \pi)$
27:     $i \leftarrow i + 1$
28: end while
29: return $\big(\mathsf{Result}(\pi) \wedge b\big) \vee \big(\overline{\mathsf{Result}(\pi)} \wedge \bar{b}\big)$
```

$\mathcal{R}$ to start a new instance π of the Ident protocol with $\mathcal{T}_{\text{ID}}$ (step 4) and obtains the first protocol message m_1 from $\mathcal{R}$ (step 5). Then $\mathcal{A}_{\text{prv}}$ randomly chooses a protocol round $j_{\mathcal{R}}$ (step 6) and starts eavesdropping on the execution of the Ident protocol up to the point after $\mathcal{R}$ has sent protocol message $m_{2j_{\mathcal{R}}-1}$ (steps 7–11). Then $\mathcal{A}_{\text{prv}}$ gets the tag state $S^{\mathcal{T}_{\text{ID}}}$ by querying the Corrupt oracle, just before $\mathcal{T}_{\text{ID}}$ receives $m_{2j_{\mathcal{R}}-1}$ (step 12). Now $\mathcal{A}_{\text{prv}}$ chooses a random bit b (step 13) to decide how to complete the protocol execution. In case $b = 1$, $\mathcal{A}_{\text{prv}}$ continues by simulating *vtag* using the state $S^{\mathcal{T}_{\text{ID}}}$ obtained by the Corrupt query (steps 14–15). In case $b = 0$, $\mathcal{A}_{\text{prv}}$ sets $S^{\mathcal{T}_{\text{ID}}}$ to a new state generated on the fly (steps 16–19). Hereafter, $\mathcal{A}_{\text{prv}}$ simulates the tag by running the algorithm $\mathcal{T}^{(i)}$ with the state set according to the bit b until the protocol terminates (steps 21–28). Finally, $\mathcal{A}_{\text{prv}}$ outputs 1 if one of the following conditions hold: either $b = 1$ and $\mathcal{R}$ accepts $\mathcal{T}_{\text{ID}}$, whose transcript has partially been computed by $\mathcal{A}_{\text{prv}}$ with the real state (i.e., the output of Result is 1), or $b = 0$, and $\mathcal{R}$ rejected $\mathcal{T}_{\text{ID}}$ since a part of the transcript has been generated using a faked state (i.e., the output of Result is 0).

Recall that Definition 6 requires the existence of a blinder $\mathcal{B}_D$ such that: $\mathbf{Adv}^{\text{prv}}_{\mathcal{A}_{\text{prv}}} = \left|\Pr\left[\mathbf{Exp}^{\text{prv-0}}_{\mathcal{A}_{\text{prv}}} = 1\right] - \Pr\left[\mathbf{Exp}^{\text{prv-1}}_{\mathcal{A}_{\text{prv}}} = 1\right]\right| = \epsilon(l)$ for a negligible function ϵ. If such $\mathcal{B}_D$ exists, then $\mathcal{B}_D$ must be able to do the following: first, $\mathcal{B}_D$ simulates both $\mathcal{R}$ and $\mathcal{T}_{\text{ID}}$, then after $\mathcal{B}_D$ gets the state $S^{\mathcal{T}_{\text{ID}}}$ of $\mathcal{T}_{\text{ID}}$ from the Corrupt query, playing only at the reader side ($\mathcal{T}_{\text{ID}}$ is simulated by $\mathcal{A}_{\text{prv}}$ running the tag algorithm using either the real or a faked tag state), $\mathcal{B}_D$ can answer the Result query as $\mathcal{R}$ would do. Thus, $\mathcal{B}_D$ is able to recognize whether the messages received from $\mathcal{A}_{\text{prv}}$ (simulating $\mathcal{T}_{\text{ID}}$) are computed with the real state of $\mathcal{T}_{\text{ID}}$ or not. One can think of $\mathcal{B}_D$ as a two-phase algorithm. In the first phase $\mathcal{B}_D$ simulates the protocol execution between $\mathcal{R}$ and a tag *vtag*. Then, in the second phase, upon receiving the state $S^{\mathcal{T}_{\text{ID}}}$ of *vtag*, playing as the reader, $\mathcal{B}_D$ can distinguish if the tag messages received are computed according to the state of the tag simulated in first phase or not.

Now we show that if $\mathcal{B}_D$ exists, then $\mathcal{B}_D$ can be used to construct a narrow-forward adversary that distinguishes between any blinder $\mathcal{B}$ and the real oracles with non-negligible probability. Hence, the existence of $\mathcal{B}_D$ contradicts narrow-forward privacy and thus in turn destructive privacy. The idea of a narrow-forward adversary $\mathcal{A}^{\mathcal{B}_D}_{\text{prv}}$ is to run $\mathcal{B}_D$ as subroutine showing to $\mathcal{B}_D$ a view that is identical to the ones that it gets when playing with $\mathcal{A}_{\text{prv}}$ in Algorithm 7. The goal of $\mathcal{A}^{\mathcal{B}_D}_{\text{prv}}$ is to exploit the capabilities of $\mathcal{B}_D$ to distinguish whether the output of the SendTag oracle is generated by the real oracle using the real tag state or by a blinder $\mathcal{B}$ for narrow-forward privacy having no information on the real tag state. Formally, $\mathcal{A}^{\mathcal{B}_D}_{\text{prv}}$ is defined in Algorithm 8 and works as follows: first, $\mathcal{A}^{\mathcal{B}_D}_{\text{prv}}$ creates two legitimate tags $\mathcal{T}_{\text{ID}}$, ${\mathcal{T}_{\text{ID}}}'$ (steps 1–2) and makes one of them accessible as *vtag* (step 3). These three steps are also internally shown to $\mathcal{B}_D$. Then, $\mathcal{A}^{\mathcal{B}_D}_{\text{prv}}$ internally asks $\mathcal{B}_D$ to start a new instance π of the Ident protocol with *vtag* (step 4) and obtains the first protocol message m_1 generated by $\mathcal{B}_D$ (step 5). Then $\mathcal{A}^{\mathcal{B}_D}_{\text{prv}}$ randomly chooses a protocol round $j_{\mathcal{R}}$ (step 6) and makes $\mathcal{B}_D$ to simulate the first $j_{\mathcal{R}}$ rounds of the protocol, up to the point after $\mathcal{B}_D$ has sent the reader message $m_{2j_{\mathcal{R}}-1}$ (steps 7–11). Then, $\mathcal{A}^{\mathcal{B}_D}_{\text{prv}}$ queries the SendTag oracle with the message $m_{2j_{\mathcal{R}}-1}$ obtained by $\mathcal{B}_D$ (step 12). Next, $\mathcal{A}^{\mathcal{B}_D}_{\text{prv}}$ makes *vtag* inaccessible by querying the Free oracle (step 13) and makes accessible a randomly chosen tag *vtag*′ by querying the Draw oracle (step 14). Note that this step corresponds to the random selection of bit b in Algorithm 7. We stress that steps 12–15 are *not* shown to $\mathcal{B}_D$. Now $\mathcal{A}^{\mathcal{B}_D}_{\text{prv}}$ queries the Corrupt oracle and obtains the state $S^{\mathcal{T}_{\text{ID}}}$ of *vtag*′ (step 15). This query and $S^{\mathcal{T}_{\text{ID}}}$ are also shown to $\mathcal{B}_D$ (step 16). Then $\mathcal{A}^{\mathcal{B}_D}_{\text{prv}}$ sends to $\mathcal{B}_D$ the message obtained by the SendTag oracle in step 12, which has either been computed by the real SendTag oracle or the blinder $\mathcal{B}$ (step 17). Hereby, $\mathcal{B}_D$ expects to receive a message that has been computed according to the state $S^{\mathcal{T}_{\text{ID}}}$ obtained by Corrupt. Now the second phase starts, where $\mathcal{A}^{\mathcal{B}_D}_{\text{prv}}$ simulates the messages of *vtag*′ using $S^{\mathcal{T}_{\text{ID}}}$ and the messages sent by $\mathcal{B}_D$, which is playing as a reader (steps 18–24), until the protocol terminates, as expected by $\mathcal{B}_D$. Now, for the hypothesis, $\mathcal{B}_D$ can distinguish whether the messages it receives are (i) computed according to the state of the tag simulated in the first phase (thus $\Gamma[vtag] = \Gamma[vtag']$) and in this case Result will output

Alg. 8. Narrow-forward adversary $\mathcal{A}_{\text{prv}}^{\mathcal{B}_D}$

```
$\mathsf{CreateTag}(\mathtt{ID})$    ▷ shown to $\mathcal{B}_D$
$\mathsf{CreateTag}(\mathtt{ID}')$    ▷ shown to $\mathcal{B}_D$
$vtag \leftarrow \mathsf{Draw}(\Pr[\mathtt{ID}] = \frac{1}{2}, \Pr[\mathtt{ID}'] = \frac{1}{2})$    ▷ shown to $\mathcal{B}_D$
$\pi \leftarrow \mathsf{Launch}()$    ▷ simulated by $\mathcal{B}_D$
$m_1 \leftarrow \mathsf{SendReader}(-, \pi)$    ▷ simulated by $\mathcal{B}_D$
$j_{\mathcal{R}} \in_R \{1, \ldots, q_{\mathcal{R}}\}$
$i \leftarrow 1$
while $i < j_{\mathcal{R}}$ do $m_{2i} \leftarrow \mathsf{SendTag}(m_{2i-1}, vtag)$    ▷ simulated by $\mathcal{B}_D$
    $m_{2i+1} \leftarrow \mathsf{SendReader}(m_{2i}, \pi)$    ▷ simulated by $\mathcal{B}_D$
    $i \leftarrow i + 1$
end while
$m_{2j_{\mathcal{R}}} \leftarrow \mathsf{SendTag}(m_{2j_{\mathcal{R}}-1}, vtag)$    ▷ computed by $vtag$
$\mathsf{Free}\ (vtag)$
$vtag' \leftarrow \mathsf{Draw}(\Pr[\mathtt{ID}] = \frac{1}{2}, \Pr[\mathtt{ID}'] = \frac{1}{2})$
$S^{\mathcal{T}_{\mathtt{ID}}} \leftarrow \mathsf{Corrupt}(vtag')$
Show $S^{\mathcal{T}_{\mathtt{ID}}} \leftarrow \mathsf{Corrupt}(vtag)$ to $\mathcal{B}_D$
$m_{2j_{\mathcal{R}}+1} \leftarrow \mathsf{SendReader}(m_{2j_{\mathcal{R}}}, \pi)$    ▷ simulated by $\mathcal{B}_D$
$i \leftarrow j_{\mathcal{R}} + 1$
while $i < q_{\mathcal{R}}$ do
    if $i \leq q_{\mathcal{T}}$ then $m_{2i} \leftarrow \mathcal{T}_{\mathtt{ID}}{}^{(i)}(S^{\mathcal{T}_{\mathtt{ID}}}, m_{2i-1})$    ▷ computed by $\mathcal{A}_{\text{prv}}^{\mathcal{B}_D}$
    end if
    $m_{2i+1} \leftarrow \mathsf{SendReader}(m_{2i}, \pi)$    ▷ simulated by $\mathcal{B}_D$
    $i \leftarrow i + 1$
end while
$b \leftarrow \mathsf{Result}(\pi)$    ▷ simulated by $\mathcal{B}_D$
return $\big(\Gamma[vtag] = \Gamma[vtag'] \wedge b\big) \vee \big(\Gamma[vtag] \neq \Gamma[vtag'] \wedge \bar{b}\big)$
```

1, or (ii) with a different state (thus $\Gamma[vtag] \neq \Gamma[vtag']$) and in this case Result will output 0. Now we show that $\mathbf{Adv}^{\text{prv}}_{\mathcal{A}_{\text{prv}}^{\mathcal{B}_D}}$ is non-negligible if $\mathcal{B}_D$ exists.

First, consider the case where $\mathcal{A}_{\text{prv}}^{\mathcal{B}_D}$ interacts with real oracles. If $\Gamma[vtag] = \Gamma[vtag']$, then due to the existence of $\mathcal{B}_D$ we have that Result returns 1 with overwhelming probability, which makes $\mathcal{A}_{\text{prv}}^{\mathcal{B}_D}$ to return 1 with the same probability. Note that even though $\mathcal{B}_D$ learns the state $S^{\mathcal{T}_{\mathtt{ID}}}$ of *vtag* only after obtaining message $m_{2j_{\mathcal{R}}}$ that has been computed from this state, by the stateless property of the scheme and thus by Lemma 2, there is no noticeable difference between the state of *vtag* before and after the computation of $m_{2j_{\mathcal{R}}}$. In case $\Gamma[vtag] \neq \Gamma[vtag']$, we have that the first message $m_{2j_{\mathcal{R}}-1}$ received by $\mathcal{B}_D$ has been computed according to the state of *vtag* and all subsequent messages are computed according to the state of *vtag'*. This deviates from what $\mathcal{B}_D$ expects and thus $\mathcal{B}_D$ could erroneously answer the Result query with 1. Let us denote with p the probability that $\mathcal{B}_D$ with input $S^{\mathcal{T}_{\mathtt{ID}}}$ answers the Result query with 0 upon receiving a message computed with a random state followed by messages computed with $S^{\mathcal{T}_{\mathtt{ID}}}$. Then we have $\Pr\left[\mathbf{Exp}^{\text{prv-0}}_{\mathcal{A}_{\text{prv}}^{\mathcal{B}_D}} = 1\right] = \frac{1}{2} \cdot (1 - \epsilon(l)) + \frac{1}{2} \cdot p \leq \frac{(1+p)}{2}$.

Now, consider the case where $\mathcal{A}_{\text{prv}}$ interacts with $\mathcal{B}$. Here we have that in both cases ($\Gamma[vtag] = \Gamma[vtag']$ and $\Gamma[vtag] \neq \Gamma[vtag']$) the output of the $\mathsf{SendTag}$ oracle computed by $\mathcal{B}$ for the forward adversary is computed with a random state that with overwhelming probability is different from the state of *vtag* and *vtag'*. Thus, in both cases $\mathcal{B}_D$ with input the state $S^{\mathcal{T}_{\text{ID}}}$ receives the first message $m_{2j_{\mathcal{R}}-1}$ computed according to a state that is different from $S^{\mathcal{T}_{\text{ID}}}$. Hence we have $\Pr\left[\mathbf{Exp}^{\text{prv-0}}_{\mathcal{A}^{\mathcal{B}_D}_{\text{prv}}} = 1\right] = \frac{1}{2} \cdot (1-p) + \frac{1}{2} \cdot p = \frac{(1-p)}{2} + \frac{p}{2} = \frac{1}{2}$. and it follows that $\mathbf{Adv}^{\text{prv}}_{\mathcal{A}^{\mathcal{B}_D}_{\text{prv}}} \leq \left|\frac{(1+p)}{2} - \frac{1}{2}\right| = \frac{p}{2}$. Note that if p is non-negligible, so is the advantage of $\mathcal{A}^{\mathcal{B}_D}_{\text{prv}}$ and the proof is finished. If instead p is negligible, then $\mathcal{B}_D$ has non-negligible probability of answering 1 to a Result query when no message originates from a valid state. (In the above experiment, this case happens when $j_{\mathcal{R}}$ corresponds to the last round of the protocol.) Obviously a reader that always expects messages being computed according to a legitimate state would output 0 to a Result query in such an experiment, and this would contradict the fact that (even a variation of) $\mathcal{B}_D$ is successful against this variation of $\mathcal{A}_{\text{prv}}$.

The last issue to address is the more general case where a reader admits wrong messages from a tag, still responding with 1 to a Result query when some messages are computed using a legitimate state. However, since the procedure of the reader is public, the above proof can be generalized to any reader strategy. Indeed, $\mathcal{A}_{\text{prv}}$ must replace some correctly computed messages with messages computed with a random state such that the replacement of the valid messages exposes the failure of $\mathcal{B}_D$. This is achieved by asking $\mathcal{A}_{\text{prv}}$ to compute each tag-side message either using a legitimate or an illegitimate tag state with probability q that comes from the description of the reader procedure for the Result query, so that the output of this query is noticeably perturbed by the replacement of a correctly computed message by a wrongly computed one. □

7 Conclusion

In this paper, we revisited the security and privacy model for RFID systems proposed by Paise and Vaudenay (PV-Model) [30]. This model is very interesting since it covers many aspects of previous works and proposes a unified RFID security and privacy framework. We showed several impossibility results that show that the formalization given in the PV-Model is too restrictive and fails in modelling real-life scenarios, where interesting privacy notions and reader authentication are intuitively achievable. A partial and shorter version of this work appeared in [1].

Acknowledgments. We thank Paolo D'Arco for several useful discussions about RFID privacy notions. This work has been supported in part by the European Commission through the FP7 programme under contract 216676 ECRYPT II, 238811 UNIQUE, and 215270 FRONTS, in part by the Ateneo Italo-Tedesco under Program Vigoni and by the MIUR Project PRIN 2008 "PEPPER: Privacy E Protezione di dati PERsonali" (prot. 2008SY2PH4).

References

1. Armknecht, F., Sadeghi, A.R., Visconti, I., Wachsmann, C.: On RFID privacy with mutual authentication and tag corruption. In: Zhou, J. (ed.) ACNS 2010. LNCS, vol. 6123, pp. 493–510. Springer, Heidelberg (2010)
2. Atmel Corporation: Innovative IDIC solutions (2007), http://www.atmel.com/dyn/resources/prod_documents/doc4602.pdf
3. Avoine, G.: Adversarial model for radio frequency identification. ePrint, Report 2005/049 (2005)
4. Avoine, G., Lauradoux, C., Martin, T.: When compromised readers meet RFID. In: The 5th Workshop on RFID Security (RFIDSec) (2009)
5. Bellare, M., Fischlin, M., Goldwasser, S., Micali, S.: Identification protocols secure against reset attacks. In: Pfitzmann, B. (ed.) EUROCRYPT 2001. LNCS, vol. 2045, pp. 495–511. Springer, Heidelberg (2001)
6. Blundo, C., Persiano, G., Sadeghi, A.R., Visconti, I.: Improved security notions and protocols for non-transferable identification. In: Jajodia, S., Lopez, J. (eds.) ESORICS 2008. LNCS, vol. 5283, pp. 364–378. Springer, Heidelberg (2008)
7. Bringer, J., Chabanne, H., Icart, T.: Efficient zero-knowledge identification schemes which respect privacy. In: Proceedings of ASIACCS 2009, pp. 195–205. ACM Press, New York (2009)
8. Burmester, M., van Le, T., de Medeiros, B.: Universally composable and forward-secure RFID authentication and authenticated key exchange. In: Proc. of ASIACCS, pp. 242–252. ACM Press, New York (2007)
9. Canetti, R., Goldreich, O., Goldwasser, S., Micali, S.: Resettable zero-knowledge (extended abstract). In: STOC, pp. 235–244 (2000)
10. D'Arco, P., Scafuro, A., Visconti, I.: Revisiting DoS Attacks and Privacy in RFID-Enabled Networks. In: Dolev, S. (ed.) ALGOSENSORS 2009. LNCS, vol. 5804, pp. 76–87. Springer, Heidelberg (2009)
11. D'Arco, P., Scafuro, A., Visconti, I.: Semi-destructive privacy in DoS-enabled RFID systems. In: The 5th Workshop on RFID Security (RFIDSec) (2009)
12. Deng, R.H., Li, Y., Yao, A.C., Yung, M., Zhao, Y.: A new framework for RFID privacy. ePrint, Report 2010/059 (2010)
13. Deng, Y., Lin, D.: Instance-dependent verifiable random functions and their application to simultaneous resettability. In: Naor, M. (ed.) EUROCRYPT 2007. LNCS, vol. 4515, pp. 148–168. Springer, Heidelberg (2007)
14. EPCglobal Inc.: (April 2008), http://www.epcglobalinc.org/
15. Garcia, F., de Koning Gans, G., Muijrers, R., van Rossum, P., Verdult, R., Wichers Schreur, R., Jacobs, B.: Dismantling MIFARE Classic. In: Jajodia, S., Lopez, J. (eds.) ESORICS 2008. LNCS, vol. 5283, pp. 97–114. Springer, Heidelberg (2008)
16. Garcia, F.D., van Rossum, P.: Modeling privacy for off-line RFID systems. In: The 5th Workshop on RFID Security (RFIDSec) (2009)
17. Goyal, V., Sahai, A.: Resettably secure computation. In: EUROCRYPT, pp. 54–71 (2009)
18. Hutter, M., Schmidt, J.M., Plos, T.: RFID and its vulnerability to faults. In: Oswald, E., Rohatgi, P. (eds.) CHES 2008. LNCS, vol. 5154, pp. 363–379. Springer, Heidelberg (2008)
19. I.C.A. Organization: Machine Readable Travel Documents, Doc 9303, Part 1 Machine Readable Passports, 5th edn (2003)
20. Juels, A.: RFID security and privacy: A research survey. Journal of Selected Areas in Communication 24(2), 381–395 (2006)

21. Juels, A., Weis, S.A.: Defining strong privacy for RFID. ePrint, Report 2006/137 (2006)
22. Kasper, T., Oswald, D., Paar, C.: New methods for cost-effective side-channel attacks on cryptographic RFIDs. In: The 5th Workshop on RFID Security (RFIDSec) (2009)
23. Kirschenbaum, I., Wool, A.: How to build a low-cost, extended-range RFID skimmer. ePrint, Report 2006/054 (2006)
24. Mangard, S., Oswald, E., Popp, T.: Power Analysis Attacks Revealing the Secrets of Smart Cards. Springer, Heidelberg (2007)
25. Ng, C.Y., Susilo, W., Mu, Y., Safavi-Naini, R.: New privacy results on synchronized RFID authentication protocols against tag tracing. In: Backes, M., Ning, P. (eds.) ESORICS 2009. LNCS, vol. 5789, pp. 321–336. Springer, Heidelberg (2009)
26. 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. 251–256. Springer, Heidelberg (2008)
27. Nithyanand, R., Tsudik, G., Uzun, E.: Readers behaving badly: Reader revocation in PKI-based RFID systems. ePrint, Report 2009/465 (2009)
28. NXP Semiconductors: MIFARE (May 2007), `http://mifare.net/`
29. NXP Semiconductors: MIFARE smartcard ICs (April 2010), `http://www.mifare.net/products/smartcardics/`
30. Paise, R.I., Vaudenay, S.: Mutual authentication in RFID: Security and privacy. In: Proc. of ASIACCS, pp. 292–299. ACM Press, New York (2008)
31. Sadeghi, A.R., Visconti, I., Wachsmann, C.: User privacy in transport systems based on RFID e-tickets. In: International Workshop on Privacy in Location-Based Applications (PiLBA) (2008)
32. Sadeghi, A.R., Visconti, I., Wachsmann, C.: Anonymizer-enabled security and privacy for RFID. In: Garay, J.A., Miyaji, A., Otsuka, A. (eds.) CANS 2009. LNCS, vol. 5888, pp. 134–153. Springer, Heidelberg (2009)
33. Sadeghi, A.R., Visconti, I., Wachsmann, C.: Efficient RFID security and privacy with anonymizers. In: The 5th Workshop on RFID Security (RFIDSec) (2009)
34. Sadeghi, A.R., Visconti, I., Wachsmann, C.: Location privacy in RFID applications. In: Bettini, C., Jajodia, S., Samarati, P., Wang, X.S. (eds.) Privacy in Location-Based Applications. LNCS, vol. 5599, pp. 127–150. Springer, Heidelberg (2009)
35. Sadeghi, A.R., Visconti, I., Wachsmann, C.: Enhancing RFID Security and Privacy by Physically Unclonable Functions. Springer, Heidelberg (2010)
36. Sadeghi, A.R., Visconti, I., Wachsmann, C.: PUF-enhanced RFID security and privacy. In: Workshop on Secure Component and System Identification (SECSI) (2010)
37. Vaudenay, S.: On privacy models for RFID. In: Kurosawa, K. (ed.) ASIACRYPT 2007. LNCS, vol. 4833, pp. 68–87. Springer, Heidelberg (2007)
38. Weis, S.A., Sarma, S.E., Rivest, R.L., Engels, D.W.: Security and privacy aspects of low-cost radio frequency identification systems. In: Hutter, D., Müller, G., Stephan, W., Ullmann, M. (eds.) Security in Pervasive Computing. LNCS, vol. 2802, pp. 50–59. Springer, Heidelberg (2004)