Adversarial Model for Radio Frequency Identification

Gildas Avoine

EPFL

Lausanne, Switzerland

(20th February 2005)

Abstract. Radio Frequency Identification (RFID) systems aim to identify objects in open environments with neither physical nor visual contact. They consist of transponders inserted into objects, of readers, and usually of a database which contains information about the objects. The key point is that authorised readers must be able to identify tags without an adversary being able to trace them. Traceability is often underestimated by advocates of the technology and sometimes exaggerated by its detractors. Whatever the true picture, this problem is a reality when it blocks the deployment of this technology and some companies, faced with being boycotted, have already abandoned its use. Using cryptographic primitives to thwart the traceability issues is an approach which has been explored for several years. However, the research carried out up to now has not provided satisfactory results as no universal formalism has been defined. In this paper, we propose an adversarial model suitable for RFID environments. We define the notions of existential and universal untraceability and we model the access to the communication channels from a set of oracles. We show that our formalisation fits the problem being considered and allows a formal analysis of the protocols in terms of traceability. We use our model on several well-known RFID protocols and we show that most of them have weaknesses and are vulnerable to traceability.

Key words: RFID, Adversarial Model, Privacy, Untraceability, Cryptanalysis.

1 Introduction

1.1 RFID Motivation

Often presented as a new technological revolution, Radio Frequency Identification (RFID) makes the identification of objects in open environments possible, with neither physical nor visual contact. They are made up of transponders inserted into the objects, of readers which communicate with the transponders using radio frequencies and usually of a database which contains information on the tagged objects.

This technology is not fundamentally new. It has existed for several decades and has been used in the public domain for several years, for example in ticketing on public transport or ski-lifts, on motorway tollgates, or even for animal identification. RFID technology is thus found on a whole range of applications which have very different purposes and therefore different needs. The boom which RFID technology is enjoying today rests essentially on the willingness to develop low-cost transponders (for around of 5 US cents) thus rendering them disposable. Such transponders are called *tags*. Advocates of this technology say that they are the super barcodes of the future. Indeed, identification by radio frequency represents a major innovation in relation to optical identification. In addition to the minute size of the tags which allows them to be implanted within objects, it allows objects to be read en masse, without the need for visual contact, and each tag has a unique identifier representing a single object, unlike barcodes.

One area of application for RFID tags is the management of stock and inventories in shops and warehouses. The American mass-marketing giant, Wal-Mart, has recently placed a requirement on its main suppliers that they use electronic tags on the palettes and cartons that are delivered to its stores. The advantages of using RFID tags can also be seen, for example, in libraries where putting an electronic tag in each book simplifies the borrowing and returning procedures and facilitates the staff's job. Several libraries in the United States have already adopted the RFID technology, e.g., the Santa Clara City Library in California, the University of Nevada, the Las Vegas Library, and the Eugene Oregon Public Library [20]. Among the actual applications, we can also cite locating people in a public area, e.g., amusement parks [22]. The aim is to help customers to keep in touch with other members of their group in the park.

1.2 RFID Primer

RFID tags are electronic microcircuits equipped with an antenna. The least expensive ones have only extremely limited computation, storage, and communication capacities, because of the cost and size restrictions dictated by the targeted applications. Capabilities of the tags ensue from the ISO standards [15] and the EPC Global Inc. standards [8].

Tags have no microprocessors and are equipped with only a few thousand logic gates at the very most, which makes it a real challenge to integrate encryption or signature algorithms into these devices. This difficulty is reinforced by the fact that the tags are passive, meaning that they do not have an internal power source: they use the power supplied by the reader. Fortunately, promising research is being done at the moment, notably the implementation of AES encryption for RFID tags proposed by Feldhofer, Dominikus and Wolkerstorfer [10]. Note that such an implementation cannot fit a very low-cost tag, but it may be suited to reasonably inexpensive tags.

The storage capacities of RFID tags are also extremely limited. The cheapest devices have between 64 and 128 bits of ROM only, which allows the unique identifier of the tag to be stored, but adding EEPROM remains an option for more developed applications. Contrary to smartcards made for secure applications (credit cards, pay TV, etc.), the tags are not tamper-resistant. This fact does not mean that all security measures are impossible. Indeed, we have to consider the cost of the attack in relation to its gain. For example, the ease of reading the content of a tag may be counter-balanced by the difficulty of getting access to it. Subcutaneous tags are a good illustration of this difficulty of access. A less extreme example is the use of tags in bracelets to locate people in enclosed spaces: the tag could be initialised when it is given out to a customer and the data could be erased when the customer gives the bracelet back. Nevertheless, it would not be secure for all the tags to contain the same secret, as the cost of the attack could become negligible when compared with the gain.

The communication distance between tags and readers depends on numerous parameters, in particular the communication frequency. Two main categories of RFID systems coexist: the systems using the frequency 13.56MHz and the systems using the frequency 860-960MHz, for which the communication range is greater. In this latter case, the information sent by the reader can be received in practice up to a hundred meters, but the information returned from the tag to the reader reaches a few meters at most. These limits, resulting from the standards and regulations in place do not mean that the tags cannot be read from a greater distance: an attacker could exceed these limits, for example by transgressing the laws relating to the maximum power.

1.3 RFID Security Issues

Security problems in RFID systems can be put into two categories. The first concerns those attacks which aim to wipe out the functioning of the system, e.g., denial of service attacks. The second category, the one which interests us, is related to privacy: the problem is information leakage, as a tag may reveal data about an object which contains it (for example the title or author of the book) and also traceability. Information leakage can be avoided if the tag only transmits one identifier which can only be used by those persons having access to the system's database. However, this does not prevent traceability. The traceability of tags, and by extension of people, is a difficulty that RFID technology must surmount if it is to be widely used. For example, companies like Gillette and Benetton have been the victims of virulent boycott campaigns [25].

Beyond hardware-based techniques [11,19], many researchers have looked into the problem in order to design protocols which allow authorised persons to identify the tags without an adversary being able to trace them. Among them, the principal players are Avoine [1,2,3], Feldhofer [9,10], Juels [12,16,17,18,19], Molnar and Wagner [13,20], Ohkubo [21], and Weis [24,27,26,28].

Most schemes are 3-round protocols (or can be reduced to this type of protocol) as described in Fig. 1. The first message may be a purely starting signal or may contain data, e.g., a nonce. The principle of the schemes rests on the fact that the information contained in the second message changes at each new identification. This information could be either the tag's identifier (which is then renewed each time the protocol is executed), or an encrypted version of this identifier (which is then static in the tag, but encrypted with a probabilistic algorithm). Whichever the solution, exchanged information is refreshed at each identification, according to a procedure which differentiates the existing protocols. Indeed, either the tag is capable of refreshing this information itself or it needs help from the reader. In the first case, the third message is not used for the identification of the tag but it may be used to ensure authentication of the reader (e.g., [10,20]). In the latter case, the third message contains data which are used by the tag in the "refreshment" process. Obviously, the less computations carried out by the tag, the less the cost of the tag is, but ensuring privacy without using any cryptographic functions would be a pipedream, as shown in [2].

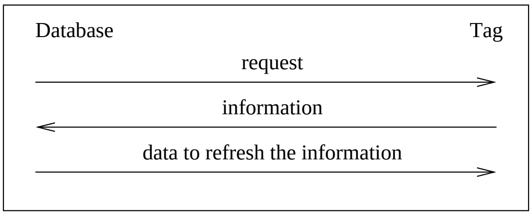

Fig. 1. Identification protocol

Relatively few people have done work which proves the security or exhibits the weaknesses of the proposed RFID protocols. Actually, designing and analysing RFID protocols is still a real challenge because no universal model has been defined: up until now designs and attacks have been made in a pedestrian way. In this paper we put forward just such a formalism for traceability suited to RFID protocols. We thus define in Sect. 2 the notions of existential and universal untraceability and we model access to the communication channels from a set of oracles. We show in Sect. 3 that our formalism fits RFID and allows a formal analysis of the protocols. We use our model in order to analyse several existing protocols and show that in a realistic model, many protocols are not resistant to traceability.

2 Adversarial Model

2.1 Modelling the System

An RFID system is made up of entities (database, readers, and tags) as well as communication channels. Looking at the security of information contained by the database and the readers has little relevance here as these devices do not have particular restrictions and can therefore make use of appropriate cryptographic techniques. In practice, either the readers do not store sensitive information (in which case they can be compared to simple physical devices of no interest from a security point of view), or they store sensitive information (in which case they can benefit from being protected by adequate measures). For these same reasons, studying the communication channel between the database and the readers is not relevant. Hence, readers and database are often considered as a single and unique entity in the security analysis.

The sources of information which can benefit an adversary are therefore limited to the communication channel between the reader and the tag as well as the contents of the memory of the tag. As we have seen in Sect. 1.2, the communication distance from a reader to a tag (*forward channel*) is generally longer than the distance from a tag to a reader (*backward channel*). It is worthwhile studying these two channels separately since certain protocols can benefit from this asymmetry. Finally, the memory of the tag can be represented as another channel, called the *memory channel*. These channels are represented in Fig. 2.

From a theoretical point of view, each of these channels can be accessed by reading and/or in writing, or not accessible at all. In practice, some of these combinations are more realistic than others and we disregard some of the less likely combinations, e.g., we do not consider writing access to the memory channel although fault injection attacks could be relevant. Moreover, we consider that an adversary will only be able to read the memory channel once. One may argue that, even if such an attack is destructive, an adversary could create a clone of the tag using the data she discovered. However, we have no way to prevent an attack where the adversary can access the memory channel as many times she wants. As we will see below, limiting the access to the memory channel is strongly related to the notion of *forward* untraceability. Finally, we consider that the contents of the tags are independent. With the aim

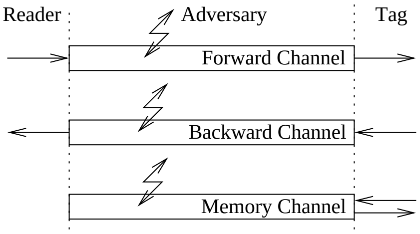

Fig. 2. Information channels of an RFID system

of obtaining a realistic and applicable model, we are limiting the adversary's means to the oracles defined in Sect. 2.2. Goals of the adversary will be defined in Sect. 2.3.

2.2 Means of the Adversary

Nowadays, the formalisation of the adversarial model is required in every security proof. Such a model consists of the means of the adversary and its goals. For example, in confidentiality, it is common practice to look at the *chosen plaintext attacks* (CPA), the *non-adaptive chosen ciphertext attacks* (CCA1), and the *adaptive chosen ciphertext attacks* (CCA2). In terms of signatures, we consider mainly the *known-message attacks* (KMA) and the *adaptive chosen message attacks* (CMA). We define below the means of an adversary $\mathcal{A}$ in an RFID system. These means are represented using oracles. We denote a *tag* T and a *reader* R that can participate in the RFID protocol P. Each of them can run several instances of P. We denote tag instances by π_T^i and reader instances by π_R^j.

- $\mathsf{Query}(\pi_T^i, m_1, m_3)$: this query models $\mathcal{A}$ sending a request m_1 to T through the forward channel and subsequently sending it the message m_3 after having received its answer.
- $\mathsf{Send}(\pi_R^j, m_2)$: this query models $\mathcal{A}$ sending the message m_2 to R through the backward channel and receiving its answer.
- $\mathsf{Execute}(\pi_T^i, \pi_R^j)$: this query models $\mathcal{A}$ executing an instance of P between T and R, obtaining so the messages exchanged on both the forward and the backward channels.
- $\mathsf{Execute}^*(\pi_T^i, \pi_R^j)$: this query models $\mathcal{A}$ executing an instance of P between T and R, but obtaining the messages exchanged on the forward channel only.
- $\mathsf{Reveal}(\pi_T^i)$: this query models $\mathcal{A}$ obtaining the content of T's memory channel. This query can be used only once such that Query, Send, Execute, and Execute* can no longer be used after.

We will say that a protocol is resistant to attacks A-$\mathcal{O}$ or that it is A-$\mathcal{O}$ if it is resistant to an attack A when the adversary has access to the oracles of $\mathcal{O} \subset \{\mathsf{Q}, \mathsf{S}, \mathsf{E}, \mathsf{E}^*, \mathsf{R}\}$ where Q, S, E, E* and R represent respectively the oracles Query, Send, Execute, Execute* and Reveal. For ease of legibility, we will simplify the notation by writing down for example A-QSE instead of A-{Q,S,E}. We will write down $\omega_i(T)$ the result of the application of an oracle Q, E, E*, or R on a tag T. We therefore have $\omega_i(T) \in \{\mathsf{Query}(\pi_T^i, *), \mathsf{Execute}(\pi_T^i, *), \mathsf{Execute}^*(\pi_T^i, *), \mathsf{Reveal}(\pi_T^i)\}$.

2.3 Goals of the Adversary

The security proof of a protocol equally rests on the formalisation of the aims of an adversary. Thus we require that a public key encryption scheme verifies, for example, the properties of *indistinguishability* (IND) or of *non-malleability* (NM), or that a signature scheme is resistant to *forgery* or to *total break*. In the framework of identification by radio frequency, we introduce the notion of *untraceability* (UNT). Untraceability is characterised by two fundamental points.

- By the very physical nature of the tags, an adversary who stays in contact with the tag is clearly capable of tracing it. This physical tracing cannot be thwarted. While the adversary is physically tracing a tag, she is in a position to determine which executions of the protocol are linked to the tag. We thus define an *interaction* as a set of executions on the same tag at a time when the adversary is in a position to physically identify it. An interaction is more precisely defined by $\Omega_I(T) = \{\omega_i(T) \mid i \in I\} \cup \{\mathsf{Send}(\pi_*^i, *) \mid i \in J\}$ where $J \subset \mathbb{N}$. By definition, the length of an interaction $\Omega_I(T)$ is $\mid I \mid$. We will suppose below that I is a sub-interval of $\mathbb{N}$.

- When an adversary is in a position to trace a tag, she can do it in a temporary way (e.g., as long as an honest reader has not interrogated it) or in a definitive way. These cases will lead to the notions of *existential* and *universal* untraceability. By way of a comparison, security of signature schemes consider *universal*, *random*, *selective*, or *existential* forgery.

Broadly speaking, after having interacted with a target T and possibly some readers and thus obtaining an interaction $\Omega_I(T)$, whose length is less than a given parameter ℓ_{ref}, an adversary $\mathcal{A}$ needs to find her target among two tags T_1 and T_2 which are presented to her. For that, she can query both T_1 and T_2, thus obtaining two interactions $\Omega_{I_1}(T_1)$ and $\Omega_{I_2}(T_2)$ whose lengths are less than a given length ℓ_{chal}. What differentiates existential and universal is the manner in which I_1 and I_2 are fixed. If there exist I_1 and I_2 such that $\mathcal{A}$ is able to succeed then we talk of *existential traceability.* If she is able to win for all I_1 and I_2, then we talk of *universal traceability.*

We consider below *Oracle* which, being given T and I sends back $\hat{\Omega}_I(T)$. *Oracle* allows us to simulate the set of oracles to which the adversary has access. Here, *Oracle* will call the oracles of $\mathcal{O} \subset \{\mathsf{Q}, \mathsf{S}, \mathsf{E}, \mathsf{E}^*, \mathsf{R}\}$ according to model chosen. The interaction $\hat{\Omega}_i(T)$ is the interaction which maximises the adversary's advantage. We also look at a *Challenger* which supplies two tags to the adversary, one of which is the target tag. We define below the notion of untraceability.

Existential Untraceability

Parameters: ℓ_{ref}, ℓ_{chal}, $\mathcal{O}$.

1. $\mathcal{A}$ requests the *Challenger* thus receiving her target T.
2. $\mathcal{A}$ chooses I and calls *Oracle*$(T, I, \mathcal{O})$ where $|I| \leq \ell_{\text{ref}}$ then receives $\hat{\Omega}_I(T)$.
3. $\mathcal{A}$ requests the *Challenger* thus receiving her challenge T_1 and T_2.
4. $\mathcal{A}$ chooses I_1 and I_2 such that $|I_1| \leq \ell_{\text{chal}}$, $|I_2| \leq \ell_{\text{chal}}$, and $(I_1 \cup I_2) \cap I = \varnothing$.
5. $\mathcal{A}$ calls *Oracle*$(T_1, I_1, \mathcal{O})$ and *Oracle*$(T_2, I_2, \mathcal{O})$, then receives $\hat{\Omega}_{I_1}(T_1)$ and $\hat{\Omega}_{I_2}(T_2)$.
6. $\mathcal{A}$ decides which of T_1 or T_2 is T, then outputs her guess T'.

Universal Untraceability

Parameters: ℓ_{ref}, ℓ_{chal}, $\mathcal{O}$.

1. $\mathcal{A}$ requests the *Challenger* thus receiving her target T.
2. $\mathcal{A}$ chooses I and calls *Oracle*$(T, I, \mathcal{O})$ where $|I| \leq \ell_{\text{ref}}$ then receives $\hat{\Omega}_I(T)$.
3. $\mathcal{A}$ requests the *Challenger* thus receiving her challenge T_1, T_2, I_1, and I_2.
4. $\mathcal{A}$ calls *Oracle*$(T_1, I_1, \mathcal{O})$ and *Oracle*$(T_2, I_2, \mathcal{O})$, then receives $\hat{\Omega}_{I_1}(T_1)$ and $\hat{\Omega}_{I_2}(T_2)$.
5. $\mathcal{A}$ decides which of T_1 or T_2 is T, then outputs her guess T'.

It is usually useful to restrict the choice of I_1 and I_2 made by the adversary (existential) or by the challenger (universal) such that $I < I_1, I_2$ (resp. $I > I_1, I_2$). We denote then $\mathsf{Existential}^+$ (resp. $\mathsf{Existential}^-$) and $\mathsf{Universal}^+$ (resp. $\mathsf{Universal}^-$). The notion of $\mathsf{Universal}^-$ is particularly relevant when the oracle R is used, and meets the notion of *forward* privacy defined in [3]. We will consequently refer to this notion as Forward-UNT. For each of these variants, we define the advantage of $\mathcal{A}$ for a given protocol P by:

$$\mathsf{Adv}_P^{\mathsf{UNT}}(\mathcal{A}) = 2\Pr(T' = T) - 1$$

where the probability space is over all the random tags. If $\mathcal{A}$'s advantage is negligible with the parameters ℓ_{ref}, ℓ_{chal}, and $\mathcal{O}$, P is said to be $\mathsf{UNT}_{\ell_{\text{ref}},\ell_{\text{chal}}}$-$\mathcal{O}$ secure, usually simply denoted by UNT-$\mathcal{O}$.

2.4 Implications and Separations

One can mix and match the goals {Existential-UNT, Forward-UNT, Universal-UNT} of the adversary and her means $\mathcal{O} \subset \{\mathsf{Q}, \mathsf{S}, \mathsf{E}, \mathsf{E}^*, \mathsf{R}\}$. From [4], we give the following relations, respectively called *implication* and *separation*:

$A \rightarrow B$: a proof that if an RFID protocol P meets the notion of security A then P also meets notion of security B.

$A \not\rightarrow B$: A construction of an RFID protocol P that provably meets notion of security A but provably not meets the notion of security B.

Definitions supplied in Sect. 2.3 lead us clearly to the relations:

$$\textsf{Existential-UNT} \quad \rightarrow \quad \textsf{Forward-UNT} \quad \rightarrow \quad \textsf{Universal-UNT}$$

Given the definitions of Existential-UNT, Forward-UNT, and Universal-UNT, proofs are straightforward. We now consider the relations between the means of the adversary. By definition, we have $\textsf{UNT-E} \rightarrow \textsf{UNT-E}^*$ but $\textsf{UNT-E}^* \not\rightarrow \textsf{UNT-E}$. Moreover,

$$\forall \textsf{A}, \textsf{B} \in \{\textsf{Q}, \textsf{S}, \textsf{E}, \textsf{R}\}, \textsf{UNT-A} \not\rightarrow \textsf{UNT-B}.$$

We have however $\textsf{QS} \rightarrow \textsf{E}$ and $\textsf{E} \not\rightarrow \textsf{QS}$. The implication comes from the fact that an adversary having access to the Q and S oracles can simulate E using a man-in-the-middle attack. The separation comes from the fact that the adversary is passive when using the E oracle and therefore cannot modify the messages, contrary to Q and S. Another important implication is:

$$(\forall \mathcal{O}, \mathcal{O}' \subset \{\textsf{Q}, \textsf{S}, \textsf{E}, \textsf{E}^*, \textsf{R}\}, \mathcal{O}' \subset \mathcal{O}) \Longrightarrow (\textsf{UNT-}\mathcal{O} \rightarrow \textsf{UNT-}\mathcal{O}').$$

Indeed, if the adversary is not able to track a tag with the set of oracles $\mathcal{O}$, she cannot succeed with a smaller set of oracles. In practice, certain combinations are more relevant than others. Thus, we will only focus on UNT-E, UNT-Q, UNT-QSE, and UNT-QSER. From the above results, we have:

$$\textsf{UNT-QSER} \rightarrow \textsf{UNT-QSE} \rightarrow \left| \begin{array}{l} \textsf{UNT-E} \\ \textsf{UNT-Q} \end{array} \right.$$

It is clear that a protocol should be both UNT-Q and UNT-E, meaning that an adversary should not be capable of tracking a tag only by querying it or only by eavesdropping the channels. In practice, a protocol must be Existential-UNT-QSE and Forward-UNT-QSER. So the adversary is never capable of tracking a tag when she can interact with both the target tag and the readers, and eavesdrop executions between the tag and readers. Moreover, obtaining the content of the tag by tampering with it does not allow the adversary to track it in the past (e.g., by analysing the reader logs). In some specific applications, Existential-UNT-QSE is enough if the adversary is not able to physically tamper with its target as we saw in Sect. 1.2. Note that designing a Existential-UNT-QSER does not make sense because if the adversary is able to obtain the content of the tag, obtaining so as information as the tag itself, she will be able to trace it in the future, at least during the identification just following the attack.

In the following section, we will present several well-known RFID protocols. We will show that most of them suffer from some weaknesses and do not fulfill the security requirements.

3 Attacks on Existing Protocols

3.1 Protocol of Golle, Jakobsson, Juels and Syverson

The protocol of Golle *et al.* [12] relies on the concept of universal re-encryption, i.e., a scheme where re-encryptions of a message m are performed neither requiring nor yielding knowledge of the public key under which m has been encrypted initially. The scheme consists of encrypting a plaintext m by appending two ciphertexts: the first one is the ElGamal encryption of m while the second one is the ElGamal encryption of the *neutral element* of $\mathcal{G}$, where $\mathcal{G}$ is the underlying group for the cryptosystem. We detail the scheme here. Let E be the ElGamal encryption scheme, and U be the corresponding re-encryption scheme, we have $U(m) := [E(m); E(1_{\mathcal{G}})]$. Let q be the order of $\mathcal{G}$, and g a generator. The universal re-encryption scheme is defined by the following four algorithms:

- *Key generation:* output the private key $x \in \mathbb{Z}$ and the public key $y = g^x$.
- *Encryption:* let (r_0, r_1) be a random element picked in $\mathbb{Z}_q^2$. The encrypted value of a message m is $U(m) = [(\alpha_0, \beta_0); (\alpha_1, \beta_1)] = [(my^{r_0}, g^{r_0}); (y^{r_1}, g^{r_1})]$.
- *Decryption:* given the ciphertext $[(\alpha_0, \beta_0); (\alpha_1, \beta_1)]$, if α_0,β_0,α_1,$\beta_1 \in \mathcal{G}$ and $\alpha_1/\beta_1^x = 1$, then the plaintext is α_0/β_0^x.

- *Re-encryption:* let (r'_0, r'_1) be a random element picked in $\mathbb{Z}_q^2$. The re-encrypted value of a ciphertext $[(\alpha_0, \beta_0); (\alpha_1, \beta_1)]$ is $[(\alpha_0 \alpha_1^{r'_0}, \beta_0 \beta_1^{r'_0}); (\alpha_1^{r'_1}, \beta_1^{r'_1})]$.

We now describe the RFID protocol suggested by Golle *et al.*, based on their universal re-encryption scheme. During the initialisation of the tag, an encrypted identifier is stored in the tag. On the other hand, this encrypted identifier as well as the secret key corresponding to the tag is stored in the database. An execution is carried out as follows: (1) The reader sends a request to the tag; (2) The tag sends back its encrypted identifier; (3) The reader re-encrypts the identifier of the tag using the universal re-encryption scheme described above and sends the new value to the tag (Fig. 3).

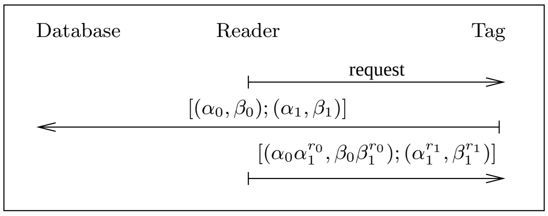

Fig. 3. Protocol of Golle, Jakobsson, Juels, and Syverson

As noted in [12], if an attacker sends a fake re-encrypted identifier to the tag, the database will not be able to identify the tag in the future. According to [12], this attack does not allow the tag to be traced, at the most it will harm the normal functioning of the system. The authors do, however, reveal an exception: when an adversary replaces the value (α_1, β_1) by $(1_\mathcal{G}, 1_\mathcal{G})$ where $1_\mathcal{G}$ represents the neutral element of $\mathcal{G}$, the future re-encryptions will no longer change the identifier. The tag can protect itself from this attack by verifying that (α_1, β_1) is not equal to $(1_\mathcal{G}, 1_\mathcal{G})$ before changing its value. However, the Golle *et al.*'s protocol suffers also from other weaknesses which we describe below.

Attack based on eavesdropping. The first thing to see is that the protocol does not resist to simple eavesdropping attacks. Indeed, since the tag sends in the second message what it received in the third message of the previous execution, an attacker is able to track the tag by eavesdropping; in other words, the protocol is not Existential-UNT-E.

Attack based on invariants. The weakness described here results from the fact that the ciphertext sent by the tag is not random. Taken independently, every element of the ciphertext $[(\alpha_0, \beta_0); (\alpha_1, \beta_1)]$ follows a uniform distribution assuming that the discrete logarithm is a random function, but these elements are not independent. Thus, if the attacker is able to choose $\alpha_0, \beta_0, \alpha_1, \beta_1$ verifying a relation invariant by re-encryption, i.e., a relation which remains verified after re-encryptions, then she will (almost certainly) be able to trace the tag. Let us take the relation $\mathcal{R}$ such that $(\alpha_0, \beta_0, \alpha_1, \beta_1)$ verifies $\mathcal{R}$ if, and only if $\alpha_1 = \beta_1$. Let us denote $\alpha_0^{(k)}$, $\beta_0^{(k)}$, $\alpha_1^{(k)}$ and $\beta_1^{(k)}$ the values contained in the tag after the k-th re-encryption. $\mathcal{R}$ is invariant by re-encryption: if $(\alpha_0, \beta_0, \alpha_1, \beta_1)$ verifies $\mathcal{R}$, then $(\alpha_0^{(k)}, \beta_0^{(k)}, \alpha_1^{(k)}, \beta_1^{(k)})$ verifies it as well for all k, since the same operation is carried out on both α_1 and β_1 during a re-encryption. In order to trace a tag, the attacker therefore has to replace α_1 and β_1 by a same value. When she next interrogates the tag and receives the reply $[(\alpha'_0, \beta'_0); (\alpha'_1, \beta'_1)]$, she verifies if $\alpha'_1 = \beta'_1$. In this case, the interrogated tag is her target with probability $1 - \frac{1}{q}$ where q is the order of $\mathcal{G}$. While the tag could detect such an attack by testing that $\alpha_1 \neq \beta_1$, there are other invariant relations, e.g., the relation $\mathcal{R}'$ such that $(\alpha_0, \beta_0, \alpha_1, \beta_1)$ verifies $\mathcal{R}'$ if, and only if, $\alpha_1 \cdot \beta_1 = 1$ in $\mathcal{G}$. Game 1 given in the appendix is a formalisation of the attack. It shows that the advantage of the adversary considering an existential attack is $1 - 1/(2q)$. So the protocol is not Existential-UNT-Q.

Theorem 1. *Golle, Jakobsson, Juels and Syverson's protocol is neither* Existential-UNT-Q *nor* Existential-UNT-E*.*

3.2 Protocol of Saito, Ryou, and Sakurai

Saito *et al.* also pointed out an attack (see [23]) against the protocol of Golle *et al.* presented in Sect. 3.1, and they subsequently suggested two protocols based on the Golle *et al.*'s protocol. The first one, described below, is called "with a check", and the second one, described in Sect. 3.3, is called "With One-Time Pad". The first protocol is thus an improvement of [12] where the operations carried out by the tag have been modified: the tag checks the new value re-encrypted by the reader before accepting it as the new identifier. The aim is to detect an adversary who would send a wrongly re-encrypted identifier. Therefore, when a tag is queried, it sends its current identifier, $[(\alpha_0, \beta_0); (\alpha_1, \beta_1)]$, and receives the new value $[(\alpha'_0, \beta'_0); (\alpha'_1, \beta'_1)]$. If $|\alpha'_0|, |\beta'_0| \neq 1$ and if $\alpha'_0/{\beta'_0}^x = 1$, where x is the private key of the tag, then $[(\alpha'_0, \beta'_0); (\alpha'_1, \beta'_1)]$ becomes the new current identifier, if not the tag does not renew its content.

Attack based on the private key. The fact that the tag carries out a test based on its public/private key transforms it into an oracle which responds whether this value has been encrypted with its public key or not. In other words, the oracle responds whether or not we are dealing with the traced tag. Let us note, however, that this response from the oracle is internal to the tag. The attacker therefore still has to recover this response. This is rather straightforward because the tag changes its identifier if and only if the test succeeds. So the attacker proceeds as follows. She requests her targeted tag for the first time obtaining thus a reference identifier $[(\alpha_0, \beta_0); (\alpha_1, \beta_1)]$. Subsequently, when the attacker wants to know if a tag corresponds to her target, she interrogates it: she receives (message 2) a value $[(\alpha'_0, \beta'_0); (\alpha'_1, \beta'_1)]$ and resends (message 3) the value $[(\alpha_0, \beta_0); (\alpha_1, \beta_1)]$ to the tag instead of resending the value $[(\alpha'_0, \beta'_0); (\alpha'_1, \beta'_1)]$ re-encrypted. She interrogates the tag once again. If she again receives $[(\alpha'_0, \beta'_0); (\alpha'_1, \beta'_1)]$, this means that the tag has not renewed its identifier and she is not dealing with the traced tag. The traced tag would have recognised $[(\alpha_0, \beta_0); (\alpha_1, \beta_1)]$ as a valid value, meaning encrypted with its public key, and would have used it to refresh its identifier. Game 2 given in the appendix formally describes the attack.

Theorem 2. *Saito, Ryou, and Sakurai's protocol is not* Existential-UNT-Q*.*

3.3 Protocol of Saito, Ryou, and Sakurai, Reloaded

The second protocol suggested in [23] is also based on the universal re-encryption scheme. The fundamental difference compared to [12] is that the re-encryptions are carried out by the tag itself and no longer by the reader. The tag not being able to carry out the exponentiations itself, pre-calculations are carried out by the database and sent to the tag from time to time. We detail the protocol below. To begin with, the tag contains an identifier $\mathrm{ID} = [(\alpha_0, \beta_0); (\alpha_1, \beta_1)]$. It also has a finite list of pairs of random values $\Delta = ((\alpha_1^{r_1}, \beta_1^{r_1}), (\alpha_1^{r_2}, \beta_1^{r_2}), \dots)$ which will allow it to re-encrypt its identifier. The tag also contains a variable i which is the session number, as well as a secret S. All these data are shared with the database. We must consider two distinct operations in this protocol: the reading of the tag and the update of its list of random values, which does not occur at every identification. The procedure unfolds in the following way (see Fig. 4):

1. The reader sends a request to the tag.
2. The tag sends back ID and replaces its identifier by

$$\mathrm{ID}' := [(\alpha_0\alpha_1^{r_k}, \beta_0\beta_1^{r_k}); (\alpha_1\alpha_1^{r_{k+1}}, \beta_1\beta_1^{r_{k+1}})] \text{ where } (\alpha_1^{r_k}, \beta_1^{r_k}),\ (\alpha_1^{r_{k+1}}, \beta_1^{r_{k+1}}) \in \Delta$$

3. If an update of Δ is needed, the reader sends to the tag a new list Δ' of random values and the key $X = h(S, i, \Delta)$, where h is a hash function. If the key is correct, then the tag replaces Δ by Δ' and increments the session number i. If not, the tag does nothing.

Attack based on random values. Knowing the list of the random values contained in the tag allows an adversary to easily trace a tag as she can calculate all the identifiers which will be used by it. So, eavesdropping the communication between the reader and the tag during an update is sufficient to subsequently trace the tag. Since the attacker has to be present during the update (which is only carried out from time to time), she can resolved very easily this problem using a man-in-the-middle attack, since

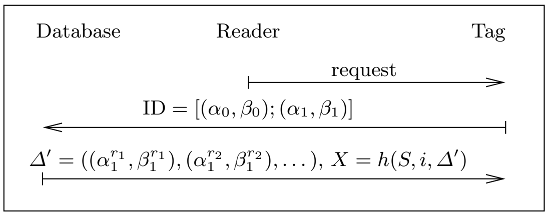

Fig. 4. Protocol of Saito, Ryou, and Sakurai

no authentication is carried out: the tag knows that Δ' has been created by the database but it does not know who is sending it this value. On the other hand, the database does not know that it is sending Δ' to the adversary instead of sending it to the tag. The session number prevents a replay-attack, not a man-in-the-middle attack. So, the protocol is not Existential-UNT-QS.

Attack based on database desynchronisation. The danger which lies in wait for the protocols using synchronised values between the tag and the database, here the session number i, is that an adversary can cause a desynchronisation between the two parties. Here, if an attacker causes the database to send the update message while the tag cannot receive it, then the session number stored by the database will be higher than that stored by the tag. Consequently, all the subsequent updates will fail as the calculation of the key X, which authorises the update, takes into account the current session number. Consequently, the protocol is not Existential-UNT-QS but worse, the protocol is not Universal-UNT-QS because the updates will definitively fail.

Theorem 3. *Reloaded Saito, Ryou, and Sakurai's protocol is not* Universal-UNT-QS*.*

3.4 Protocol of Henrici and Müller

In the protocol of Henrici *et al.* [14], whose flaws have been pointed out by Avoine *et al.* [2], the tag needs to store a (non-static) identifier ID and two variables k and k_{last}. When the system is launched, the tag contains its current identifier ID, the current session number k (both are set up with random values), and k_{last} which is equal to k. The database contains such a 3-uplet per tag it manages, which is initially equal to the values stored in the tag. An identification done out as follows (see Fig. 5):

1. The reader sends a request to the tag.
2. The tag increases its current session number k by one and then sends back $h(\mathrm{ID})$, $h(k \oplus \mathrm{ID})$ and $\Delta k := k - k_{\text{last}}$. $h(\mathrm{ID})$ allows the database to recover the tag's identity; Δk allows the database to recover k and thus to compute $h(k \oplus \mathrm{ID})$, and $h(k \oplus \mathrm{ID})$ aims at thwarting replay attacks.
3. The database checks the validity of these values according to its recorded data. If all is fine, it sends a random number r and $h(r \oplus k \oplus \mathrm{ID})$ to the tag and stores the new values. Since the tag knows k and ID and receives r, it can check whether or not $h(r \oplus k \oplus \mathrm{ID})$ is correct. If this is case, it replaces its identifier by $r \oplus \mathrm{ID}$ and k_{last} by k. Otherwise it does not refresh its identifier.

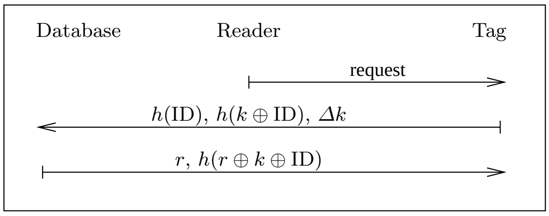

Fig. 5. Protocol of Henrici and Müller

Attack based on non-random information. This attack consists of tracking a tag, taking advantage of the information supplied by Δk. Indeed, since the tag increases its value k every time it receives a request (Step 2) even if the identification finally fails, while k_{last} is updated only when the identification succeeds (Step 3), an attacker may interrogate the tag several times to abnormally increase k and therefore Δk. Thanks to the fact that this value is sent in clear in the second message, the attacker is then able to recognise its target later according to this value: an abnormally high Δk, i.e., far from the expected Δk when no attack occurs. Consequently, the protocol of Henrici and Müller is not Existential-UNT-Q. Game 3 in the appendix formally describes the attack.

Attack based on refreshment avoidance. When a reader requests a tag, the attacker interrogates this tag as well before the reader carries out the third step. Receiving the request from the attacker, the tag increases k. Consequently, the hash value sent by the reader seems to be incorrect since k has now changed. This attack is another example showing that the protocol is not Existential-UNT-Q.

Attack based on database desynchronisation. A more subtle and definitive attack consists of desynchronising the tag and the database. For that, the attacker performs the identification so that the random value r she sends is the neutral element of $\oplus$: the attacker replaces r by the null bit-string and replaces $h(r \oplus k \oplus \text{ID})$ by $h(k \oplus \text{ID})$ obtained by eavesdropping the second message of the current identification. We have trivially $h(\mathbf{0} \oplus k \oplus \text{ID}) = h(k \oplus \text{ID})$. Thus, the tag cannot detect the attack. Then it replaces its identifier by $\mathbf{0} \oplus \text{ID}$ (which is equal to its "old" identifier) and it updates k_{last}. In the next identification, the tag and the database will be desynchronised, since the tag computes the hash value using the "new" k_{last} whereas the database checks the hash value with the "old" k_{last}: the test fails and the received message is discarded. Consequently, the database will never send the third message to refresh the tag's identifier and the tag is definitively traceable. This proves that the protocol is not Universal-UNT-QE.

Theorem 4. *Henrici and Müller's protocol is neither* Existential-UNT-Q*, nor* Universal-UNT-QE.

3.5 Protocol of Weis, Sarma, Rivest, and Engels

We describe in this section the protocol of Weis *et al.* [28] with "Randomised Access Control". In this protocol (see Fig. 6), the information sent by the tag each time it is queried consists of a random value r and a randomised hash value $h(\text{ID}||r)$ where ID is the static identifier of the tag. In order to compute this information, the tag needs a PRNG and an embedded hash function but stores its identifier only. When the database wants to identify the queried tag, it computes from r and the n identifiers it manages the hash values until finding the expected $h(\text{ID}||r)$.

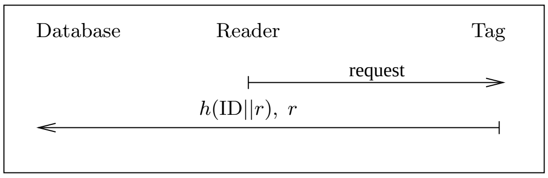

Fig. 6. Protocol of Weis, Sarma, Rivest, and Engels

Analysis. In the random oracle model, the information sent by the tags gives no useful information to an attacker. On the other hand, the reader sends no (useful) information either. Consequently, the protocol is Existential-UNT-QSE which is the strongest security requirement when the attacker cannot tamper with the tag, i.e., use the Reveal oracle. Otherwise, if an attacker tampers with the tag, she obtains its static identifier and therefore she can track all the past events of the tag. Hence, the Weis *et al.*'s protocol is not Forward-UNT-QSER.

Theorem 5. *Weis, Sarma, Rivest, and Engels' protocol is* Existential-UNT-QSE *in the random oracle model but is not* Forward-UNT-QSER.

3.6 Protocol of Ohkubo, Suzuki, and Kinoshita

The protocol of Ohkubo *et al.* [21] is rather similar to the protocol of Weis *et al.* [28]. It consists also of modifying the information sent by the tag each time it is queried by a reader. The difference is that the hash operation is not randomised but is applied on the (non-static) identifier only which is refreshed by the tag itself. For this, the tag needs two hash functions h_1 and h_2. Let ID be the current tag's identifier. The initial value of ID is known by the database. When a reader queries a tag (see Fig. 7), this latter sends $h_1(\mathrm{ID})$ and replaces its identifier by $h_2(\mathrm{ID})$. When the reader receives the tag's response, it sends it to the database which has to identify the corresponding tag. To do this, the database constructs n hash chains (n is the number of tags managed by the database) from the initial identifiers it stores until it finds the expected $h_1(\mathrm{ID})$.

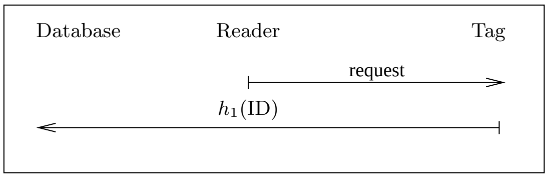

Fig. 7. Protocol of Ohkubo, Suzuki, and Kinoshita

Analysis. In the random oracle model, the information sent by the tags gives no useful information to an attacker. The reader sends no useful information either. Consequently, the protocol is Existential-UNT-QSE. By tampering with the tag, an attacker can obtain its current identifier but she cannot track the tag's past events because h_2 is one way: the protocol is therefore Forward-UNT-QSER.

Theorem 6. *Ohkubo, Suzuki, and Kinoshita's protocol is both* Existential-UNT-QSE *and* Forward-UNT-QSER *in the random oracle model.*

4 Results

In this paper, we have introduced an adversarial model adapted to RFID protocols. We have used this model to analyse the untraceability of many protocols. Due to the lack of space, we presented our analysis of [12], [14], [21], [23], and [28] only. We sum up the results obtained in Table 1.

Protocol	is	is not
Golle, Jakobsson, Juels, and Syverson [12]	–	Existential-UNT-Q Existential-UNT-E
Saito, Ryou, and Sakurai [23]	–	Existential-UNT-Q
Saito, Ryou, and Sakurai, reloaded [23]	–	Universal-UNT-QS
Henrici and Müller [14]	–	Existential-UNT-Q Universal-UNT-QE
Weis, Sarma, Rivest, and Engels [28]	Existential-UNT-QSE	Forward-UNT-QSER
Ohkubo, Suzuki, and Kinoshita [21]	Existential-UNT-QSE Forward-UNT-QSER	

Table 1. Analysis of existing RFID protocols

Note that most of the analysed protocols do not respect the minimum security criteria we could expect, namely Existential-UNT-QSE. However, those which respect these criteria [21,28] suffer from a huge computation complexity. Indeed, the complexity of one identification is linear in terms of tags managed by the database but the complexity becomes quadratic when all the tags managed by the database are identified at the same time (this case appears in many applications, e.g., locating people in an amusement park). Compared with classical cryptographic protocols, the difference comes from the fact that the verifier does not know the identity of the entity with which it is communicating when the protocol starts, and consequently it cannot determine which key it should use. Using asymmetric encryption would be a way to reduce the complexity but this approach is quite unrealistic. Some other approaches have been proposed, e.g., [3] which is based on time-memory trade-offs, and [20] which relies on a three.

Finally, the work presented in this paper is the first step towards the formalisation of the security of RFID protocols in terms of traceability. Whereas the research in this field has been carried out up to now in a rather empirical manner, the need to prove security has henceforth become necessary. Although the RFID technology is widely used, its use in everyday goods has been put back due to traceability issues. Whereas today the organisations for standardisation require security proofs for encryption or signature schemes, no such requirements have been made within the framework of RFID. Our hope is that, in the near future, security proofs will also be required for RFID protocols.

References

1. Gildas Avoine. Privacy issues in RFID banknote protection schemes. In Jean-Jacques Quisquater, Pierre Paradinas, Yves Deswarte, and Anas Abou El Kadam, editors, *International Conference on Smart Card Research and Advanced Applications – CARDIS*, pages 33–48, Toulouse, France, August 2004. IFIP, Kluwer Academic Publishers.
2. Gildas Avoine and Philippe Oechslin. RFID traceability: A multilayer problem. In *Financial Cryptography – FC'05*, LNCS, Roseau, The Commonwealth Of Dominica, February – March 2005. IFCA, Springer-Verlag.
3. Gildas Avoine and Philippe Oechslin. A scalable and provably secure hash based RFID protocol. In *IEEE International Workshop on Pervasive Computing and Communication Security – PerSec 2005*, Kauai Island, Hawaii, USA, March 2005. IEEE, IEEE Computer Society Press.
4. Mihir Bellare, Anand Desai, David Pointcheval, and Phillip Rogaway. Relations among notions of security for public-key encryption schemes. In Hugo Krawczyk, editor, *Advances in Cryptology – CRYPTO'98*, volume 1462 of *LNCS*, pages 26–46, Santa Barbara, California, USA, August 1998. IACR, Springer-Verlag.
5. Mihir Bellare, David Pointcheval, and Phillip Rogaway. Authenticated key exchange secure against dictionary attacks. In Preneel Bart, editor, *Advances in Cryptology – EUROCRYPT'00*, volume 1807 of *LNCS*, pages 139–155, Bruges, Belgium, May 2000. IACR, Springer-Verlag.
6. Emmanuel Bresson, Olivier Chevassut, and David Pointcheval. Security proofs for an efficient password-based key exchange. In Sushil Jajodia, Vijayalakshmi Atluri, and Trent Jaeger, editors, *Conference on Computer and Communications Security – CCS'03*, pages 241–250, Washington, DC, USA, October 2003. ACM, ACM Press.
7. David Chaum. A new paradigm for individuals in the information age. In *IEEE Symposium on Security and Privacy*, pages 99–106, Oakland, California, USA, April 1984. IEEE, IEEE Computer Society Press.
8. Electronic Product Code Global Inc. http://www.epcglobalinc.org.
9. Martin Feldhofer. An authentication protocol in a security layer for RFID smart tags. In *IEEE Mediterranean Electrotechnical Conference – MELECON 2004*, volume 2, pages 759–762, Dubrovnik, Croatia, May 2004. IEEE.
10. Martin Feldhofer, Sandra Dominikus, and Johannes Wolkerstorfer. Strong authentication for RFID systems using the AES algorithm. In Marc Joye and Jean-Jacques Quisquater, editors, *Workshop on Cryptographic Hardware and Embedded Systems – CHES 2004*, volume 3156 of *LNCS*, pages 357–370, Boston, Massachusetts, USA, August 2004. IACR, Springer-Verlag.
11. Kenneth Fishkin, Sumit Roy, and Bing Jiang. Some methods for privacy in rfid communication. In Claude Castelluccia, Hannes Hartenstein, Christof Paar, and Dirk Westhoff, editors, *Security in Ad-hoc and Sensor Networks, First European Workshop – ESAS 2004*, volume 3313 of *Lecture Notes in Computer Science*, pages 42–53, Heidelberg, Germany, August 2005. Springer-Verlag.
12. Philippe Golle, Markus Jakobsson, Ari Juels, and Paul Syverson. Universal re-encryption for mixnets. In Tatsuaki Okamoto, editor, *The Cryptographers' Track at the RSA Conference – CT-RSA*, volume 2964 of *LNCS*, pages 163–178, San Francisco, California, USA, February 2004. Springer-Verlag.
13. Nathan Good, John Han, Elizabeth Miles, David Molnar, Deirdre Mulligan, Laura Quilter, Jennifer Urban, and David Wagner. Radio frequency identification and privacy with information goods. In Sabrina De Capitani di Vimercati and Paul Syverson, editors, *Workshop on Privacy in the Electronic Society – WPES*, pages 41–42, Washington, DC, USA, October 2004. ACM, ACM Press.

14. Dirk Henrici and Paul Müller. Hash-based enhancement of location privacy for radio-frequency identification devices using varying identifiers. In Ravi Sandhu and Roshan Thomas, editors, *IEEE International Workshop on Pervasive Computing and Communication Security – PerSec 2004*, pages 149–153, Orlando, Florida, USA, March 2004. IEEE, IEEE Computer Society Press.
15. International Organization for Standardization. http://www.iso.org.
16. Ari Juels. Minimalist cryptography for low-cost RFID tags. In *Conference on Security in Communication Networks – SCN'04*, LNCS, Amalfi, Italia, September 2004. Springer-Verlag.
17. Ari Juels. "yoking-proofs" for RFID tags. In Ravi Sandhu and Roshan Thomas, editors, *IEEE International Workshop on Pervasive Computing and Communication Security – PerSec 2004*, pages 138–143, Orlando, Florida, USA, March 2004. IEEE, IEEE Computer Society Press.
18. Ari Juels and Ravikanth Pappu. Squealing euros: Privacy protection in RFID-enabled banknotes. In Rebecca N. Wright, editor, *Financial Cryptography – FC'03*, volume 2742 of *LNCS*, pages 103–121, Le Gosier, Guadeloupe, French West Indies, January 2003. IFCA, Springer-Verlag.
19. Ari Juels, Ronald Rivest, and Michael Szydlo. The blocker tag: Selective blocking of RFID tags for consumer privacy. In Vijay Atluri, editor, *Conference on Computer and Communications Security – CCS'03*, pages 103–111, Washington, DC, USA, October 2003. ACM, ACM Press.
20. David Molnar and David Wagner. Privacy and security in library RFID: Issues, practices, and architectures. In Birgit Pfitzmann and Peng Liu, editors, *Conference on Computer and Communications Security – CCS'04*, pages 210–219, Washington, DC, USA, October 2004. ACM, ACM Press.
21. Miyako Ohkubo, Koutarou Suzuki, and Shingo Kinoshita. Cryptographic approach to "privacy-friendly" tags. In *RFID Privacy Workshop*, MIT, MA, USA, November 2003.
22. SafeTzone. http://www.safetzone.com.
23. Junichiro Saito, Jae-Cheol Ryou, and Kouichi Sakurai. Enhancing privacy of universal re-encryption scheme for RFID tags. In Laurence T. Jang, Minyi Guo, Guang R. Gao, and Niraj K. Jha, editors, *Embedded and Ubiquitous Computing – EUC 2004*, volume 3207 of *LNCS*, pages 879–890, Aizu-Wakamatsu City, Japan, August 2004. Springer-Verlag.
24. Sanjay Sarma, Stephen Weis, and Daniel Engels. RFID systems and security and privacy implications. In Burton Kaliski, Çetin Kaya ço, and Christof Paar, editors, *Workshop on Cryptographic Hardware and Embedded Systems – CHES 2002*, volume 2523 of *LNCS*, pages 454–469, Redwood Shores, CA, USA, August 2002. Springer-Verlag.
25. Stop RFID. http://www.stoprfid.org/.
26. Stephen Weis. Security and privacy in radio-frequency identification devices (master thesis), May 2003.
27. Stephen Weis. Security parallels between people and pervasive devices. In *IEEE International Workshop on Pervasive Computing and Communication Security – PerSec 2005*, Kauai Island, Hawaii, USA, March 2005. IEEE, IEEE Computer Society Press.
28. Stephen Weis, Sanjay Sarma, Ronald Rivest, and Daniel Engels. Security and privacy aspects of low-cost radio frequency identification systems. In Dieter Hutter, Günter Müller, Werner Stephan, and Markus Ullmann, editors, *International Conference on Security in Pervasive Computing – SPC 2003*, volume 2802 of *LNCS*, pages 454–469, Boppard, Germany, March 2003. Springer-Verlag.

Appendix

Game 1 (Existential-UNT-Q)
Parameters: $\ell_{\mathrm{ref}} = 1$, $\ell_{\mathrm{chal}} = 1$, $\mathcal{O} = \{\mathsf{Q}\}$.

1. $\mathcal{A}$ requests the *Challenger* then receives her target T.
2. $\mathcal{A}$ chooses $I = \{i\}$ and calls $\mathsf{Query}(\pi_T^i, [(\alpha_0, \beta_0); (\alpha_1, \alpha_1)])$ where $\alpha_0, \beta_0, \alpha_1 \in_R \mathcal{G}$.
   She then receives $\hat{\Omega}_I(T)$.
3. $\mathcal{A}$ requests the *Challenger* then receives T_1 and T_2.
4. $\mathcal{A}$ chooses $I_1 = I_2 = \{i+1\}$.
5. $\mathcal{A}$ calls $\mathsf{Query}(\pi_{T_1}^{i+1}, *)$ and $\mathsf{Query}(\pi_{T_2}^{i+1}, *)$. She then receives

$$[(\alpha_{0_1}, \beta_{0_1}); (\alpha_{1_1}, \beta_{1_1})] \text{ from } \hat{\Omega}_{I_1}(T_1) \text{ and } [(\alpha_{0_2}, \beta_{0_2}); (\alpha_{1_2}, \beta_{1_2})] \text{ from } \hat{\Omega}_{I_2}(T_2).$$

6. If $((\alpha_{1_1} = \beta_{1_1}) \wedge (\alpha_{1_2} \neq \beta_{1_2}))$ then $\mathcal{A}$ outputs T_1 else
   $((\alpha_{1_1} \neq \beta_{1_1}) \wedge (\alpha_{1_2} = \beta_{1_2}))$ then $\mathcal{A}$ outputs T_2 else
   we have $((\alpha_{1_1} = \beta_{1_1}) \wedge (\alpha_{1_2} = \beta_{1_2}))$ therefore $\mathcal{A}$ picks $i \in_R \{1, 2\}$ and outputs T_i.

The advantage of $\mathcal{A}$ is:

$$\mathsf{Adv}^{\mathsf{UNT}}_{\mathrm{Golle}}(\mathcal{A}) = 2\left(1 - \frac{1}{2}\Pr((\alpha_{1_1} = \beta_{1_1}) \wedge (\alpha_{1_2} = \beta_{1_2}))\right) - 1 = 1 - \frac{1}{2q},$$

where q is the order of $\mathcal{G}$. Consequently, the protocol is not Existential-UNT-Q. Note however, the protocol is Universal-UNT-Q because the database re-initialises the tag when a fake re-encryption is found and so the attacker can no longer track the tag. Nevertheless readers cannot detect such a fake-encryption: the database only is able to detect it.

Game 2 (Existential-UNT-Q)
Parameters: $\ell_{\mathrm{ref}} = 1$, $\ell_{\mathrm{chal}} = 2$, $\mathcal{O} = \{\mathsf{Q}\}$.

1. $\mathcal{A}$ requests the *Challenger* then receives her target T.
2. $\mathcal{A}$ chooses $I = \{i\}$ and calls $\mathsf{Query}(\pi^i_T, *)$, thus receiving $[(\alpha_0, \beta_0); (\alpha_1, \beta_1)]$ from $\hat{\Omega}_I(T)$.
3. $\mathcal{A}$ requests the *Challenger* then receives T_1 and T_2.
4. $\mathcal{A}$ chooses $I_1 = I_2 = [i+1, i+2]$.
5. $\mathcal{A}$ calls
   - $\mathsf{Query}(\pi^{i+1}_{T_1}, [(\alpha_{0_1}, \beta_{0_1}); (\alpha_{1_1}, \beta_{1_1})])$ thus receiving $[(\alpha'_{0_1}, \beta'_{0_1}); (\alpha'_{1_1}, \beta'_{1_1})]$ and
   - $\mathsf{Query}(\pi^{i+2}_{T_1}, *)$ thus receiving $[(\alpha''_{0_1}, \beta''_{0_1}); (\alpha''_{1_1}, \beta''_{1_1})]$.
6. If $[(\alpha'_{0_1}, \beta'_{0_1}); (\alpha'_{1_1}, \beta'_{1_1})] = [(\alpha''_{0_1}, \beta''_{0_1}); (\alpha''_{1_1}, \beta''_{1_1})]$ then $\mathcal{A}$ outputs T_1 else $\mathcal{A}$ outputs T_2.

The advantage of $\mathcal{A}$ is clearly 1 because

$$\Pr([(\alpha'_{0_1}, \beta'_{0_1}); (\alpha'_{1_1}, \beta'_{1_1})] = [(\alpha''_{0_1}, \beta''_{0_1}); (\alpha''_{1_1}, \beta''_{1_1})] \mid T_1 \text{ is not the target tag}) = 0.$$

In other word, no tag answers the same value during two consecutive identifications if no attack occurs. Consequently, the protocol is not Existential-UNT-Q.

Game 3 (Existential-UNT-Q)
Parameters: $\ell_{\mathrm{ref}} = n$, $\ell_{\mathrm{chal}} = 1$, $\mathcal{O} = \{\mathsf{Q}\}$.

1. $\mathcal{A}$ requests the *Challenger* then receives her target T.
2. $\mathcal{A}$ chooses $I = [i+1, i+n]$, calls $\mathsf{Query}(\pi^j_T, *)$ for j from $i+1$ to $i+n$. She therefore receives Δk from $\omega_{i+n}(T) \subset \hat{\Omega}_I(T)$.
3. $\mathcal{A}$ requests the *Challenger* thus receiving T_1 and T_2.
4. $\mathcal{A}$ chooses $I_1 = I_2 = \{i+n+1\}$.
5. $\mathcal{A}$ calls $\mathsf{Query}(\pi^{i+n+1}_{T_1}, *)$ and $\mathsf{Query}(\pi^{i+n+1}_{T_2}, *)$, then receives Δk_1 from $\hat{\Omega}_{I_1}(T_1)$ and Δk_2 from $\hat{\Omega}_{I_2}(T_2)$.
6. If $\Delta k_1 = \Delta k + 1$ then $\mathcal{A}$ outputs T_1 otherwise she outputs T_2.

Therefore the advantage of $\mathcal{A}$ is $\mathsf{Adv}^{\mathsf{UNT}}_{\mathrm{Henrici}}(\mathcal{A}) = 2(1 - \frac{1}{2}\Pr(\Delta k_1 = \Delta k_2 = \Delta k + 1)) - 1$ what is non-negligible when $n \gg \tilde{n}$ where $\tilde{n}$ is the expected number of requests between two refreshments of the identifiers. Consequently, the protocol of Henrici and Müller is not Existential-UNT-Q. More precisely, it is not Existential-$\mathsf{UNT}_{n,1}$-Q, meaning that the adversary needs n queries during the initial step of the attack.