# Cryptanalysis of a New Ultralightweight RFID Authentication Protocol—SASI

Raphael C.-W. Phan, *Member*, *IEEE*

**Abstract**—Since RFID tags are ubiquitous and at times even oblivious to the human user, all modern RFID protocols are designed to resist tracking so that the location privacy of the human RFID user is not violated. Another design criterion for RFIDs is the low computational effort required for tags, in view that most tags are passive devices that derive power from an RFID reader's signals. Along this vein, a class of ultralightweight RFID authentication protocols has been designed, which uses only the most basic bitwise and arithmetic operations like exclusive-OR, OR, addition, rotation, and so forth. In this paper, we analyze the security of the SASI protocol, a recently proposed ultralightweight RFID protocol with better claimed security than earlier protocols. We show that SASI does not achieve resistance to tracking, which is one of its design objectives.

**Index Terms**—Security of cryptographic protocols, pervasive and embedded computing, RFID, authentication, ultralightweight, cryptanalysis, traceability, SASI.

## 1 Introduction

RFIDs have found widespread use in many commercial as well as national security applications, ranging from e-passports [10], [4], [9], [12], [19], contactless credit cards [8] to supply chain management [1], [5], [17], [18], [31].

Since RFID tags are mobile and very tiny, attached to diverse items, and often oblivious to the human user, privacy is a major concern in the design and use of RFIDs. Indeed, these tags are commonly embedded in personal devices carried around by an individual wherever he is, e.g., credit cards, e-passports, personal digital assistants (PDAs), Bluetooth devices, clothes that s/he wears, tires on his/her car, and so forth. So if an RFID tag can be tracked, it means the human user's whereabouts can be tracked. It can consequently be argued that one of the fundamental human rights of an individual is that his location or movements should not be trackable, especially if it is without his consent or worse without his knowledge [34]. Thus, designers of RFID protocols want to ensure that RFID tags cannot be tracked, so that location privacy of the human RFID user can be safeguarded. This issue of untraceability has been treated formally in security models, e.g., by Avoine [2], Juels and Weis [11], Le et al. [13], and Vaudenay [32], [33]. Indeed, being able to guarantee untraceability is the first step in achieving privacy even in the sense of anonymity. This is because if an adversary can break the anonymity of RFID protocols, he can then trivially break untraceability, while the converse is not necessarily true. Thus if an RFID protocol is proven to achieve untraceability, then by implication it also achieves anonymity. Untraceability is hence a strong notion of privacy that subsumes anonymity.

Another issue related to the design of RFIDs is the computational effort required at the tag side. This is because most common tags are passive devices in the sense that they derive electrical power from the signals sent by a reader. Thus, most tags cannot be expected to perform computationally intensive operations.

• *The author is with the Department of Electronic and Electrical Engineering, Loughborough University, Loughborough LE11 3TU, Leics, U.K. E-mail: r.phan@lboro.ac.uk.*

*Manuscript received 23 Nov. 2007; accepted 5 June 2008; published online 19 June 2008.*
*For information on obtaining reprints of this article, please send e-mail to: tdsc@computer.org, and reference IEEECS Log Number TDSC-2007-11-0177. Digital Object Identifier no. 10.1109/TDSC.2008.33.*

Peris-Lopez et al. [24], [25], [26] initiated the design of the so-called *ultralightweight* RFID protocols, which involve only simple bitwise logical or arithmetic operations like exclusive-OR (XOR), OR, addition, subtraction, bit rotation, and so forth. Subsequent work appeared in [14], [15], [7], and [6].

In particular, Chien [6] presented the SASI protocol, which is designed to offer better security than previous protocols of Peris-Lopez et al. [24], [25], [26]. SASI is claimed to achieve a list of security properties, including resistance to tracking, i.e., untraceability.

In this paper, we show that SASI does not achieve its design objective of untraceability. Our attack exploits the relationship between the bitwise operations used within SASI. We then draw some lessons to be learned from this.

## 2 Preliminaries

### 2.1 Notations

For better clarity, Table 1 summarizes the symbols and indexing terms used in this paper.

### 2.2 The Juels-Weis Untraceability Model

Rather than reproduce the detailed definitions of the untraceability model proposed by Juels and Weis [11], we briefly describe here, in a style commonly used to define security protocol models [3], the basic ideas of the model [21], [22] that will be sufficient to present our attack later.

A protocol party $\mathcal{P}$ is a $\mathcal{T} \in Tags$ or $\mathcal{R} \in Readers$ interacting in protocol sessions. Adversary $\mathcal{A}$ controls the communications between all protocol parties by interacting either passively or actively with them as defined by the protocol. $\mathcal{A}$'s interactions are formally captured by its ability to issue the following queries:

- $\mathrm{Execute}(\mathcal{R}, \mathcal{T}, i)$ query. This models *passive* attacks, where adversary $\mathcal{A}$ by eavesdropping gets read access to an honest execution of the protocol session $i$ between $\mathcal{R}$ and $\mathcal{T}$.
- $\mathrm{Send}(\mathcal{P}_1, \mathcal{P}_2, i, m)$ query. This models *active* attacks by allowing the adversary $\mathcal{A}$ to impersonate some party $\mathcal{P}_1$ ($\mathcal{P}_1 = \mathcal{R}$ respectively $\mathcal{P}_1 = \mathcal{T}$) in some protocol session $i$ and send a message $m$ of its choice to an instance of some other party $\mathcal{P}_2$ ($\mathcal{P}_2 = \mathcal{T}$ respectively $\mathcal{P}_2 = \mathcal{R}$). Note that this query is a generalization of the TagInit and ReaderInit queries as well as challenge and response messages defined in the Juels-Weis model.
- $\mathrm{Corrupt}(\mathcal{T}, K')$ query. This query allows the adversary $\mathcal{A}$ to learn the stored secret $K$ of the tag $\mathcal{T} \in Tags$, and which further sets the stored secret to $K'$, and is the equivalent of the SetKey query of the Juels-Weis model. This kind of attack is possible in view that RFID tags are typically not designed to be tamper-resistant, thus once tags are deployed it is possible for an adversary to tamper with the tag to read from or write to its nonvolatile memory in which stored secrets are kept. This attack is an invasive one that is much stronger than active attacks captured by the Send query; because Corrupt queries mean that the adversary has physical access to the tag, compared to Send queries where the adversary has access only to the communication channel between the reader and tag. Indeed, in the event that Corrupt queries are possible, i.e., the adversary can read and tamper with the tag's nonvolatile memory used to store secrets, the most that can still be offered is that security (respectively privacy) of previously completed sessions are not compromised. This notion is known as forward security (respectively forward privacy), and it captures the extent of the damage caused by the compromise of the tag's stored secret.

TABLE 1
Notations and Indexing Terms

| | |
|---|---|
| $\mathcal{P}$ | A protocol party: either a reader or a tag |
| $\mathcal{R}$ | An RFID reader |
| $\mathcal{T}$ | An RFID tag |
| $Readers$ | Set of RFID readers |
| $Tags$ | Set of RFID tags |
| $\mathcal{A}$ | An adversary |
| $x << y$ | Bit rotation of $x$ to the left by $y$ bits |
| $\|\|$ | Concatenation |
| $\|x\|$ | The magnitude of $x$ |
| $X_{\mathsf{LSB}}$ | Least significant bit (LSB) of $X$ |
| $\Pr[X]$ | Probability of event $X$ |
| $\varepsilon(k)$ | Negligible function of $k$, i.e. one that approaches zero faster than the reciprocal of any polynomial $p(k)$. More formally, $\varepsilon : \mathbb{N} \rightarrow \mathbb{R}$ is negligible if for any nonzero polynomial $p(\cdot)$ there exists an $m$ such that $\forall n > m, \|\varepsilon(n)\| < \frac{1}{p(n)}$ |
| dB | Record in memory containing stored secret(s) of a reader or tag |
| $Untraceability$ (UNT) | The fact that an adversary cannot tell between any two tags |
| $Indistinguishability$ (IND) | A common notion used in cryptography to mean that an adversary cannot tell which of any two plaintexts is associated with a particular ciphertext |
| Execute | A query made by an adversary to model passive attacks i.e. eavesdropping |
| Send | A query made by an adversary to model active attacks on the communication channel |
| Corrupt | A query made by an adversary to model invasive physical attacks that allow to recover stored secret(s) of the tag |
| Test | A query made by an adversary to allow the indistinguishability-based definition of untraceability |

- $\mathrm{Test}(i, \mathcal{T}_0, \mathcal{T}_1)$ query. This query does not correspond to any of $\mathcal{A}$'s abilities but rather is included so that we can define the indistinguishability-based [20], [30] notion of *untraceability* (UNT). Upon the issuance of a Test query for session $i$, then depending on a randomly chosen bit $b \in \{0, 1\}$, $\mathcal{A}$ is given $ID_b$ from the set $\{ID_0, ID_1\}$ corresponding to tags $\{\mathcal{T}_0, \mathcal{T}_1\}$. Informally, $\mathcal{A}$ succeeds if it can guess the bit $b$.

Now that the adversary's abilities are clear, then untraceability (UNT) is defined using the game $\mathcal{G}$ played between an adversary $\mathcal{A}$ and a collection of reader and tag instances. $\mathcal{A}$ runs the game $\mathcal{G}$ whose setting is given as follows (see Fig. 1):

**Phase 1 (Learning).** $\mathcal{A}$ can send any Execute, Send, and Corrupt queries. This phase models the adversary $\mathcal{A}$ interacting with reader and tag instances in protocol sessions, and its ability to mount passive attacks to eavesdrop on protocol messages, or active attacks to modify, insert, or delete messages, or even to tamper with the tag's nonvolatile memory.

**Phase 2 (Challenge).**

1. Sometime during $\mathcal{G}$, $\mathcal{A}$ chooses two fresh tag identifiers $ID_0$, $ID_1$ (corresponding to tags $\mathcal{T}_0$, $\mathcal{T}_1$) to be tested and sends a Test query corresponding to this. Freshness means that the tags have not been issued any Corrupt query. Depending on a randomly chosen bit $b \in \{0, 1\}$, $\mathcal{A}$ is given a challenger identifier $ID_b$ from the set $\{ID_0, ID_1\}$.
2. $\mathcal{A}$ continues making any Execute, Send, and Corrupt queries, subjected to the restriction that the tags $\mathcal{T}_0$, $\mathcal{T}_1$ are not issued any Corrupt query.

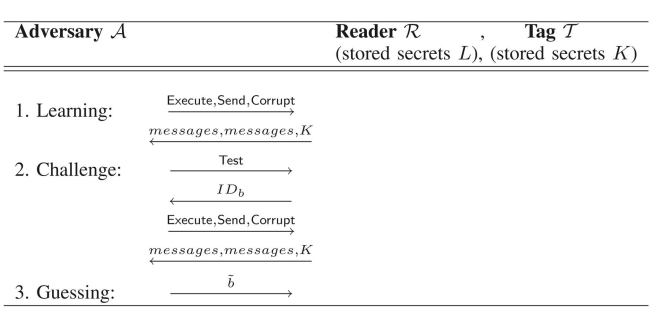


Fig. 1. The untraceability (UNT) model.

**Phase 3 (Guessing).** Eventually, $\mathcal{A}$ terminates the game and outputs a bit $\tilde{b}$ as its guess of the value of $b$.

The definition of this game is similar in style to the indistinguishability-based game definitions for security protocols, e.g., [3]; indeed, an RFID authentication protocol is a security protocol. This UNT game models the untraceability notion because if $\mathcal{A}$ cannot even be able to distinguish between any two tags, i.e., it fails to win the game, then clearly it cannot track any tag since it cannot tell if tags are the same ones or not. This models the fact that the adversary cannot get even one bit of privacy information from the protocol.

The success of $\mathcal{A}$ in winning $\mathcal{G}$ therefore translates to its success of breaking untraceability and is quantified in terms of $\mathcal{A}$'s advantage in distinguishing whether $\mathcal{A}$ received $ID_0$ or $ID_1$, i.e., it correctly guesses $b$, compared to randomly flipping a coin for the value of $b$. This is denoted by $\mathrm{Adv}_{\mathcal{A}}^{\mathrm{UNT}}(k)$, where $k$ is the security parameter, e.g., bit length of some secret unknown to the adversary.

Then, we have

$$\mathrm{Adv}_{\mathcal{A}}^{\mathrm{UNT}}(k) = |\Pr[\mathcal{A} \text{ wins}] - \Pr[\text{random coin flip}]| \tag{1}$$

$$= \left|\Pr[\tilde{b} = b] - \frac{1}{2}\right|, \tag{2}$$

where $\Pr[\tilde{b} = b]$ for a typical adversary is a function of $k$. An RFID protocol achieves untraceability (UNT) if $\mathrm{Adv}_{\mathcal{A}}^{\mathrm{UNT}}(k) < \varepsilon(k)$ for some negligible function $\varepsilon(\cdot)$. As an illustrative example, consider if the probability for the adversary to win the game, i.e., $\Pr[\tilde{b} = b]$, is $\frac{1}{2}$. Then, $\mathrm{Adv}_{\mathcal{A}}^{\mathrm{UNT}}(k)$ is zero, meaning the adversary has zero advantage in winning the game since he could just as well have flipped a coin to make the guess, which would have given him the same probability of winning. Thus, the protocol did not leak even one bit of information to the adversary.

| **Reader** $\mathcal{R}$ | | **Tag** $\mathcal{T}_j$ |
|---|---|---|
| $(IDS_i, K1_i, K2_i), ID_j \in$ dB | | $(IDS_{old}, K1_{old}, K2_{old}), (IDS_{next}, K1_{next}, K2_{next}), ID_j \in$ dB |
| | $\xrightarrow{hello}$ | $IDS = IDS_{next}$, $K1 = K1_{next}$, $K2 = K2_{next}$. |
| | | If $hello$ received a second time, |
| | $\xleftarrow{IDS}$ | then $IDS = IDS_{old}$, $K1 = K1_{old}$, $K2 = K2_{old}$. |
| If not $\exists IDS : (IDS_i, K1_i, K2_i) \in$ dB: | | |
| resend $hello$ to $\mathcal{T}_j$ and wait for $IDS$. | | |
| Else: | | |
| $IDS = IDS_i$, $K1 = K1_i$, $K2 = K2_i$. | | |
| Randomly generate $n1$, $n2$. | | |
| $A = IDS \oplus K1 \oplus n1$. | | |
| $B = (IDS \vee K2) + n2$. | | |
| $\hat{K}1 = (K1 \oplus n2) << K1$. | | |
| $\hat{K}2 = (K2 \oplus n1) << K2$. | | |
| $C = (K1 \oplus \hat{K}2) + (\hat{K}1 \oplus K2)$. | | |
| | $\xrightarrow{A\|\|B\|\|C}$ | |
| | | $n1 = A \oplus IDS \oplus K1$. |
| | | $n2 = B - (IDS \vee K2)$. |
| | | $\hat{K}1 = (K1 \oplus n2) << K1$. |
| | | $\hat{K}2 = (K2 \oplus n1) << K2$. |
| | | $\hat{C} = (K1 \oplus \hat{K}2) + (\hat{K}1 \oplus K2)$. |
| | | If $C = \hat{C}$: |
| | $\xleftarrow{D}$ | $D = (\hat{K}2 + ID_i) \oplus ((K1 \oplus K2) \vee \hat{K}1)$. |
| $\hat{D} = (\hat{K}2 + ID_i) \oplus ((K1 \oplus K2) \vee \hat{K}1)$ | | $IDS_{old} = IDS$, $IDS_{next} = (IDS + ID_i) \oplus (n2 \oplus \hat{K}1)$. |
| If $D = \hat{D}$: | | $K1_{old} = K1$, $K2_{old} = K2$. |
| $IDS_i = (IDS_i + ID_i) \oplus (n2 \oplus \hat{K}1)$ | | $K1_{next} = \hat{K}1$, $K2_{next} = \hat{K}2$. |
| $K1_i = \hat{K}1$, $K2_i = \hat{K}2$. | | |

Fig. 2. The SASI protocol.

## 3 THE SASI PROTOCOL

SASI [6] is a very recent ultralightweight RFID protocol, which is designed to be more secure than earlier such kinds of protocols [24], [25],[26].

Since the communication between the reader and the back-end server is assumed to be secure, SASI considers the reader and server as one entity. Each tag $\mathcal{T}_j$ has a 96-bit static identification $ID_j$ and for a particular session $i$ preshares with the reader a 96-bit pseudonym $IDS_i$ and two secret keys $K1_i$, $K2_i$ each of 96 bits. Every tag keeps two entries, each of the form $(IDS_i, K1_i, K2_i)$, where one corresponds to old values used in the most recent completed protocol session, while the other corresponds to the stored values to be used in the next protocol session.

A tag is not expected to perform any computations except for basic bitwise logical or arithmetic operations like XOR ($\oplus$), OR ($\vee$), addition ($+$), subtraction ($-$), and bit rotation ($<<$).

The SASI protocol consists of the tag identification phase, mutual authentication phase, and updating phase (see Fig. 2 for more details).

**Tag Identification:**

1. The reader $\mathcal{R}$ sends a $hello$ message to the tag $\mathcal{T}$.
2. $\mathcal{T}$ sets the pseudonym $IDS$ to the value of $IDS_{next}$ from its record. It also sets $K1$ and $K2$ to, respectively, the values of $K1_{next}$ and $K2_{next}$. It then sends $IDS$ to $\mathcal{R}$.
3. $\mathcal{R}$ checks if there exists an entry $IDS_i$ in its record that equals the received $IDS$. If not, it resends the $hello$ message to $\mathcal{T}$ and waits for an $IDS$ message.
4. Upon receiving a second $hello$ message, $\mathcal{T}$ now sets the pseudonym $IDS$ to the value of $IDS_{old}$ from its record, and correspondingly $K1$ and $K2$ are set equal to $K1_{old}$ and $K2_{old}$, respectively.
5. Once $\mathcal{R}$ finds an entry $IDS_i$ in its record that is equal to the received $IDS$, it proceeds to the next steps with $IDS_i$ and corresponding $K1_i$, $K2_i$ from the record entry.

**Mutual Authentication:**

6. $\mathcal{R}$ randomly generates two numbers $n1$, $n2$, and proceeds to compute the values $A$, $B$, $\hat{K}1$, $\hat{K}2$, $C$ as per Fig. 2, involving XOR, OR, addition, and bit rotation.
7. The concatenation of $A\|B\|C$ is then sent to $\mathcal{T}$.
8. $\mathcal{T}$ computes the numbers $n1$, $n2$ and values $\hat{K}1$, $\hat{K}2$ from the received $A\|B\|C$. It then computes $\hat{C}$ from the values of $K1$, $K2$, $\hat{K}1$, $\hat{K}2$ via the XOR and addition operations, as per Fig. 2.
9. If the computed $\hat{C}$ is equal to the received $C$, then $\mathcal{T}$ computes $D$ via XOR, addition, and OR operations, as per Fig. 2.
10. This $D$ is sent to $\mathcal{R}$, and $\mathcal{T}$ now proceeds to the Updating phase.
11. $\mathcal{R}$ computes $\hat{D}$ as per Fig. 2 and checks if it equals the received $D$. If so, $\mathcal{R}$ proceeds to the Updating phase.

**Updating:**

12. $\mathcal{R}$ updates its record entry for $(IDS_i, K1_i, K2_i)$, while $\mathcal{T}$ updates its record entry for $(IDS_{old}, K1_{old}, K2_{old})$, $(IDS_{next}, K1_{next}, K2_{next})$ as per Fig. 2.

At the completion of the protocol, both the reader and the tag have successfully authenticated each other and, furthermore, have updated their stored record entries in preparation for the next protocol session. Since these updates are functions of the newly computed and exchanged $\hat{K}1$, $\hat{K}2$, and they have verified the received $C$, $D$ against their own computed $\hat{C}$, $\hat{D}$, which are functions of $\hat{K}1$, $\hat{K}2$, then both the reader and tag are also assured that they have the same $\hat{K}1$, $\hat{K}2$ values and are in synchrony, thus preventing desynchronization attacks.

SASI is claimed to provide a list of security properties, including mutual authentication, tag anonymity, untraceability,

and forward security. Its untraceability is claimed due to the frequent update of the pseudonym $IDS$ dependent on random numbers, so the argument is that any two pseudonyms should be random and not linkable with each other.

## 4 Breaking the Tracking Resistance of SASI

One of the goals of SASI's design was to achieve resistance to tracking, i.e., untraceability, and it is claimed to offer more security properties than earlier ultralightweight RFID protocols like LMAP, M$^2$AP, and EMAP [24], [25], [26].

We show how to track tags in the SASI protocol and thus break untraceability. In particular, consider an adversary $\mathcal{A}$ performing the following steps:

1. **Learning:** Issue an Execute query to eavesdrop on a protocol session between the reader and a tag $\mathcal{T}_0$, to obtain $C$ and $D$.
2. **Challenge:** Some time later, the adversary $\mathcal{A}$ chooses two fresh tags $\mathcal{T}_0$, $\mathcal{T}_1$ with identifiers $ID_0$, $ID_1$, where $ID_0 \equiv 0 \bmod 2$ and $ID_1 \equiv 1 \bmod 2$. $\mathcal{A}$ then sends a Test query for these. Adversary $\mathcal{A}$ is then given a test challenge identifier $ID_b \in \{ID_0, ID_1\}$. Note that by construction, $b = ID_{b\mathrm{LSB}}$.
3. **Guessing:** The adversary $\mathcal{A}$ outputs a guess $\tilde{b} = C_{\mathrm{LSB}} \oplus D_{\mathrm{LSB}}$.

Next, we show why $\mathrm{Adv}_{\mathcal{A}}^{\mathrm{UNT}}(k)$ is nonnegligible:

$$\mathrm{Adv}_{\mathcal{A}}^{\mathrm{UNT}}(k) = \left|\Pr[\mathcal{A} \text{ wins}] - \frac{1}{2}\right| \tag{3}$$

$$= \left|\Pr[\tilde{b} = b] - \frac{1}{2}\right| \tag{4}$$

$$= \left|\Pr[C_{\mathrm{LSB}} \oplus D_{\mathrm{LSB}} = b] - \frac{1}{2}\right| \tag{5}$$

$$= \left|\Pr[C_{\mathrm{LSB}} \oplus D_{\mathrm{LSB}} = ID_{b\mathrm{LSB}}] - \frac{1}{2}\right| \tag{6}$$

$$= \left|\frac{3}{4} - \frac{1}{2}\right| \tag{7}$$

$$= \frac{1}{4} > \varepsilon(k), \tag{8}$$

where the security parameter $k$ is essentially the bit length of the unknown secret $ID_j$, compromise of which would allow the adversary to win the game.

Equations (3) to (6) are straightforward from the definition of the untraceability game and from the description of the adversary above. To see the reasoning behind (7), recall the values of $C$ and $D$:

$$C = (K1 \oplus \hat{K}2) + (\hat{K}1 \oplus K2), \tag{9}$$

$$D = (\hat{K}2 + ID_i) \oplus \big((K1 \oplus K2) \vee \hat{K}1\big). \tag{10}$$

If we only concentrate on the LSBs of $C$ and $D$, we obtain

$$C_{\mathrm{LSB}} = K1_{\mathrm{LSB}} \oplus \hat{K}2_{\mathrm{LSB}} \oplus \hat{K}1_{\mathrm{LSB}} \oplus K2_{\mathrm{LSB}}, \tag{11}$$

$$D_{\mathrm{LSB}} = \hat{K}2_{\mathrm{LSB}} \oplus ID_{i\mathrm{LSB}} \oplus \big((K1_{\mathrm{LSB}} \oplus K2_{\mathrm{LSB}}) \vee \hat{K}1_{\mathrm{LSB}}\big). \tag{12}$$

Equations (11) and (12) stem from the fact that addition $(+)$ equals XOR $(\oplus)$ for the LSB.

Now, by inspection of the truth table for XOR $(\oplus)$ and OR $(\vee)$, we see that they are equal for a fraction $\frac{3}{4}$ of the time. Thus, we can rewrite (12) as a probabilistic equation depending on a probability $p = \frac{3}{4}$:

$$D_{\mathrm{LSB}} \stackrel{p}{=} \hat{K}2_{\mathrm{LSB}} \oplus ID_{i\mathrm{LSB}} \oplus K1_{\mathrm{LSB}} \oplus K2_{\mathrm{LSB}} \oplus \hat{K}1_{\mathrm{LSB}}. \tag{13}$$

Combining (11) and (13), we have

$$C_{\mathrm{LSB}} \oplus D_{\mathrm{LSB}} \stackrel{p}{=} ID_{b\mathrm{LSB}},$$

and (7) follows.

Since $\mathrm{Adv}_{\mathcal{A}}^{\mathrm{UNT}}(k) = \frac{1}{4}$ is nonnegligible, hence SASI does not achieve untraceability.

## 5 Concluding Remarks and Discussion

We have shown that the SASI protocol cannot achieve untraceability even under a passive attack. The weakness we exploit is that the public $C$ and $D$ messages are each a function of the same unknown secrets $K1$, $K2$, $\hat{K}1$, $\hat{K}2$, and the static identifier $ID$ is only contained in $D$; thus by further exploiting the bit interaction between the operators $\oplus$, $+$, and $\vee$ and canceling out the secrets $K1$, $K2$, $\hat{K}1$, $\hat{K}2$, we showed that $C$ and $D$ in combination leaks at least one bit of information about the static identifier $ID$ of a tag.

The claim in [6] that SASI achieves untraceability is grounded on the fact that the pseudonym $IDS$ is updated at every session as a function of random numbers, and hence any two pseudonyms are expected to look random and thus be unlinkable. Yet, our attack does not exploit nonrandomness in the pseudonyms at all and instead shows that the LSB of $C$ and $D$ are less randomized compared to other bits, and what is worse for SASI is that with nonnegligible probability their XOR difference is independent of the randomly generated $\hat{K}1$ and $\hat{K}2$ and instead is only dependent on the static identification $ID$. To put this into perspective: rather than linking two pseudonyms $IDS$ to track a tag, our attack instead directly tracks a tag by obtaining information about its static identification $ID$. So, the flaw in the untraceability argument of SASI in [6] was assuming a particular way (that of linking two pseudonyms) for which an adversary mounts a tracking attack.

Our attack is a passive one. Passive attacks are feasible in practice since they only require eavesdropping, which is a typical threat in RFID setting where the physical wireless communication channel is open to parties within transmission range. With the ubiquity of RFID tags, whose existence is often oblivious to the human user or whose embedded presence around the human is often beyond his/her control, the location privacy of the user therefore comes under threat when s/he goes about daily activities, coming into contact with other parties that share the common physical wireless space, and that could potentially launch passive attacks without being noticed.

Interestingly, the earlier ultralightweight RFID protocols LMAP, M$^2$AP, and EMAP by Peris-Lopez et al. [24], [25], [26] do not exhibit the abovementioned properties that we exploited for our attack on SASI. The main reason is because any combination of their messages $A$, $B$, $C$, $D$, $(E)$ does not allow to cancel out all of the unknown secrets $K1$, $K2$, $(K3)$, $(K4)$ or random numbers $n1$, $n2$; and so it is not possible to leak information about the static identifier $ID$ in this way.

This paper is a further case to support [28], [29], [27] that newer protocol versions designed with better security should not necessarily be taken for granted to be more secure than older

versions, even against attacks considered by both old and new designs, e.g., in this case, untraceability.

## ACKNOWLEDGMENTS

Part of this work was done while the author was with the Laboratoire de Sécurité et de Cryptographie (LASEC), Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland. The author would like to thank the anonymous referees for constructive comments on this paper; as a consequence, the model, protocol, and results are treated in more detail. The author would also like to thank God for His many blessings.

## REFERENCES

[1] "Albertsons Announces Mandate," *RFID J.*, http://www.rfidjournal.com/article/articleview/819/1/1/, Mar. 2004.
[2] G. Avoine, *Adversarial Model for Radio Frequency Identification*, Cryptology ePrint Archive, report 2005/049, IACR ePrint Archive, http://eprint.iacr.org/2005/049, Feb. 2005.
[3] M. Bellare, D. Pointcheval, and P. Rogaway, "Authenticated Key Exchange Secure against Dictionary Attacks," *Proc. Int'l Conf. Theory and Application of Cryptographic Techniques (EUROCRYPT '00)*, pp. 139-155, 2000.
[4] D. Carluccio, K. Lemke, and C. Paar, "E-Passport: The Global Traceability or How to Feel Like a UPS Package," *Proc. Seventh Int'l Workshop Information Security Applications (WISA '07)*, pp. 391-404, 2007.
[5] CASPIAN, *Boycott Benetton*, http://www.boycottbenetton.com, 2007.
[6] H.-Y. Chien, "SASI: A New Ultralightweight RFID Authentication Protocol Providing Strong Authentication and Strong Integrity," *IEEE Trans. Dependable and Secure Computing*, vol. 4, no. 4, pp. 337-340, Oct.-Dec. 2007.
[7] H.-Y. Chien and C.-W. Huang, "Security of Ultra-Lightweight RFID Authentication Protocols and Its Improvements," *ACM Operating System Rev.*, vol. 41, no. 2, pp. 83-86, 2007.
[8] T.S. Heydt-Benjamin, D.V. Bailey, K. Fu, A. Juels, and T. O'Hare, "Vulnerabilities in First-Generation RFID-Enabled Credit Cards," *Proc. 11th Int'l Conf. Financial Cryptography and Data Security (FC '07)*, pp. 2-14, 2007.
[9] J.-H. Hoepman, E. Hubbers, B. Jacobs, M. Oostdijk, and R.W. Schreur, "Crossing Borders: Security and Privacy Issues of the European e-Passport," *Proc. First Int'l Workshop Security (IWSEC '06)*, pp. 152-167, 2006.
[10] A. Juels, D. Molnar, and D. Wagner, "Security and Privacy Issues in E-Passports," *Proc. First IEEE Conf. Security and Privacy for Emerging Areas in Comm. Networks (SecureComm '05)*, http://eprint.iacr.org/2005/095, last revised Sept. 2007, pp. 74-88, 2005.
[11] A. Juels and S.A. Weis, "Defining Strong Privacy for RFID," *Proc. Fifth Ann. IEEE Int'l Conf. Pervasive Computing and Comm. (PerCom '07)*, http://eprint.iacr.org/2006/137, pp. 342-347, Mar. 2007.
[12] E. Kosta, M. Meints, M. Hensen, and M. Gasson, "An Analysis of Security and Privacy Issues Relating to RFID Enabled ePassports," *Proc. 22nd IFIP TC-11 Int'l Information Security Conf. (IFIP SEC '07)*, vol. 232, pp. 467-472, 2007.
[13] T.V. Le, M. Burmester, and B. de Medeiros, "Universally Composable and Forward-Secure RFID Authentication and Authenticated Key Exchange," *Proc. ACM Symp. Information, Computer and Comm. Security (ASIACCS '07)*, http://eprint.iacr.org/2007/051, pp. 242-252, 2007.
[14] T. Li and R.H. Deng, "Vulnerability Analysis of EMAP—An Efficient RFID Mutual Authentication Protocol," *Proc. Second Int'l Conf. Availability, Reliability and Security (AReS '07)*, pp. 238-245, 2007.
[15] T. Li and G. Wang, "Security Analysis of Two Ultra-Lightweight RFID Authentication Protocols," *Proc. 22nd IFIP TC-11 Int'l Information Security Conf. (IFIP SEC '07)*, vol. 232, pp. 109-120, 2007.
[16] T. Li, G. Wang, and R.H. Deng, "Security Analysis on a Family of Ultra-Lightweight RFID Authentication Protocols," *J. Software*, vol. 3, no. 3, pp. 1-10, 2008.
[17] "Michelin Embeds RFID Tags in Tires," *RFID J.*, http://www.rfidjournal.com/article/articleview/269/1/1/, Jan. 2003.
[18] "Mitsubishi Electric Asia Switches on RFID," *RFID J.*, http://www.rfidjournal.com/article/articleview/2644/, Sept. 2006.
[19] J. Monnerat, S. Vaudenay, and M. Vuagnoux, "About Machine-Readable Travel Documents: Privacy Enhancement Using (Weakly) Non-Transferable Data Authentication," *Proc. Third Workshop RFID Security (RFIDSec '07)*, pp. 15-28, 2007.
[20] M. Naor and M. Yung, "Public-Key Cryptosystems Provably Secure against Chosen Ciphertext Attacks," *Proc. ACM Symp. Theory of Computing (STOC '90)*, pp. 427-437, 1990.
[21] K. Ouafi and R.C.-W. Phan, "Privacy of Recent RFID Authentication Protocols," *Proc. Fourth Information Security Practice and Experience Conf. (ISPEC '08)*, pp. 263-277, 2008.
[22] K. Ouafi and R.C.-W. Phan, "Traceable Privacy of Recent Provably-Secure RFID Protocols," *Proc. Sixth Int'l Conf. Applied Cryptography and Network Security (ACNS '08)*, pp. 479-489, 2008.
[23] R.I. Paise and S. Vaudenay, "Mutual Authentication in RFID," *Proc. ACM Symp. Information, Computer and Comm. Security (ASIACCS '08)*, pp. 292-299, 2008.
[24] P. Peris-Lopez, J.C. Hernandez-Castro, J.M. Estevez-Tapiador, and A. Ribagorda, "LMAP: A Real Lightweight Mutual Authentication Protocol for Low-Cost RFID Tags," *Proc. Second Workshop RFID Security (RFIDSec '06)*, July 2006.
[25] P. Peris-Lopez, J.C. Hernandez-Castro, J.M. Estevez-Tapiador, and A. Ribagorda, "EMAP: A Efficient Mutual Authentication Protocol for Low-Cost RFID Tags," *Proc. OTM Information Security Workshop (IS '06)*, pp. 352-361, 2006.
[26] P. Peris-Lopez, J.C. Hernandez-Castro, J.M. Estevez-Tapiador, and A. Ribagorda, "$M^2AP$: A Minimalist Mutual-Authentication Protocol for Low-Cost RFID Tags," *Proc. Third Int'l Conf. Ubiquitous Intelligence and Computing (UIC '06)*, pp. 912-923, 2006.
[27] R.C.-W. Phan, K.-K.R. Choo, and S.-H. Heng, "Security of a Leakage-Resilient Protocol for Key Establishment and Mutual Authentication," *Proc. Int'l Conf. Provable Security (ProvSec '07)*, pp. 169-177, 2007.
[28] R.C.-W. Phan and B.-M. Goi, "Cryptanalysis of the N-Party Encrypted Diffie-Hellman Key Exchange Using Different Passwords," *Proc. Fourth Int'l Conf. Applied Cryptography and Network Security (ACNS '06)*, pp. 226-238, 2006.
[29] R.C.-W. Phan and B.-M. Goi, "Cryptanalysis of Two Provably Secure Cross-Realm C2C-PAKE Protocols," *Proc. Seventh Int'l Conf. Cryptology in India (Indocrypt '06)*, pp. 104-117, 2006.
[30] C. Rackoff and D. Simon, "Non-interactive Zero-Knowledge Proof of Knowledge and Chosen Ciphertext Attack," *Proc. 11th Ann. Int'l Cryptology Conf. (CRYPTO '91)*, pp. 433-444, 1991.
[31] "Target, Wal-Mart Share EPC Data," *RFID J.*, http://www.rfidjournal.com/article/articleview/642/1/1/, Oct. 2005.
[32] S. Vaudenay, "RFID Privacy Based on Public-Key Cryptography," *Proc. Ninth Ann. Int'l Conf. Information Security and Cryptology (ICISC '06)*, pp. 1-6, 2006.
[33] S. Vaudenay, "On Privacy Models for RFID," *Proc. 13th Ann. Int'l Conf. Theory and Application of Cryptology and Information Security (Asiacrypt '07)*, pp. 68-87, 2007.
[34] A. Westin, *Privacy and Freedom*. Atheneum, 1967.