Quasi-Linear Cryptanalysis of a Secure RFID Ultralightweight Authentication Protocol

Pedro Peris-Lopez[1], Julio Cesar Hernandez-Castro[2], Raphael C.-W. Phan[3], Juan M. E. Tapiador[4], and Tieyan Li[5]

[1] Security & Privacy Lab, Faculty of EEMCS, Delft University of Technology
[2] School of Computing, University of Portsmouth
[3] Department of Electronic and Electrical Engineering, Loughborough University
[4] Department of Computer Science, University of York
[5] Institute for Infocomm Research, A*STAR Singapore

Abstract. In 2010, Yeh, Lo and Winata [1] proposed a process-oriented ultralightweight RFID authentication protocol. This protocol is claimed to provide strong security and robust privacy protection, while at the same time the usage of resources on tags is optimized. Nevertheless, in this paper we show how the protocol does not achieve any of its intended security objectives; the main result is that the most valuable information stored on the tag, that is, the static identifier ID, is easily recovered even by a completely passive attacker in a number of ways. More precisely, we start by presenting a traceability attack on the protocol that allows tags to be traced. This essentially exploits the fact that the protocol messages leak out at least one bit of the static identifier. We then present a passive attack (named Norwegian attack) that discloses $\lfloor \log_2 L \rfloor$ bits of the ID, after observing roughly $O(L)$ authentication sessions. Although this attack may seem less feasible in retrieving the full 96-bits of the ID due to the large number of eavesdropped sessions involved, it is already powerful enough to serve as a basis for a very effective traceability attack. Finally, our last attack represents a step forward in the use of a recent cryptanalysis technique (called Tango attack [2]), which allows for an extremely efficient full disclosure attack, capable of revealing the value of the whole ID after eavesdropping only a very small number of sessions.

Keywords: RFID, Cryptanalysis, Ultralightweight, Authentication

1 Introduction

In the RFID context, some researchers have dealt with the stimulating challenge of designing secure RFID protocols based only on simple bitwise logical or arithmetic operations such as bitwise XOR, OR, AND and modular addition. This type of RFID protocols are categorized as ultralightweight protocols, and are intended for very low-cost tags. In 2006, the UMAP family of protocols [3–5] was introduced and attracted certain attention of the research community. After some rounds of cryptanalysis of these schemes, many (if not all) of its security objectives were circumvented, e.g. with active attacks [7, 8] and later with

passive attacks [9, 10]. They served, however, as interesting thought-provoking proposals that influenced later ultralightweight RFID designs. In 2007, Chien proposed the SASI protocol [11], which aims to provide a better security margin and requires only a tiny footprint. The main contribution was the addition of the bitwise rotations to the set of operations supported on tag. Despite this twist in the design of the protocol, some attacks were subsequently published [6, 12–14]. In 2008, Peris-Lopez et al. introduced a new protocol, named Gossamer inspired by both the UMAP family and SASI. The operations on tags are limited in this case to bitwise XOR, addition and left rotation. A key factor in the design of Gossamer is the inclusion of the $MixBits$ function. This is a very lightweight function with highly non-linear relations between inputs and outputs (see the original paper for details [15]). A desynchronization attack [16] conducted by an active attacker is, to the best of our knowledge, the only attack to date proposed against Gossamer. As an alternative to Gossamer, Yeh, Lo and Winata recently presented a new ultralightweight authentication protocol [1]. The protocol is claimed to provide strong security, and to optimize the use of the tag memory in comparison with Gossamer.

This paper presents various cryptanalytic results on the Yeh-Lo-Winata protocol. All the attacks can be mounted by passive adversaries, and thus are highly feasible. The organization of the paper is as follows. In Section 2, the protocol is briefly introduced. Then in subsequent sections, three passive attacks are presented, the last two being able to disclose the static identifier ID, thus breaking tag privacy (i.e. information and location). The reader should note that it is commonly assumed that ultralightweight RFID protocols should be resistant against passive attacks, but not necessarily to active ones. Section 3 presents a traceability attack that shows how the protocol messages leak at least one bit of information on the static identifier ID. In Section 4 we present a Norwegian attack [6] that allows to disclose $\lfloor \log_2 L \rfloor$ bits of the ID, after observing roughly $O(L)$ authentication sessions. While the number of sessions that the adversary has to eavesdrop may be large, nevertheless the attack is quite effective when only the knowledge of some bits is needed to guarantee a successful attack, e.g. in a traceability attack. In Section 5, a much more powerful and efficient full disclosure attack (the Tango attack [2]) is introduced and discussed. More precisely, the adversary listens to a small number of legitimate sessions and the protected ID is almost fully disclosed. Finally, we draw some conclusions and end with some recommendations on the design of future ultralightweight protocols.

2 Yeh et al. Protocol

Yeh et al. proposed a process-oriented ($flag = 0$ or $flag = 1$) ultralightweight RFID authentication protocol for very low-cost RFID tags. Simple bitwise operations such as AND, OR, XOR, addition mod 2^m and circular shift rotations[1]

[1] We confirmed by personal communication with one of the authors that the most common definition of rotation is used. That is, $Rot(X, Y) = X << (Y \mod m)$, where m is the bit length of the variables.

are the set of operations assumed to be supported on-chip by tags. The authors assume that m is the bit length of all the variables in the scheme[2]. Both backward and forward channels are exposed to passive attacks. On the other hand, the communication channel between the reader and the back-end database is assumed to be secure.

In the initialization process, three values are stored in the tag's memory: 1) an authentication key K_t; 2) index-pseudonym IDS_t; and 3) an unique static identifier ID_t. Correspondingly, the back-end database maintains four values: 1) an authentication key K_{tr}; 2-3) two index pseudonyms to avoid desynchronization attacks $\{IDS_{tr_{old}}, IDS_{tr_{new}}\}$; 4) a static identifier ID_{tr}.

The protocol considers two different situations depending on the success/fail of completion of the previous protocol session ($flag = 0$ / $flag = 1$). The protocol works as follows:

$Reader \rightarrow Tag : Hello$ The reader sends to the tag a request message.
$Tag \rightarrow Reader : IDS_t$ The tag replies to the reader with its index-pseudonym.
$Reader \rightarrow Tag : A||B||C||flag$ The reader uses the received IDS_t as a search index to allocate all the information linked to an specific tag. The authentication key of the current session is set depending on the succesful/failed completion of the previous session:

$$\begin{cases} \text{if } (IDS = IDS_{tr_{new}}) & K = K_{tr} \quad \text{and} \quad flag = 0 \\ \text{if } (IDS = IDS_{tr_{old}}) & K = ID_{tr} \quad \text{and} \quad flag = 1 \end{cases} \tag{1}$$

The reader generates two nonces $\{n_1, n_2\}$, computes messages $\{A, B, C\}$ and sends this tuple to the tag. Messages $\{A, B, C\}$ are defined by:

$$\text{Compute:} \quad A = (IDS \oplus K) \oplus n_1 \tag{2}$$
$$B = (IDS \vee K) \oplus n_2 \tag{3}$$
$$C = (\overline{K} \oplus n_1) + n_2 \tag{4}$$
$$\overline{K} = Rot(K \oplus n_2, n_1) \tag{5}$$

$Tag \rightarrow Reader : D$ Upon receiving the tuple $\{A||B||C||flag\}$, the tag sets the value D depending on the flag. It extracts the nonces $\{n_1, n_2\}$ from A and B. Then, the correctness of C is checked:

$$\begin{cases} \text{if } (flag = 0) & K = K_t \\ \text{if } (flag = 1) & K = ID_t \end{cases} \tag{6}$$

$$\text{Compute:} \quad \overline{K} = Rot(K \oplus n_2, n_1) \tag{7}$$
$$\overline{C} = (\overline{K} \oplus n_1) + n_2 \tag{8}$$
$$\text{Verify:} \quad \overline{C} = C \tag{9}$$

[2] In our experimentation, we assume that the bit length of the variables used in the protocol is 96 ($m = 96$). If fact this is one of the most common bit length values for the static identifier ID of a tag (e.g GID-96, SGTIN-96, GIAI-96, etc. [17])

If the reader is authenticated ($\overline{C} = C$), the tag computes the values $\{\overline{K'}, D\}$ and sends message D to the reader.

$$\overline{K'} = Rot(K \oplus n_1, n_2) \tag{10}$$
$$D = (\overline{K'} \oplus n_2) + n_1 \tag{11}$$

Reader Updating On receiving D, the reader checks its correctness. If so, the tag is authenticated and the reader updates its internal values:

$$\text{Compute:} \quad \overline{K'} = Rot(K \oplus n_1, n_2) \tag{12}$$
$$\overline{D} = (\overline{K'} \oplus n_2) + n_1 \tag{13}$$
$$\text{Verify:} \quad \overline{D} = D \tag{14}$$

Updating phase:

$$\text{If } \overline{D} = D, \text{ compute:} \quad IDS_{tr_{old}} = IDS \tag{15}$$
$$IDS_{tr_{new}} = (IDS + (ID \oplus \overline{K'})) \oplus n_1 \oplus n_2 \tag{16}$$
$$K_{tr} = \overline{K} \tag{17}$$

Otherwise, the protocol is aborted.

Finally, the reader sends an *Update command* to the tag.

Tag Updating Upon receiving the *Update command*, the update phase of the internal values is executed:

$$\text{Compute:} \quad IDS = (IDS + (ID \oplus \overline{K'})) \oplus n_1 \oplus n_2 \tag{18}$$
$$K_t = \overline{K} \tag{19}$$

3 Traceability Attack

Within the untraceability (UNT) model [14], tags ($\mathcal{T}$) and readers ($\mathcal{R}$) interact in protocol sessions, while the adversary ($\mathcal{A}$) is assumed to control the communications between all parties. In order to model $\mathcal{A}$'s capabilities, the following oracle queries are defined:

- Execute($\mathcal{R}$, $\mathcal{T}$, i) query. This models a passive attacker. $\mathcal{A}$ eavesdrops on the channel, and gets read access to the exchanged messages between $\mathcal{R}$ and $\mathcal{T}$ in session i of a genuine protocol execution.
- Test(i, $\mathcal{T}_0$, $\mathcal{T}_1$) query. This is defined to simplify the modelling of the untraceability notion. When this query is invoked for test session i, a random bit is generated $b \in \{0, 1\}$. Then, a pseudonym IDS^i corresponding to either of $\{ID^{\mathcal{T}_0}, ID^{\mathcal{T}_1}\}$ depending on the bit b is given to $\mathcal{A}$.

The untraceability (UNT) notion is then defined as a game $\mathcal{G}$ comprising three phases:

Phase 1 (Learning): $\mathcal{A}$ can make any number of $\mathsf{Execute}$ queries, which model the eavesdropping of exchanged messages, i.e. a passive attack, over the insecure radio channel.

Phase 2 (Challenge): $\mathcal{A}$ chooses two fresh tags whose associated identifiers are $ID^{\mathcal{T}_0}$ and $ID^{\mathcal{T}_1}$. Then he sends a $\mathsf{Test}(i,\ \mathcal{T}_0,\ \mathcal{T}_1)$ query. As result, $\mathcal{A}$ is given a pseudonym IDS^i corresponding to either of $\{ID^{\mathcal{T}_0}, ID^{\mathcal{T}_1}\}$ depending on a randomly chosen bit $b \in \{0,1\}$.

Phase 3 (Guessing) $\mathcal{A}$ ends the game and outputs a bit $\tilde{b} \in \{0,1\}$ as its guess of the value of b.

$\mathcal{A}$'s success in winning $\mathcal{G}$ is equivalent to the success of breaking the untraceability property offered by the protocol. Thus the advantage of $\mathcal{A}$ in distinguishing whether the pseudonym corresponds to $\mathcal{T}_0$ or $\mathcal{T}_1$ is defined below, where t is a security parameter (i.e. the bit length of the key shared between the tag and the reader) and r is the number of times $\mathcal{A}$ runs an $\mathsf{Execute}$ query.

$$Adv_{\mathcal{A}}^{\mathsf{UNT}}(t,r) = |Pr[\tilde{b} = b] - \frac{1}{2}| \tag{20}$$

So, an RFID protocol offers resistance against traceability if $Adv_{\mathcal{A}}^{\mathsf{UNT}}(t,r) < \varepsilon(t,r)$, where $\varepsilon(\cdot,\cdot)$ symbolizes some negligible function.

We now show that the RFID authentication protocol by Yeh et al. does not achieve untraceability (UNT). The adversary mounts the attack as follows:

Phase 1 (Learning): $\mathcal{A}$ issues an $\mathsf{Execute}(\mathcal{R},\ \mathcal{T}_0,\ i)$ query, thereby obtaining $\langle IDS^i, A^i, B^i, C^i, flag^i, D^i\rangle$.

Phase 2 (Challenge): $\mathcal{A}$ chooses two fresh tags whose associated identifiers are $ID^{\mathcal{T}_0}$ and $ID^{\mathcal{T}_1}$, where $\mathsf{lsb}(ID^{\mathcal{T}_1}) = \neg\mathsf{lsb}(ID^{\mathcal{T}_0})$, $\mathsf{lsb}(\cdot)$ denotes the least significant bit and $\neg x$ symbolizes the bitwise NOT of x. Then he sends a $\mathsf{Test}(i+1,\ \mathcal{T}_0,\ \mathcal{T}_1)$ query. As result, $\mathcal{A}$ is given a new pseudonym IDS^{i+1} corresponding to either of $\{ID^{\mathcal{T}_0}, ID^{\mathcal{T}_1}\}$ depending on a chosen random bit $b \in \{0,1\}$.

Phase 3 (Guessing) $\mathcal{A}$ computes $d = \mathsf{lsb}(IDS^{i+1} \oplus IDS^i \oplus D^i)$. It sets $\tilde{b} = 0$ if $d = \mathsf{lsb}(ID^{\mathcal{T}_0})$; else $\tilde{b} = 1$. Then $\mathcal{A}$ ends $\mathcal{G}$ and outputs a bit $\tilde{b}$ as its guess of the value b.

We now analyze the success probability of the adversary in winning the game. The adversary computes a bit d which is the least significant bit (lsb) of the value

$$IDS^{i+1} \oplus IDS^i \oplus D^i = ((IDS^i + (ID \oplus \overline{K'}^i)) \oplus n_1^i \oplus n_2^i) \oplus IDS^i \oplus (\overline{K'}^i \oplus n_2^i + n_1^i) \tag{21}$$

Since we are dealing only with the lsb, thus XOR equals addition ($+$), so equation (21) becomes

$$\begin{aligned}\mathsf{lsb}(IDS^{i+1} \oplus IDS^i \oplus D^i) &= \mathsf{lsb}(((IDS^i \oplus ID \oplus \overline{K'}^i) \oplus n_1^i \oplus n_2^i) \oplus \\ &\qquad IDS^i \oplus (\overline{K'}^i \oplus n_2^i \oplus n_1^i)) \\ &= \mathsf{lsb}(ID),\end{aligned} \tag{22}$$

and thus we have

$$d = \mathsf{lsb}(ID), \tag{23}$$

where ID is the static identifier which is either of $\{ID^{\mathcal{T}_0}, ID^{\mathcal{T}_1}\}$ depending on the bit b. So the adversary just checks the $\mathsf{lsb}(IDS^{i+1} \oplus IDS^i \oplus D^i)$ to determine if it is $ID^{\mathcal{T}_0}$ or $ID^{\mathcal{T}_1}$, thus it wins the game with probability 1.

Hence we have

$$Adv_{\mathcal{A}}^{\mathsf{UNT}}(t,1) = |Pr[\tilde{b} = b] - \frac{1}{2}| = 1 - \frac{1}{2} = \frac{1}{2} > \varepsilon. \tag{24}$$

4 Full Disclosure Norwegian Attack

RFID tags have a static identifier ID that facilitates the unequivocally identification of labeled items. RFID protocols should transmit this value in a secure way (e.g. after an unilateral or mutual authentication protocol) to avoid traceability attacks. In this section, we present a passive attack able of recovering $\lfloor \log_2 L \rfloor$ bits of the ID after eavesdropping $O(L)$ legitimate authentication sessions.

In Yeh et al.'s protocol, the authors use two random numbers $\{n_1, n_2\}$ to guarantee the freshness of each session – among other security objectives. Nevertheless, the proposed protocol slightly abuses the usage it makes of these nonces, and misuses how these are computed. We can analyze what happens to the protocol when these two nonces happen to have the same value module L, i.e.:

$$n_1 \mod L = n_2 \mod L \tag{25}$$

Here L should be a power of two. However, we will later show that all these equations probabilistically hold for any positive integer. (We shall provide some guidelines regarding how to choose L later on this section.) Under the assumption that equation 25 holds, we can probabilistically greatly simplify the index-pseudonym updating equation (Equation 16). More precisely, the least significant bits of the last two terms are canceled out ($n_1 \oplus n_2 \mod L = 0$):

$$\begin{aligned} IDS_{tr_{new}} \mod L &= (IDS + (ID \oplus \overline{K'})) \oplus n_1 \oplus n_2 \mod L \\ &= (IDS + (ID \oplus \overline{K'})) \mod L \end{aligned} \tag{26}$$

If, on the other hand, the public message D is examined and we approximate addition by XOR:

$$\begin{aligned} D \mod L &= (\overline{K'} \oplus n_2) + n_1 \mod L \\ &\simeq \overline{K'} \oplus n_2 \oplus n_1 \mod L = \overline{K'} \mod L \end{aligned} \tag{27}$$

Combining Equations 26 and 27, and working out the value of ID, we get an approximation where only public messages transmitted on the insecure radio channel are involved:

$$ID \mod L = (IDS_{tr_{new}} - IDS) \oplus D \mod L \tag{28}$$

The only remaining question is how to recognize when the condition $n_1 \mod L = n_2 \mod L$ holds, since $\{n_1, n_2\}$ are secret values. From Equations 5 and 10, it is relatively straightforward to deduce that the above mentioned condition implies $\overline{K} \mod L = \overline{K'} \mod L$. The next step is the correlation of this condition with some values or test on the public exchanged messages transmitted over the channel. Specifically, we can use the approximation of the sum by the XOR operation in messages C and D, and finally compare these values:

$$C \mod L = (\overline{K} \oplus n_1) + n_2 \mod L \simeq \overline{K} \oplus n_1 \oplus n_2 \mod L \quad (29)$$
$$\simeq \overline{K} \mod L$$
$$D \mod L = (\overline{K'} \oplus n_1) + n_2 \mod L \simeq \overline{K'} \oplus n_1 \oplus n_2 \mod L$$
$$\simeq \overline{K'} \mod L \quad (30)$$

So, by comparing the values of public messages C and D ($mod\ L$), we are able to probabilistically detect the condition that opens the door to the disclosure of the static identifier of the tag by passively eavesdropping on the channel. However, our testing condition $C \mod L = D \mod L$ may hold just by pure chance while $n_1 \mod L \neq n_2 \mod L$ does not. As a consequence of this, we have to filter and analyze the results to obtain the pursued value of $ID \mod L$. Basically, we repeat this process many times to obtain different candidates for the $ID \mod L$ value, we count the number of times each of these values is observed and pick the maximum as our guess of the static identifier. We sketch the steps of the Norwegian attack below:

1. For $i = 0$ to L
2. $\quad Observations[i] = 0$
3. Repeat a sufficiently high number of times N the following steps:
4. $\quad$ Observe an authentication session and get IDS, A, B, C and D
5. $\quad$ Check if for these values it holds that $C \mod L = D \mod L$
6. $\quad$ If this is not the case, go to step 4.
7. $\quad$ Perform the following tasks:
8. $\quad\quad$ Wait for the authentication session to finish.
9. $\quad\quad$ Send to the tag a "Hello" message to obtain $IDS_{tr_{new}}$.
10. $\quad\quad$ Compute $ID_{estimated} \mod L = (IDS_{tr_{new}} - IDS) \oplus D \mod L$
11. $\quad\quad$ Increment $Observations[ID_{estimated}]$
12. Filter: find $ID_{conjecture}$, the maximum of the values in $Observations[i]$.
13. Guess that $ID_{conjecture} = ID \mod L$.

To further clarify the Norwegian attack, in Appendix A (Figure 2) we display an example of the *Observations* vector obtained for $L = 128$ and $N = 2^{18}$.

Finally, in Appendix B (Figure 3) we can observe, for several values of L, the adversary's success probability depending on the number of eavesdropped sessions. Although the attack just presented can be run independently for any value of L, it is highly recommended to select one which is a power of 2 (the probabilistic equations presented before hold with greater probabilities) in order

to have a higher success probability and to minimize the number of snooped sessions. An interesting point of the Norwegian attack is its success regardless of the rotation definition used. In other words, the attack is feasible even if the Hamming weight based rotation[3] is used.

The main drawback of the proposed attack in that the number of eavesdropped sessions needed to recover the whole value of the ID may be excessive. Nevertheless, the knowledge of some bits of the static identifier is informative enough to conduct a successful traceability attack. In fact, only one bit is required in the formal privacy model introduced in [14]. In case that we need to recover the whole ID, the attack described in the following Section 5 is much more convenient.

5 Full Disclosure Tango Attack

In [2], a new technique reminiscent of Linear Cryptanalysis, named Tango attack, is introduced. In this section, we present a very efficient passive attack against Yeh et al.'s scheme, based on Tango cryptanalysis principles. We emphasize here that despite of residing in the same bases, we need an extra twist in the aforementioned technique to success in our attack. More precisely, we use a non-linear approach instead of the completely linear approach used in [2].

The proposed attack reveals the most valuable information stored on the tags memory, the static identifier ID, which is the information the protocol was built to protect. The main singularity of the Tango attack compared with the Norwegian attack presented in the previous section is its much higher efficiency and devastating consequences – from a security point of view – for Yeh et al.'s protocol. The eavesdropping of a very small number of authentication sessions in this case is enough to reveal the complete ID.

Before presenting the inner details of our attack, we first sketch its general approach. Variables can be represented in a m-dimensional space instead of considering them as numerical values. Recall that m is the bit length of variables used in the protocol. More precisely, if a variable z is represented in binary format, the coefficients a_i are the values of the vector Z in each dimension:

$$\begin{aligned} z &= \sum_{i=0}^{m-1} a_i \cdot 2^i, \ a_i \in \{0,1\} \\ Z &= [a_0 \ a_1 \cdots a_{m-1}] \end{aligned} \tag{31}$$

The attacker follows some simple steps. Firstly, she eavesdrops an authentication session, computes an approximation of the static identifier as a function of the observed messages, and stores this vector:

$$ID_{approx} = f(IDS(k), A, B, C, D, IDS(k+1)) \tag{32}$$

[3] $Rot(X, Y) = X << wt(Y)$, where $wt(Y)$ stands for the Hamming weight of vector Y.

The above step is repeated during N eavesdropped sessions. Then, the attacker combines (in our proposal, he simply adds up) all the vectors obtained in this way, and an average value of this resulting vector becomes the conjectured static identifier $ID_{conjecture}$. We provide a numerical example for clarification purposes. For simplicity, we set $m = 8$ in the example.

- **Session k:**
  Eavesdropping of vectors $\{IDS(k), A, B, C, D, IDS(k+1)\}$
  Computing of an approximation: i.e. $ID_{approx}(1) = [0\ 1\ 0\ 1\ 1\ 1\ 1\ 1]$
- **Session $k+1$:**
  Eavesdropping of vectors $\{IDS(k+1), A', B', C', D', IDS(k+2)\}$
  Computing of an approximation: i.e. $ID_{approx}(2) = [0\ 1\ 0\ 1\ 0\ 1\ 0\ 0]$
- **Session $k+2$:**
  Eavesdropping of vectors $\{IDS(k+2), A'', B'', C'', D'', IDS(k+3)\}$
  Computing of an approximation: i.e. $ID_{approx}(3) = [0\ 1\ 1\ 0\ 0\ 1\ 0\ 1]$
- **Conjecture ID:**

Sum of the vectors:

$$\begin{array}{rl} & [0\ 1\ 0\ 1\ 1\ 1\ 1\ 1] \\ & [0\ 1\ 0\ 1\ 0\ 1\ 0\ 0] \\ & [0\ 1\ 1\ 0\ 0\ 1\ 0\ 1] \\ + & \hline \\ ID_{approx} = & [0\ 3\ 1\ 2\ 1\ 3\ 1\ 2] \end{array}$$

Average value:

$$\begin{cases} \text{if } (id_i^{approx} \geq \gamma) & id_i^{conjecture} = 1 \\ \text{if } (id_i^{approx} < \gamma) & id_i^{conjecture} = 0 \end{cases}$$

i.e. If $\gamma = 1.5$ $\quad ID_{conjecture} = [0\ 1\ 0\ 1\ 0\ 1\ 0\ 1]$

Conjecture: $\quad ID_{conjecture}(base10) = 85$

Of course, much more complex combinations of the different approximations, and more elaborate filters are possible, but for the protocol at hand this approach works exceedingly well so we do not feel justified to introduce any additional complexity into the attack.

Now, we provide the details of the Tango based attack. We start with the search of good approximations to the static identifier ID. Basically, the attacker captures all the public messages exchanged over the insecure radio channel and combines these values to compute approximations for ID. Of course, not all combinations produce good results. Only those that are closer (on average) to the static identifier are useful. The Hamming weight can be used as an effective (but not the only) metric to evaluate the quality of an approximation. More precisely, if the average Hamming weight between an approximation X and the target value ID is below $\frac{m}{2}$, X is a good approximation:

$$\begin{cases} \text{if } \langle wt(X, ID) \rangle < m/2 & X \text{ is a good approximation} \\ \text{if } \langle wt(\neg X, ID) \rangle < m/2 & \neg X \text{ is a good approximation} \\ \text{Otherwise} & X \text{ is ruled out} \end{cases}$$

where $\langle \cdot \rangle$ denotes the average value and $\neg x$ symbolizes the bitwise NOT of x. In Appendix C (Table 1) we summarize the results obtained by all possible

combinations of the public exchanged messages as approximations of the ID. Unfortunately, none of these approximations – contrary to what happens in [2] – constitutes a good approximation of the static identifier. In all cases, the Hamming weight obtained is so close to the optimal value (i.e. $m/2 = 48$) that the alternate hypothesis (the approximation under scrutiny does not leak any useful information about the ID) cannot be rejected. This seems to be quite a powerful result in showing that the protocol is well though-off and not easy to crack by any linear approximation. One can easily be tempted to believe that there is no information leakage in the public messages as Yeh et al. claim themselves. Nevertheless, an additional twist can shed more light on this issue. We have to carefully analyze the updating equation for the index-pseudonym:

$$IDS_{tr_{new}} = (IDS + (ID \oplus \overline{K'})) \oplus n_1 \oplus n_2 \tag{33}$$

This is the only message in which the ID takes part. We can work out this variable from the above equation,

$$ID = ((IDS_{tr_{new}} \oplus n_1 \oplus n_2) - IDS) \oplus \overline{K'} \tag{34}$$

In a slightly more elaborate approach to the problem, we can try to approximate individually the different unknown components of the above equation, instead of doing all globally in a single step. IDS and $IDS_{tr_{new}}$ are the current index-pseudonym and the potential new index-pseudonym. If the adversary eavesdrops two consecutive legitimate sessions, these values are thus known since they are transmitted in the clear on the channel. So, the adversary has to find good approximations for $n_1 \oplus n_2$ and K'. By combining messages A and B a good approximation of the XOR between the nonces n_1 and n_2 can be obtained as shown below. K' can be approximated by using the above equation and the public message $D = (\overline{K'} \oplus n_2) + n_1$.

$$\begin{array}{ll} n1 \oplus n2 \simeq A \oplus B & \langle wt(A \oplus B, n1 \oplus n2)\rangle \quad = 23.9411 \pm 4.2505 \\ \overline{K'} \simeq \neg(D + (A \oplus B)) & \langle wt(\neg(D + (A \oplus B)), \overline{K'})\rangle = 40.4185 \pm 5.2096 \end{array}$$

From all the above, we have an approximation for the ID value which only involves public values:

$$ID_{approx} = ((IDS_{tr_{new}} \oplus A \oplus B) - IDS) \oplus (\neg(D + (A \oplus B))) \tag{35}$$

This is enough to mount a powerful Tango attack. From here on, the attacker simply eavesdrops a session and the new index-pseudonym of the new session, computes an approximation of $ID_{approx}(i)$ using Equation 35 and finally stores this vector. Once the adversary has eavesdropped N sessions, the sum of all the approximations is computed:

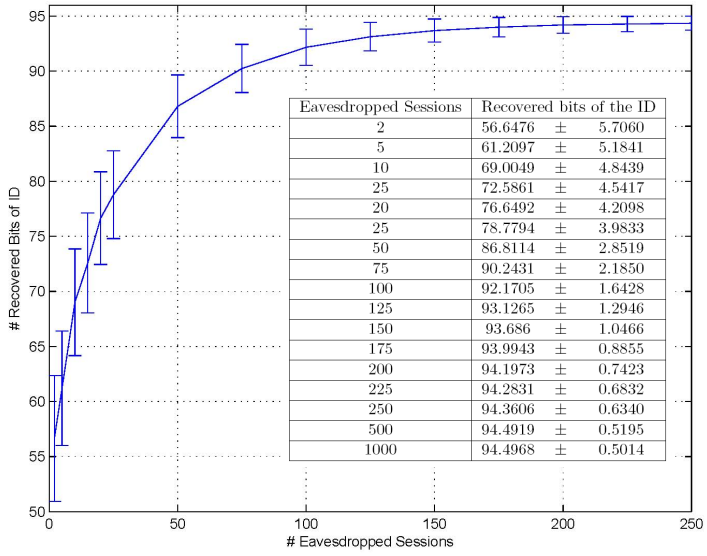
$$ID_{approx} = \sum_{i=1}^{N} ID_{approx}(i) \tag{36}$$

The only remaining question is to obtain the average value of the above vector (i.e. $ID_{conjecture} = g(ID_{approx})$). We propose using the following g function which is simple but entirely effective. The components of the input and output vector in each axis are denoted by id_i^{approx} and $id_i^{conjecture}$, respectively. The parameter γ is set to $\frac{N}{2}$.

$$For\ i = 0, m-1 \begin{cases} \text{if } (id_i^{approx} \geq \gamma) & id_i^{conjecture} = 1 \\ \text{if } (id_i^{approx} < \gamma) & id_i^{conjecture} = 0 \end{cases} \tag{37}$$

Finally, the attacker concludes $ID_{conjecture} = \sum_{i=0}^{m-1} id_i^{conjecture} \cdot 2^i$ as its conjecture of the static identifier ID.

To evaluate the effectiveness of our Tango attack, we have ran several simulations. First, we randomly initialize the secret values $\{K, ID, IDS_{old}, IDS_{new}\}$ stored in the tag and the back-end database. Then, we simulate N sessions of the protocol and follow the algorithm just described. To measure the adversary's success, we compare the conjecture value $ID_{conjecture}$ with the real value of the target ID. For each value of N, we repeat the experiment $10,000$ times[4].



Eavesdropped Sessions	Recovered bits of the ID
2	56.6476 ± 5.7060
5	61.2097 ± 5.1841
10	69.0049 ± 4.8439
25	72.5861 ± 4.5417
20	76.6492 ± 4.2098
25	78.7794 ± 3.9833
50	86.8114 ± 2.8519
75	90.2431 ± 2.1850
100	92.1705 ± 1.6428
125	93.1265 ± 1.2946
150	93.686 ± 1.0466
175	93.9943 ± 0.8855
200	94.1973 ± 0.7423
225	94.2831 ± 0.6832
250	94.3606 ± 0.6340
500	94.4919 ± 0.5195
1000	94.4968 ± 0.5014

Fig. 1. Adversary success in recovering ID

In Figure 1, we display the mean and standard deviation of the number of bits successfully recovered for various values of eavesdropped sessions. The

[4] The code of this attack can be downloaded from: `http://www.lightweightcryptography.com/research/ywl/ywl.html`

attack is quite effective and only a low number of eavesdropped sessions are required to disclosure a significant part of the static identifier. More precisely, if the attacker eavesdrops $\{2, 15, 50, 100\}$ sessions, $\{50, 75, 90, 95\}\%$ of the 96-bits ID are disclosed. The threshold of our attack is the recovery of 94 bits of the static identifier. The two least significant bits of the static identifer are not obtained by the adversary, even for very high values of eavesdropped sessions. Nevertheless, the attacker has reduced the search of the static identifier from 2^{96} to 2^2 candidates. Additionally, we can use the Norwegian attack *in parallel* as described before in this same article for recovering these two bits very efficiently (in less than 100 sessions).

6 Conclusions

At the start of 2010, Yeh, Lo and Winata proposed a new ultralightweight authentication protocol. The security analysis carried out by the designers was mainly based on evaluating the randomness of the messages exchanged over the insecure radio channel. Basically, once the internal secret values of the tag (and back-end database) are randomly initialized, the execution of the protocol is simulated for a large number of sessions. The generated messages $\{IDS, A, B, C, D\}$ during each session are stored in a file. Finally, this file is exposed to a battery of statistical tests (i.e. NIST test suite). Yeh et al. concluded that messages looked sufficiently random since the file passed all the test at hand. This randomness based study is certainly an interesting analysis, but unfortunately randomness of the exchanged messages is neither a sufficient nor necessary condition for protocol security.

In this paper we explicitly show that a good degree of randomness in the public messages does not guarantee by itself the security of the protocol. In fact, we show how even a passive attacker is able to disclose the full static identifier of the tag by simply combining wisely some public messages – passed over the insecure radio channel – and using both the Norwegian and the Tango attack.

Apart from the cryptanalytic results on the Yeh et al. protocol, we believe that the Tango attack can be very useful for the analysis and design of new ultra lightweight protocols, as we have shown it is quite powerful and efficient. The only almost negligible limitation of not being able to retrieve the full ID but only 94 out of 96 bits instead can easily be solved by its combined utilization together with the Norwegian attack (i.e. with $L = 4$), thus becoming nicely complementary attacks.

If we had to point out the design mistakes that led the Yeh et al. protocol to this full disclosure attack we should say that, as already shown in the literature [6], Hamming weight based rotations seem to generally provide more secure proposals. So choosing circular shift rotations instead, while not being a major mistake can certainly be considered suboptimal. Nevertheless, the Tango attack introduced in this paper is independent of the definition of rotation used (e.g. hamming weight or circular shift rotations). Additionally, the key K is too exposed in messages A and B, which have a striking similitude that almost com-

pletely leaks out the value of $n_1 \oplus n_2$. The value of $\overline{K'}$ should probably depend on that of K' (not of K only) to increase its strength. And again, while being aesthetically pleasing the construction of messages D and C are so symmetric that, as we have shown, they leak too much information.

As future work, we will continue to analyze new proposals in the light of these cryptanalysis techniques, and use them to motivate new design criteria.

References

1. K.-H. Yeh, N.W. Lo, E. Winata. "An Efficient Ultralightweight Authentication Protocol for RFID Systems". *Proc. of RFIDSec Asia'10*, volume 4 of *Cryptology and Information Security Series*, pages 49–60, IOS Press, 2010.
2. J. C. Hernandez-Castro, P. Peris-Lopez, Raphael C.-W. Phan, J. M. E. Tapiador. "Cryptanalysis of the David-Prasad RFID Ultralightweight Authentication Protocol". *Proc. of Workshop on RFID Security'10*, 2010.
3. 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. In *Hand. of Workshop on RFID and Lightweight Crypto*, 2006.
4. P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estevez-Tapiador, and A. Ribagorda. M2AP: A minimalist mutual-authentication protocol for low-cost RFID tags. In *Proc. of UIC'06*, volume 4159 of *LNCS*, pages 912–923. Springer-Verlag, 2006.
5. P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estevez-Tapiador, and A. Ribagorda. EMAP: An efficient mutual authentication protocol for low-cost RFID tags. In *Proc. of IS'06*, volume 4277 of *LNCS*, pages 352–361. Springer-Verlag, 2006.
6. J. C. Hernandez-Castro, J. E. Tapiador, P. Peris, T. Li, J.-J. Quisquater Cryptanalysis of the SASI Ultralightweight RFID Authentication Protocol with Modular Rotations. In *Proc. of WCC'09*, May 10-15, 2009.
7. T. Li and G. Wang. Security analysis of two ultra-lightweight RFID authentication protocols. In *Proc. of IFIP-SEC'07*, 2007.
8. H. Y. Chien and C.-W. Huang. Security of ultra-lightweight RFID authentication protocols and its improvements. *SIGOPS Oper. Syst. Rev.* 41(4):83–86, 2007.
9. M. Bárász, B. Boros, P. Ligeti, K. Lója, and D. Nagy. "Breaking LMAP", *Proc. of RFIDSec'07*, 2007.
10. M. Bárász, B. Boros, P. Ligeti, K. Lója, and D. Nagy. "Passive attack against the M2AP mutual authentication protocol for RFID tags", *Proc. of First International EURASIP Workshop on RFID Technology*, 2007.
11. H.-Y. Chien. "SASI: A new ultralightweight RFID authentication protocol providing strong authentication and strong integrity". *IEEE Transactions on Dependable and Secure Computing* 4(4):337–340. Oct.-Dec. 2007.
12. T. Cao, E. Bertino, and H. Lei. "Security Analysis of the SASI Protocol". *IEEE Transactions on Dependable and Secure Computing* 6(1):73–77. Jan.-Mar. 2009.
13. P. D'Arco and A. De Santis. "Weaknesses in a Recent Ultra-Lightweight RFID Authentication Protocol". In *Proc. of AFRICACRYPT'08*, volume 5023 of *LNCS*, pages 27–39. Springer-Verlag, 2008.
14. R. Phan. Cryptanalysis of a new ultralightweight RFID authentication protocol - SASI. *IEEE Transactions on Dependable and Secure Computing* 6(4):316–320. Oct.-Dec. 2009.

15. P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estevez-Tapiador, and A. Ribagorda. Advances in Ultralightweight Cryptography for Low-cost RFID Tags: Gossamer Protocol. In *Proc. of WISA'08*, Volume 5379 of *LNCS*, pages 56-68. Springer-Verlag, 2008.
16. Kuo-Hui Yeh and N. W. Lo. "Improvement of Two Lightweight RFID Authentication Protocols". *Information Assurance and Security Letters*. In Press (2010).
17. GS1 EPCglobal. EPCglobal Tag Data Standards Version 1.4. http://www.epcglobalinc.org/standards/, Ratified on June 2008.

APPENDIX

A Algorithm – Norwegian Attack

In Figure 2, we display an example of the *Observations* vector obtained for $L = 128$ and $N = 2^{18}$. By simply inspection of the above figure, we can easily detect a pick and correctly conjecture this value is the target value – in the example $ID \mod L = ID_{conjecture} \mod L = 21$. It is interesting to observe that half of the times the *Observations* vector has a zero value, and that most of the other peaks occur at values that share the least significant bits with the real value (i.e. $21 + 64 = 85$, $21 + 64 + 32 = 117$).

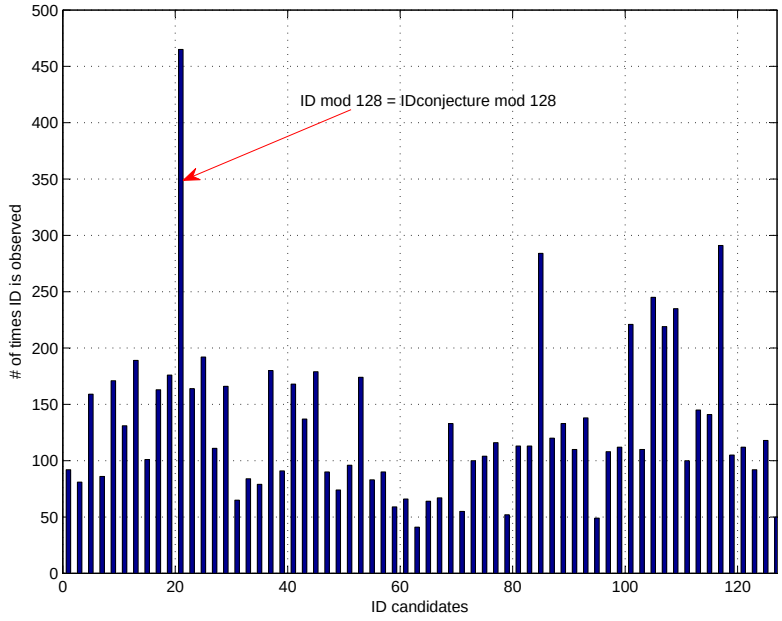


Fig. 2. Histogram of ID candidates ($L = 128$, $N = 2^{18}$)

B Adversary Success's Probability – Norwegian Attack

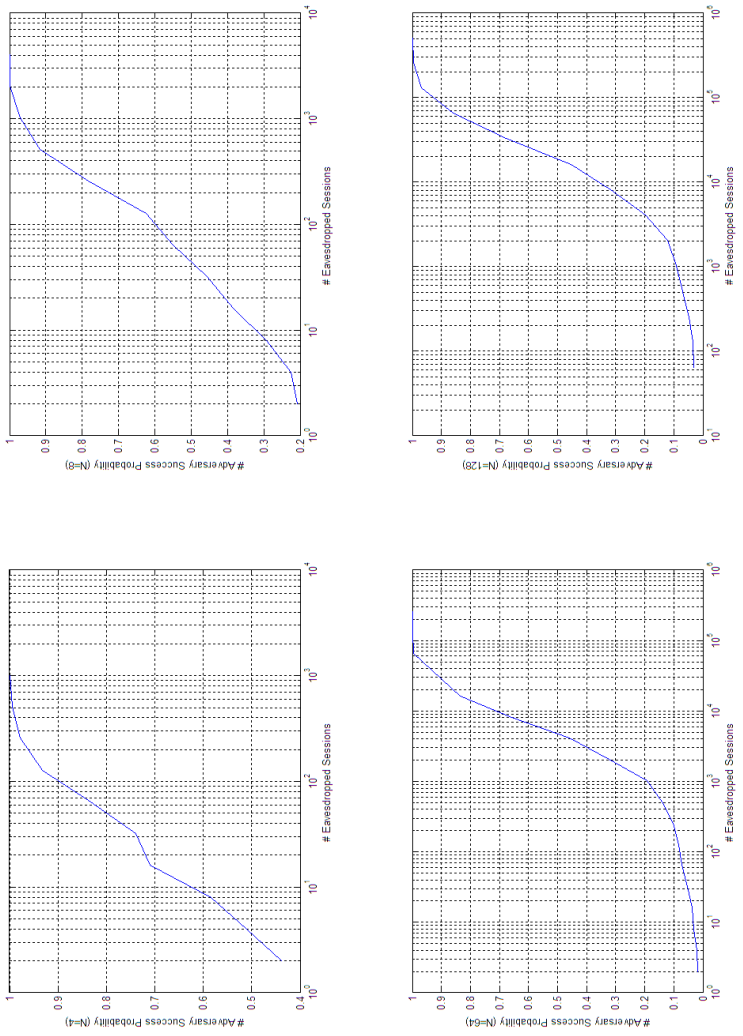


Fig. 3. Adversary success probability for various values of L ($L = \{4, 8, 64, 128\}$)

C Approximations to the ID – 10.000 Experiments

$\mathbf{X}$	$\overline{\mathbf{wt}}(\mathbf{X} \oplus \mathbf{ID})$
A	47.9473 ± 4.9481
B	48.0286 ± 4.9290
C	47.9155 ± 4.9111
D	47.8964 ± 4.8949
$IDS_{tr_{old}}$	48.0107 ± 4.9459
$IDS_{tr_{new}}$	48.0115 ± 4.9452
$A \oplus B$	47.9671 ± 4.8567
$A \oplus C$	48.0044 ± 4.9504
$A \oplus D$	48.0557 ± 4.8726
$A \oplus IDS_{tr_{old}}$	47.9872 ± 4.9812
$A \oplus IDS_{tr_{new}}$	48.0166 ± 4.9565
$B \oplus C$	47.9951 ± 4.8832
$B \oplus D$	48.0074 ± 4.9196
$B \oplus IDS_{tr_{old}}$	48.0009 ± 4.8799
$B \oplus IDS_{tr_{new}}$	47.9677 ± 4.9498
$C \oplus D$	47.9513 ± 4.9492
$C \oplus IDS_{tr_{old}}$	47.9710 ± 4.8698
$C \oplus IDS_{tr_{new}}$	47.9370 ± 4.8724
$D \oplus IDS_{tr_{old}}$	47.9303 ± 4.9247
$D \oplus IDS_{tr_{new}}$	48.0183 ± 4.8454
$IDS_{tr_{old}} \oplus IDS_{tr_{new}}$	47.9936 ± 4.9573
$A \oplus B \oplus C$	48.0768 ± 4.8960
$A \oplus B \oplus D$	48.0815 ± 4.9432
$A \oplus B \oplus IDS_{tr_{old}}$	47.9438 ± 4.8561
$A \oplus B \oplus IDS_{tr_{new}}$	48.0544 ± 4.9069
$A \oplus C \oplus D$	48.0350 ± 4.9326
$A \oplus C \oplus IDS_{tr_{old}}$	47.9515 ± 4.9163
$A \oplus C \oplus IDS_{tr_{new}}$	47.9745 ± 4.8637
$A \oplus D \oplus IDS_{tr_{old}}$	47.9592 ± 4.8990
$A \oplus D \oplus IDS_{tr_{new}}$	48.0550 ± 4.9093
$A \oplus IDS_{tr_{old}} \oplus IDS_{tr_{new}}$	47.9939 ± 4.9177
$B \oplus C \oplus D$	47.9697 ± 4.8648
$B \oplus C \oplus IDS_{tr_{old}}$	47.9726 ± 4.8819
$B \oplus C \oplus IDS_{tr_{new}}$	47.9550 ± 4.9445
$B \oplus D \oplus IDS_{tr_{old}}$	48.0133 ± 4.9098
$B \oplus D \oplus IDS_{tr_{new}}$	47.9009 ± 4.8951
$B \oplus IDS_{tr_{old}} \oplus IDS_{tr_{new}}$	48.0706 ± 4.9132
$C \oplus D \oplus IDS_{tr_{old}}$	48.0520 ± 4.9140
$C \oplus D \oplus IDS_{tr_{new}}$	47.9936 ± 4.9111
$C \oplus IDS_{tr_{old}} \oplus IDS_{tr_{new}}$	47.9995 ± 4.8690
$D \oplus IDS_{tr_{old}} \oplus IDS_{tr_{new}}$	47.2532 ± 6.2162
$A \oplus B \oplus C \oplus D$	48.0122 ± 4.8277
$A \oplus B \oplus C \oplus IDS_{tr_{old}}$	47.9755 ± 4.9338
$A \oplus B \oplus C \oplus IDS_{tr_{new}}$	47.9941 ± 4.8607
$A \oplus B \oplus D \oplus IDS_{tr_{old}}$	47.9482 ± 4.9107
$A \oplus B \oplus D \oplus IDS_{tr_{new}}$	48.0028 ± 4.8607
$A \oplus B \oplus IDS_{tr_{old}} \oplus IDS_{tr_{new}}$	48.0869 ± 4.8411
$A \oplus C \oplus D \oplus IDS_{tr_{old}}$	48.0133 ± 4.8971
$A \oplus C \oplus D \oplus IDS_{tr_{new}}$	47.9855 ± 4.9035
$A \oplus D \oplus IDS_{tr_{old}} \oplus IDS_{tr_{new}}$	48.0503 ± 4.9134
$B \oplus C \oplus D \oplus IDS_{tr_{old}}$	48.0292 ± 4.8694
$B \oplus C \oplus D \oplus IDS_{tr_{new}}$	48.0808 ± 4.8879
$B \oplus C \oplus IDS_{tr_{old}} \oplus IDS_{tr_{new}}$	47.9523 ± 4.8812
$B \oplus D \oplus IDS_{tr_{old}} \oplus IDS_{tr_{new}}$	48.0086 ± 4.9228
$C \oplus D \oplus IDS_{tr_{old}} \oplus IDS_{tr_{new}}$	48.0069 ± 4.8666
$A \oplus D \oplus IDS_{tr_{old}} \oplus IDS_{tr_{new}}$	48.0170 ± 4.9237
$A \oplus B \oplus C \oplus D \oplus IDS_{tr_{old}}$	48.0317 ± 4.9723
$A \oplus B \oplus C \oplus D \oplus IDS_{tr_{new}}$	47.9339 ± 4.9490
$A \oplus B \oplus C \oplus IDS_{tr_{old}} \oplus IDS_{tr_{new}}$	48.0242 ± 4.8784
$A \oplus B \oplus D \oplus IDS_{tr_{old}} \oplus IDS_{tr_{new}}$	48.0149 ± 4.8631
$A \oplus C \oplus D \oplus IDS_{tr_{old}} \oplus IDS_{tr_{new}}$	47.9792 ± 4.9179
$B \oplus C \oplus D \oplus IDS_{tr_{old}} \oplus IDS_{tr_{new}}$	48.0441 ± 4.9483
$A \oplus B \oplus C \oplus D \oplus IDS_{tr_{old}} \oplus IDS_{tr_{new}}$	48.0488 ± 4.9018