Hindawi Publishing Corporation
EURASIP Journal on Wireless Communications and Networking
Volume 2011, Article ID 541283, 9 pages
doi:10.1155/2011/541283

Research Article

A Salient Missing Link in RFID Security Protocols

Imran Erguler,[1,2] Emin Anarim,[2] and Gokay Saldamli[3]

[1] *National Research Institute of Electronics and Cryptology, TUBITAK, 41470 Kocaeli, Turkey*
[2] *EE Department, Bogazici University, 34342 Istanbul, Turkey*
[3] *MIS Department, Bogazici University, 34342 Istanbul, Turkey*

Correspondence should be addressed to Imran Erguler, ierguler@uekae.tubitak.gov.tr

Received 20 January 2011; Accepted 14 February 2011

Academic Editor: Damien Sauveron

In side channel analysis, an attacker utilizes some legitimate function queries in order to collect the corresponding responses of a cryptographic system while it is functioning in a normal mode. If those responses reveal some unwanted information about the secrecy or privacy, this leakage is called side channel information and these responses are called side channels. In this respect, careless deployments of "secure" RFID authentication protocols are not exceptions and subject to side channel attacks. Focusing on lightweight RFID security protocols; we examine the server responses for several RFID tags and realize that if the database querying is performed through a static process, the RFID system is subject to timing attacks that could easily jeopardize the system's untraceability criteria. We demonstrate our attack on some well-known protocols and outline a countermeasure by precisely describing the database query mechanism. Furthermore, we analyze the success probability of the attack in terms of the system parameters such as the number of tags, number of cryptographic operations that have to be carried out, and server's computational power.

1. Introduction

As a result of their low production costs and tiny size, RFID tags are considered as the replacement technology for bar codes and other means of traditional identification tools which traditionally find many applications in manufacturing, supply chain management, and inventory control. A typical RFID system consists of mainly three components: tags, one or more readers, and a back-end server. On top of this hardware, a set of networking rules including the authentication (or identification) protocols reside.

RFID technology raises significant privacy issues regarding the traceability concerns. While a person could be traced by tracking his/her mobile phone through a carrier, such a method is no more useful once the phone is turned off. However, this is not the case for someone carrying an RFID gadget. First of all, most users are not aware that they are carrying RFID tags. In fact, even if they know it, tags could not be turned off in general and worse, it automatically responds to queries via radio signals. Therefore, in RFID systems, the attack scenarios and accompanying countermeasures are quite different than the typical wired or wireless systems.

Although public key cryptography has the necessary primitives to solve this sort of problems in various networks, it is not trivial to implement these primitives in networks having constraint devices such as RFID tags without breaking the cost boundaries. In fact, it is a challenging task to design authentication protocols for low-cost RFID tags resisting all of the known attacks and threats and at the same time fulfill the so-called RFID tag specifications. Therefore, solving this delicate task has recently aroused the interest of the security community, and many authentication protocols have been proposed for RFID security. Unfortunately, most of these, like [1–11], failed to address the requirements to a satisfactory extent partially because of not having a common adversary and system definitions.

In this study, our goal is to point out a salient missing link in RFID security protocols, namely, the back-end server (or the database) role and potential pitfalls or side channels in RFID system realization. In side channel analysis, an attacker utilizes some legitimate function queries in order to collect the corresponding responses of a cryptographic system while it is functioning in a normal mode. If those responses reveal some unwanted information about the secrecy or privacy,

this leakage is called side channel information and these responses are called side channels. In this respect, careless deployments of "secure" RFID authentication protocols are not exceptions and subject to side channel attacks.

Focusing on lightweight RFID security protocols, we examine the server responses for several RFID tags and realize that if the database querying is performed through a static process, the RFID system is subject to a timing attack that could easily jeopardize the system's untraceability criteria. Supporting analysis and experiments of this observation are presented with the following outline. In Section 2, after giving a brief update on related work, we describe the basic authentication protocol (BAP) as a building block used in describing our attack model. BAP will further be a basis for various RFID authentication protocols that are vulnerable to the attack. In Section 3, we present our attack model and give probability of its success in terms of the system parameters. Section 4 investigates security of some RFID protocols against the proposed attack. In Section 5, we propose solutions to fix the security flaw. Finally, we conclude in Section 6.

2. Background

2.1. Related Work. The potential security risks in RFID systems hinging the differences in computation time are mentioned in a few published work. Juels and Weis [12] introduced the idea that witnessing a reader's success in identifying a tag could be used in distinguishing two different tags, that is, breaking the privacy of the protocol. For instance, opening a door with a proximity card or acceptance of a payment card can give this information. This ability of the adversary is also touched by Vaudenay [13] and it is formalized in Vaudenay's privacy model. Moreover, Juels-Weis point that computation time of the reader can shed critical light on protocol design and showed that O-TRAP protocol [14] cannot provide strong privacy under this side channel information.

In [14], Burmester et al. briefly considered timing phenomenon by claiming: "In particular the time taken for each pass must be constant. This can be done by inserting an artificial delay on the trusted server". Alternatively, Tsudik [10] has investigated the RFID security protocols against timing attacks targeting computations carried in the tag. It is stated that the time variance in tag computations corresponds to different states of the tags that might make them distinguishable. More recently, Erguler et al. [15] and Erguler and Anarim [16] have exploited the time differences in reader/server responses for different tag states in order to distinguish the tags. They have shown that two protocols described in [17, 18] are vulnerable to such attacks.

At the time this paper was under review, Avoine et al. [19] had extended the Vaudenay's privacy model by formalizing the computational time of the reader. The authors define a new privacy level—TIMEFUL—which is determined by leaked information from the computational time of the reader and add this notion to the privacy levels of model in [13]. Moreover, they present theoretical solutions to the time problem by assigning boolean decisions about TIMEFUL-PRIVACY of a protocol. However, the parameters that may affect the success of the adversary, such as precision of reader time measurement, have been addressed as an engineering problem.

In this paper, we present the actual implementation results and probabilistic analysis for successful timing attack. To be more precise, we give the success probability of the attack in terms of the system parameters such as the number of tags, number of cryptographic operations that have to be carried out, and server's computational power.

2.2. Notation and the BAP. In general, an RFID mutual authentication protocol requires at least three rounds: the reader initiates the communication (Round 1), the tag produces a challenge and sends it to the reader (Round 2), and the reader replies to the challenge (Round 3). In most lightweight RFID authentication protocols, including [1–3, 5, 20–29] (Weis et al.'s the randomized access control scheme [20] performs an exhaustive search in identification of the tag), the server could need to query its entire database in order to authenticate responder tag in Round 2. In fact, this should not be confused with the simple database search since this querying corresponds to a cryptographic exhaustive search where every single query needs a cryptographic operation having a nontrivial time complexity. Therefore, the time complexity of the authentication phase becomes linear in time (i.e., $O(N)$ where N denotes the number of tags in the system). We define these systems as follows.

Definition 1. An RFID system is called linear-time authentication system denoted with LAS if its server performs an exhaustive search to identify or authenticate a tag.

In order to measure the running time differences for different tag searches, it is sufficient to have an exhaustive search process which is identical for each search instance. In fact, for some cases it is possible to achieve some side channels even if the processes are not identical. However, we keep these cases out of our scope and formally define the exhaustive search process as follows.

Definition 2. Let P be the item to be searched, let $\mathbb{S}$ be the set of the search space, and let $(C_t) = \{c_1, c_2, \ldots, c_t, \ldots\}$ be a sequence on $\mathbb{S}$ (i.e., $(C_t) : \mathbb{N} \mapsto \mathbb{S}$). If (C_t) is one-to-one on $\mathbb{S}$ and $C_t = 0$ for $t > N$, then we call (C_t) the query sequence for P.

Note that the query sequence gives the order of the exhaustive search process. For instance, the query sequence having the general term $C_t = t$ for $t \in \mathbb{N}$ with the initial condition $c_1 = 1$ clearly gives the standard exhaustive search process as shown in Figure 1. If this is taken as the process for every search item P, it would be possible to compare the measurements of search time differences.

Definition 3. An LAS RFID scheme is said to be static linear-time authentication system represented as SLAS if the query sequence for all searched items is identical.

It is equivalent to say that for an SLAS RFID scheme, in tag identification/authentication step the order for choosing

the candidates amongst the whole database is the same for all sessions. As the number of tags in the system increases, variance in elapsed time of the reader responses corresponding to the different RFID tags can be measurable for an SLAS RFID scheme. If an adversary is able to access this time difference (an adversary may know the amount of time spent for the tag authentication procedures on the server by simply measuring the elapsed time between the tag's authentication request and its response from the server. Note that this may not be a challenge response; it may be the protocol payload showing whether the server is succeeded or not, in identifying a legitimate tag), then this information will be used as a tool to trace the tags in our attack model.

2.3. BAP. BAP is a generic challenge response authentication protocol used as a basis for most of the RFID authentication protocols. We use the following notations:

$\mathcal{T}$: RFID tag or transponder ,

$\mathcal{R}$: RFID reader or transceiver,

$\mathcal{DB}$: The back-end server,

ID: Identity of a tag,

r_R: Random nonce generated by reader $\mathcal{R}$,

r_T: Random nonce generated by tag $\mathcal{T}$,

Δ: Elapsed time between 2nd and 3rd message flow,

N: Number of tags,

$H()$: One-way hash function.

A step by step description of the BAP that satisfies the LAS properties is given below.

Step 1. $\mathcal{R}$ challenges $\mathcal{T}$ with a random nonce r_R.

Step 2. $\mathcal{T}$ chooses a random nonce r_T and computes $M_1 = f_K(r_R, r_T)$, where K is the secret information and different for each tag and $f()$ is a symmetric cryptographic operation. Then it transmits the result with r_T to $\mathcal{R}$.

Step 3. $\mathcal{R}$ delivers the messages from $\mathcal{T}$ to $\mathcal{DB}$ with r_R.

Step 4. $\mathcal{DB}$ maintains a list of pairs (ID; K_i) and identifies $\mathcal{T}$ by performing an exhaustive search of all stored tag records by computing $M_1' = f_{K_i}(r_R, r_T)$ for each stored ID_i in turn, until it finds a match with M_1. If a match is found, $\mathcal{DB}$ regards the ID as the identity of $\mathcal{T}$.

Step 5. $\mathcal{R}$ replies to challenge of $\mathcal{T}$.

Note that throughout this text, BAP will be used in description of our attack model and will be a basis for many RFID authentication protocols that are vulnerable to the attack.

3. The Timing Attack

Timing attacks provide an attacker with secrets maintained in a security system by measuring the time it takes the system to respond to various queries. For instance, Kocher [30] designed a timing attack to recover secret keys used for RSA decryption. In addition, Brumley and Boneh presented a timing attack on unprotected OpenSSL implementations and showed that such attack was practical, that is, an attacker could measure the response-time variances of a secure Web server and could derive that servers RSA private key [31]. With a similar approach, since in different steps of the RFID protocols, tags, and the server execute different processes, if time taken to execute these steps differs based on the input of tags' state or responses, an attacker can attempt to mount a timing attack to distinguish the tags by analyzing the time variances corresponding to their input. So, with precise measurements of the time difference, an attacker can easily trace the tags and break the untraceability property of the protocol.

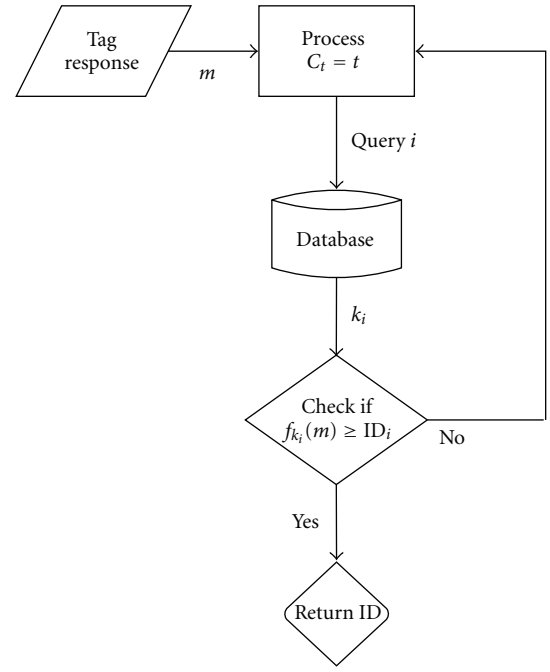

Figure 1: Standard exhaustive search having the general term $C_t = t$.

Intuitively, a protocol satisfies untraceability if an adversary is not able to recognize a previously observed tag [32]. Untraceability issue has been treated formally in different security models, notably driven by Avoine in [33], by Vaudenay in [13], by Van Le et al. in [34], by Juels and Weis in [12], and by Deursen et al. in [35]. The Juels-Weis model characterizes a very strong adversary with a relatively simple definition and according to this model untraceability is defined in terms of privacy experiments by which an adversary could distinguish two different tags within the limits of its computational power and functionality-call bounds. Throughout this text, we adopt the terms and notions of [12] to our needs. In this privacy model, two of the available functions for an adversary are ReaderInit and TagInit. When receiving a ReaderInit message, $\mathcal{R}$ initializes a new session and returns the first challenge of an interactive challenge-response protocol. On the other side, by receiving TagInit message a tag $\mathcal{T}$ is involved in the

corresponding protocol session and it may respond to a protocol message or challenge. For an RFID system $\mathcal{S}$, the privacy experiment, $\text{Exp}_{\mathcal{S}}^{\text{priv}}$, consists of the following two phases:

(1) *Learning Phase.* in this phase, according to Juels-Weis privacy model [12], an adversary $\mathcal{A}$ might initiate a communication with the reader $\mathcal{R}$ (ReaderInit) or a tag $\mathcal{T}$ (TagInit). Also, $\mathcal{A}$ has the ability to modify, insert, or delete messages that agree with the corresponding protocol's procedures. In other words, $\mathcal{A}$ controls the communication channel between $\mathcal{R}$ and each $\mathcal{T}$ and may make any ReaderInit or TagInit calls in any interleaved order without exceeding its parameter bounds.

(2) *Challenge Phase.* in this phase, the adversary $\mathcal{A}$ selects two tag candidates $\mathcal{T}_0$ and $\mathcal{T}_1$ and tests these with the identifers ID_0 and ID_1, respectively. Depending on a randomly chosen bit $b \in \{0,1\}$, $\mathcal{A}$ is given a challenger identifier ID_b from the set $\{\text{ID}_0, \text{ID}_1\}$. That is, $\mathcal{A}$ is given access to one of these tags randomly, called $\mathcal{T}_b$. The adversary might again interact with the reader and the tags. Eventually, at some point, $\mathcal{A}$ decides to terminate the experiment and returns the bit $\widehat{b}$ as its guess for the value of b. The success of $\mathcal{A}$ in guessing b is equivalent to its success of breaking the untraceability, and is quantified as $\mathcal{A}$'s advantage in distinguishing the tag's identity compared to random selection. This is expressed formally as:

$$\text{Adv}_{\mathcal{A},\mathcal{S}}^{\text{Exp}}(k) = \left| \Pr\left[\widehat{b} = b\right] - \frac{1}{2} \right|, \tag{1}$$

where k is the security parameter (i.e., the bit length of the unknown secret ID). An RFID system, $\mathcal{S}$, achieves untraceability if $\text{Adv}_{\mathcal{A},\mathcal{S}}^{\text{Exp}}(k) < \varepsilon(k)$ for some negligible function $\varepsilon()$.

It is equivalent to say that an attack is successful in tracing the tags if the adversary has a nonnegligible advantage in guessing the selected tag. As an illustrative example, assume that the probability of a correct guess is 1/2 (i.e., $\Pr[\widehat{b} = b] = 1/2$). In this case, $\text{Adv}_{\mathcal{A},\mathcal{S}}^{\text{Exp}}(k)$ is zero. Thus, the adversary, $\mathcal{A}$, does not have any advantage in guessing b.

Definition 4. Let γ denote the attacker's precision in distinguishing elapsed time of the reader responses and expressed in terms of seconds, that is, timing resolution.

In our attack model, we suppose the examined protocol is based on SLAS BAP which we call SBAP and for the privacy experiments, the adversary can follow the steps below.

In the learning phase, two tags $\mathcal{T}_0$ and $\mathcal{T}_1$ are randomly selected and then the adversary observes successful authenticated protocols between the reader and the tags and notes respective elapsed time of the reader responses as Δ_0 and Δ_1.

Learning Phase

(1) $\mathcal{A}$ randomly chooses a pair of distinct tags $\mathcal{T}_0$ and $\mathcal{T}_1$.

(2) $\mathcal{A}$ initiates communication with $\mathcal{R}$ using ReaderInit and gets r_{R_0}.

(3) $\mathcal{A}$ initiates communication with $\mathcal{T}_0$ using TagInit.

(4) $\mathcal{A}$ transmits r_{R_0} to $\mathcal{T}_0$.

(5) $\mathcal{A}$ delivers $\mathcal{T}_0$ response to $\mathcal{R}$.

(6) $\mathcal{A}$ measures elapsed time, Δ_0, between 2nd and 3rd message flow.

(7) $\mathcal{A}$ initiates communication with $\mathcal{R}$ using ReaderInit and gets r_{R_1}.

(8) $\mathcal{A}$ initiates communication with $\mathcal{T}_1$ using TagInit.

(9) $\mathcal{A}$ transmits r_{R_1} to $\mathcal{T}_1$.

(10) $\mathcal{A}$ delivers $\mathcal{T}_1$ response to $\mathcal{R}$.

(11) $\mathcal{A}$ measures elapsed time, Δ_1, between 2nd and 3rd message flow.

Notice that in our attack, $\mathcal{A}$ always provides an answer. Thus in the challenge phase, if $|\Delta_0 - \Delta_1| < \gamma$, he makes a random guess for the selected tag $\mathcal{T}_b^*$. On the other hand, if $|\Delta_0 - \Delta_1| \geq \gamma$, the adversary only observes a successful authentication between the legitimate reader and the selected tag and records time duration between the second and the third message flow, call it Δ_*. If $\Delta_* \approx \Delta_0$, the challenge tag is $\mathcal{T}_0$; otherwise the selected tag is $\mathcal{T}_1$.

Challenge Phase

(i) If $|\Delta_0 - \Delta_1| < \gamma$, then $\mathcal{A}$ randomly flips a coin for the value of b.

(ii) If $|\Delta_0 - \Delta_1| \geq \gamma$, then:

(1) $\mathcal{A}$ takes $\mathcal{T}_0$ and $\mathcal{T}_1$ as its challenge candidates.

(2) $\mathcal{A}$ initiates communication with $\mathcal{R}$ using ReaderInit and gets r_R.

(3) $\mathcal{A}$ transmits r_R to the selected tag $\mathcal{T}_b^*$.

(4) $\mathcal{A}$ delivers $\mathcal{T}_b^*$ response to $\mathcal{R}$.

(5) $\mathcal{A}$ measures elapsed time, Δ_*, between 2nd and 3rd message flow.

(6) If $\Delta_* \approx \Delta_0$, $\mathcal{A}$ guesses $b = 0$ and decides $\mathcal{T}_b^* = \mathcal{T}_0$; otherwise, it guesses $\mathcal{T}_b^* = \mathcal{T}_1$.

Lemma 1. *Suppose in exhaustive search of database for each item m, cryptographic operations are evaluated and each operation can be carried out in β seconds. Let n denote the maximum index difference of two candidate elements c_x and c_y of the query sequence (C_t), related to the tags $\mathcal{T}_i$ and $\mathcal{T}_j$, respectively, such that the adversary cannot distinguish the tags by using the above attack. It is equivalent to say that*

$$n \triangleq \max_{x,y\in[1,N]} |x - y| \quad \text{such that } c_x = item_i,\ c_y = item_j,$$
$$\text{Adv}_{\mathcal{A},\text{SBAP}}^{\text{Exp}}(k) < \varepsilon(k). \tag{2}$$

Then we can express $n = \lfloor \gamma/m \cdot \beta \rfloor$.

Proof. If $|\Delta_i - \Delta_j| < \gamma$, then the $\mathcal{A}$ cannot realize time difference, and so does not have a nonnegligible advantage in distinguishing the tags. We know that $|\Delta_i - \Delta_j| = |x - y| \cdot m \cdot \beta$, so $|x - y| \cdot m \cdot \beta < \gamma$ must be satisfied to give a negligible advantage to the adversary. Hence the maximum index difference n can be expressed as

$$n = \left\lfloor \frac{\gamma}{m \cdot \beta} \right\rfloor. \tag{3}$$

□

Definition 5. For privacy experiment Exp_z, suppose $\mathcal{T}_i$ and $\mathcal{T}_j$ are selected and $c_x = \text{item}_i$, $c_y = \text{item}_j$ denote the respective candidates in the exhaustive search process. Then the discrete random variable Q^{Exp_z}, describing the probability of being $|x-y| > n$, that is, $\mathcal{A}$ can sense the time difference, is defined as below

$$Q^{\text{Exp}_z} = \begin{cases} 1, & \text{if } |\text{x} - \text{y}| > n, \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

Proposition 1. *If N denotes number of tags in the database, then for any selected tags $\mathcal{T}_i$ and $\mathcal{T}_j$, the $\mathcal{A}$'s advantage by considering the described attack is expressed as*

$$\text{Adv}^{\text{Exp}}_{\mathcal{A},\text{SBAP}}(k) = \frac{1}{2}\left[1 - \frac{1}{N}\sum_{i=1}^{N} \frac{\min(i-1,n) + \min(N-i,n)}{N-1}\right]. \tag{5}$$

Proof. Success probability of the attack depends on $\Pr[Q^{\text{Exp}} = 1]$. If $Q^{\text{Exp}} = 0$, that is, $|x - y| \leq n$, the adversary has zero advantage since he could just as well have flipped a coin to make the guess, which would have given him the same probability of correct guessing. On the other hand if $Q^{\text{Exp}} = 1$, he can recognize the time difference of the reader responses and makes a correct guess for the privacy experiment with maximum advantage. Therefore, the correct guess probability can be expressed as

$$\begin{aligned} \Pr\left[\hat{b} = b\right] &= \sum_{\forall a \in \{0,1\}} \Pr\left[\hat{b} = b \mid Q^{\text{Exp}} = a\right] \times \Pr\left[Q^{\text{Exp}} = a\right] \\ &= \Pr\left[\hat{b} = b \mid Q^{\text{Exp}} = 0\right] \times \Pr\left[Q^{\text{Exp}} = 0\right] \\ &\quad + \Pr\left[\hat{b} = b \mid Q^{\text{Exp}} = 1\right] \times \Pr\left[Q^{\text{Exp}} = 1\right]. \end{aligned} \tag{6}$$

□

The marginal probabilities of $\Pr[Q^{\text{Exp}}]$ is derived as follows:

$$\Pr\left[Q^{\text{Exp}}\right] = \begin{cases} \Pr[Q^{\text{Exp}} = 0] = \Pr[\,|x - y| \leq n] \\ \qquad = \dfrac{1}{N}\displaystyle\sum_{i=1}^{N} \frac{\min(i-1,n) + \min(N-i,n)}{N-1}, \\ \Pr[Q^{\text{Exp}} = 1] = 1 - \Pr[Q^{\text{Exp}} = 0] \\ \qquad = 1 - \dfrac{1}{N}\displaystyle\sum_{i=1}^{N} \frac{\min(i-1,n) + \min(N-i,n)}{N-1}. \end{cases} \tag{7}$$

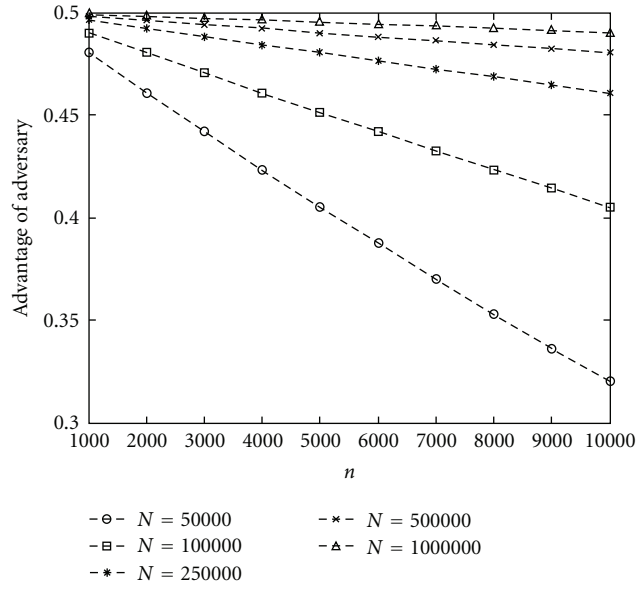

Figure 2: Advantage of the adversary for different n and N.

Notice that $\Pr[\hat{b} = b | Q^{\text{Exp}} = 1] = 1$ and $\Pr[\hat{b} = b | Q^{\text{Exp}} = 0] = \Pr[\text{random coin flip}] = 1/2$. Thus, if we replace these values together with those given in (7) in (6), we obtain

$$\Pr\left[\hat{b} = b\right] = 1 - \frac{1}{2N}\sum_{i=1}^{N} \frac{\min(i-1,n) + \min(N-i,n)}{N-1}. \tag{8}$$

From (1) we obtain

$$\text{Adv}^{\text{Exp}}_{\mathcal{A},\text{SBAP}}(k) = \frac{1}{2}\left[1 - \frac{1}{N}\sum_{i=1}^{N} \frac{\min(i-1,n) + \min(N-i,n)}{N-1}\right]. \tag{9}$$

In Figure 2, advantage of the adversary for different n and N values is shown. Note that as $N \gg n$, $\text{Adv}^{\text{Exp}}_{\mathcal{A}}(k)$ becomes closer to 1/2, which is the maximum advantage.

Remark 1. Since $2n/N > 2n/(N-1) > 1/N \sum_{i=1}^{N}((\min(i-1,n) + \min(N-i,n))/(N-1))$, $\text{Adv}^{\text{Exp}}_{\mathcal{A},\text{SBAP}}(k) > 1/2 - n/N$. Therefore, for $N \gg n$, advantage of an adversary can be approximated as

$$\text{Adv}^{\text{Exp}}_{\mathcal{A},\text{SBAP}}(k) \approx \frac{1}{2} - \frac{n}{N}. \tag{10}$$

In order to illustrate the realization of the presented attack, we consider a real life scenario in a library, where tags are used to identify books and the protocol is an SBAP.

Example 1. We consider an SBAP RFID scheme installed in a public library to substitute bar codes on the books. Suppose the library has got about 1 million of books, $N = 1\,000000$, and we assume that there are about 100000 candidates between c_x and c_y as the candidates related to some books

book$_i$ and book$_j$, respectively, in the database, that is, $|x - y| = 100000$. Besides, we suppose that timing resolution of an adversary who attempts to apply the proposed attack is one millisecond and also to identify a single tag the server needs a single cryptographic operation which is performed in one microsecond. Thus, $\beta = 10^{-6}$, $\gamma = 10^{-3}$, and $m = 1$.

In the light of given information, it is obvious that book$_i$ identification process will be faster than those of book$_j$'s. Moreover, by using (3) we obtain $n = 1000$. Because $n < |x - y| = 100000$, the adversary can easily distinguish these two tags by comparing the elapsed time between 2nd and 3rd rounds of the protocols for each tag.

4. Analysis of Some RFID Schemes

In this section, we examine some RFID privacy schemes proposed in the literature: Those of Song and Mitchell (SM) [5], a challenge/response-based protocol by [3], the scheme of Duc et al. [22], and the model proposed by Ohkubo et al. (OSK) [2]. In our analysis, we do not consider whether or not these protocols have been cryptanalyzed previously by some different type attacks like denial of service, tag impersonation, replay attacks, or others; rather we focus on realization of our attack for these schemes. The common point of these models is that how the candidates are chosen from database in the search process is not defined exactly, and this makes them vulnerable against our attack. In the following parts, we assume that the system relies on a single computer which takes 2^{-20} seconds to carry out a cryptographic operation, that is, $\beta = 2^{-20}$, the number of tags in the system, N, is 2^{20} and $\gamma = 2^{-10}$ seconds (i.e., $\approx$1 ms), unless otherwise is stated. Furthermore, below we only give the three message flows of the protocol since these parts interest us. Update of the secrets and other details can be found in the corresponding study.

4.1. The SM Protocol and Analysis. Song and Mitchell proposed an RFID authentication protocol in [5]. In this scheme, a server stores secrets u_i and t_i for each tag T_i as well as the most recent secrets $\hat{u}_i$ and $\hat{t}_i$. Initially, secret u_i is a string of l bits assigned to T_i, and $t_i = H(u_i)$. Firstly, the reader sends a random bit-string r_1 to the tag. The tag generates a random bit-string r_2, computes two messages $M_1 = t_i \oplus r_2$ and $M_2 = f_{t_i}(r_1 \oplus r_2)$, where f is a keyed hash function, and sends (M_1, M_2) back to the reader. The reader delivers (r_1, M_1, M_2) to the back-end server for tag authentication. The server will search in its database for a record (u_j, t_j) or $(\hat{u}_j, \hat{t}_j)$ such that $M_2 = f_{t_j}(r_1 \oplus M_1 \oplus t_j)$. If a match is found, the server computes $r_2 = M_1 \oplus t_j$, and then computes $M_3 = u_j \oplus (r_2 \gg l/2)$. Finally, the reader forwards M_3 to tag.

The steps of our attack described in Section 3 can be directly applied on SM protocol. In exhaustive search process for each candidate, two cryptographic operations, $f_{t_j}(r_1 \oplus M_1 \oplus t_j)$ and $f_{\hat{t}_j}(r_1 \oplus M_1 \oplus \hat{t}_j)$, are executed, so $m = 2$. By using (3), we obtain $n = 512$. Also from (9), $\mathrm{Adv}^{\mathrm{Exp}}_{\mathcal{A},\mathrm{SM}}(k) = 0.4995$ is computed.

Besides, in [29] an improvement to SM protocol is proposed and to the best of our knowledge it has received no attacks yet. However, same weakness also exists in this protocol and it can be easily broken with our attack.

4.2. The Rhee's Protocol and Analysis. A challenge-response authentication protocol based on a hash function is proposed in [3]. The scheme is vulnerable to our attack as the back-end database is required to perform an ID search to find the specific information related to the tag requesting authentication. The protocol can be summarized as follows.

The reader transmits *Query* and a random number R_{reader}. The tag generates a random number R_{tag}, computes $H(\mathrm{ID} \| R_{\mathrm{reader}} \| R_{\mathrm{tag}})$ generated by itself, and sends the result with R_{tag} to the reader. The reader delivers the tag's response to the back-end server. Next, for each ID stored in the back-end database, the back-end database concatenates ID, R_{reader}, and R_{tag} then hashes it and checks whether or not it is equal to hash result obtained from the tag to authenticate it. The search process continues till a match is found. If the authentication is successful, the back-end database sends $H(\mathrm{ID} \| R_{\mathrm{tag}})$ to the reader and the reader forwards it to the tag.

In the brute force search of server for each candidate, one cryptographic operation is done, so $m = 1$. If we replace the values in (3), we get $n = 1024$ and this leads to $\mathrm{Adv}^{\mathrm{Exp}}_{\mathcal{A}}(k) = 0.499$ for our attack.

4.3. The Duc et al.'s Protocol and Analysis. In [22], a challenge-response protocol for RFID was proposed by Duc et al. According to the protocol, the server stores the following values for each tag: EPC$_i$, tag's access pin PIN$_i$ and the tag key K_i. We can briefly describe the steps of the protocol as given below.

The reader firstly queries a request to tag. The tag generates a random number r, computes $M_1 = \mathrm{CRC}(\mathrm{EPC}_i \| r) \oplus K_i$ and $C = \mathrm{CRC}(M_1 \oplus r)$. Then the tag sends the values (M_1, C, r) back to the reader, which will forward these values to the server, where EPC$_i$ is electronic product code and CRC stands for cyclic redundancy code. For each tuple (EPC_i, K_i) in back-end database, the server verifies that $M_1 \oplus K_i$ equals $\mathrm{CRC}(\mathrm{EPC} \| r)$ and $C = \mathrm{CRC}(M_1 \oplus r)$. If it can find a match, then the tag is successfully identified and authenticated. Next, the server computes $M_2 = \mathrm{CRC}(\mathrm{EPC}_i \| \mathrm{PIN} \| r) \oplus K_i$ and sends M_2 to the tag through the reader.

Now let us apply the proposed attack on Duc et al.'s protocol. Since CRC computation consumes less time than hash or other symmetric encryption, we assume server can evaluate a CRC operation in 2^{-28} seconds so $\beta = 2^{-28}$. Moreover, for each entry from database, one CRC is calculated; $m = 1$. For these values, $n = 2^{18}$ is evaluated and from (9) it means $\mathrm{Adv}^{\mathrm{Exp}}_{\mathcal{A}}(k) = 0.28$ for our attack.

4.4. The OSK Protocol and Analysis. The protocol proposed in [2] relies on hash chains. When a tag is queried by a reader, it sends a hash of its current identifier with H_1 and then updates it using a second hash function H_2. Each tag stores in its memory a random identifier s_i^1. The message flows of the protocol can be depicted as follows: the reader

Table 1: The configuration used in the experiments.

Operating system	Windows XP SP3
CPU	Intel Core2 Quad Q8300
Compiler	MS Visual C#
Cryptographic library	.net System.Security.Cryptography

sends an identification request to the tag and receives back $r_i^k = H_1(s_i^k)$ where s_i^k is the current identifier of the tag. Then tag replaces s_i^k by $s_i^{k+1} = H_2(s_i^k)$. On the server side, from r_i^k, the system identifies the corresponding tag. In order to do this, it constructs the hash chains from each N initial value until it finds the match with r_i^k or until it reaches a given maximum limit δ on the chain length. The threshold δ is the number of read operations on a single tag between two updates of the database. A suited size for δ could be 128 as mentioned in [36].

Notice that OSK protocol does not exactly fit to the steps of BAP, because after identification of the tag, the reader does not send any message to the tag. Hence, how we can apply the proposed attack on OSK could arise in our minds. Although the reader does not respond in the third message flow, as presented in [12] the adversary can record the elapsed time till tag identification is realized by observing a validation event. For example, opening a door with a proximity card or acceptance of a payment card can be used as validation events. Nevertheless, one can argue that the work of attacker is more difficult than the cases of previous protocols. Therefore, we assume that the attacker's time distinguishing capability may be lower and set $\gamma = 1$ seconds. In addition, according to OSK protocol for each trial, 2 δ hash operations are computed. For $\delta = 128$, we get $m = 256$, and by using (3), $n = 4096$ is evaluated. As a result, from the equation for advantage of the adversary $\mathrm{Adv}^{\mathrm{Exp}}_{\mathcal{A},\mathrm{OSK}}(k) = 0.496$ is obtained.

4.5. Experimental Results. We experimentally examine the capabilities of the proposed timing analysis by implementing the Rhee's and Duc et al.'s protocols. These are chosen for simplicity though similar experimental result can be achieved for other mentioned protocols in this section.

The source code was compiled using the MS Visual C# compiler with default optimizations. All of the experiments were run under the configuration shown in Table 1. We used random keys generated by .net System.Random class key generation routine. We measured the time using stopwatch class and take the averages in order to measure elapsed time accurately.

In the implementation of Rhee's protocol, we use the standard SHA-1 in MS .net System.Security.Cryptography class where we write a custom class for CRC implementation used in Duc et al.'s protocol. Our CRC routine uses a lookup table which reflects the lightweight feature of the protocol as it outperforms the SHA-1 implementation. In Tables 2 and 3, timings for exhaustive search steps of the protocols are tabulated, where "index difference" stands for $|x - y|$ as mentioned in Lemma 1. As formulated in the previous section, timing attacks could be very powerful in case a poor search process is chosen.

Table 2: Timing attack on Rhee's protocol [3].

Index difference	Δ time
500	12 ms
1000	27 ms
10 K	298 ms
100 K	3033 ms
500 K	14852 ms
1 M	29780 ms

Table 3: Timing attack on Duc et al.'s protocol [22].

Index difference	Δ time
500	0.7 ms
1000	1.1 ms
10 K	20 ms
100 K	97 ms
500 K	417 ms
1 M	807 ms

5. Countermeasures

Before giving our countermeasure against the proposed attack, we want to elaborate on some other obvious but not efficient techniques that remedy the security flaw.

With consideration of the previous parts, intuitively one can provide security against the proposed timing attack by realizing the condition that the server response time variation for different tags is negligible, in other words, the server responds with an equal time and this is same for all tags. This condition can be achieved by using look-up tables as mentioned in [19] or artificially padding the delay in reader responses for all tags as reported in [14]. For the lookup-table model the server stores all possible answers of tags that are precomputed previously. Thus, in authentication phase the server avoids to make an exhaustive search, and instead responds in constant-time. Note that although this solution fixes the problem, constructing such a scheme with satisfying security and implementation requirements is impractical. In fact, if such a system exists, then clearly the use of exhaustive search in tag identification would be abandoned. On the other hand, inserting an artificial delay at the server side, determined by the worst case time, definitely eradicates the security flaw. However, this is clearly undesirable, because it reduces the efficiency of the overall system.

Our timing analysis and the experimental results of the previous section exploit the use of an SLAS RFID scheme which uses a static exhaustive search process on server authentication. However, if a dynamic search process is employed the described timing analysis would fail in measuring the running time differences for different tag searches. In this respect, the simplest countermeasure to avoid such attacks would be simply changing the starting

point of the exhaustive search process. We formulate this countermeasure as follows.

Countermeasure 1. Let an RFID system implements an SLAS scheme, having the same query sequence (C_t) for all of its search items such as $\{P_0, P_1, \ldots, P_l\}$ for some positive integer l. Choosing nonidentical random query sequences (C_t) for all search items gives the desired protection for the described attacks.

Although Countermeasure 1 gives a wide range of selection having different implementation complexities, naturally, the simplest countermeasures come from setting minimal differences between query sequences. Observe that query sequences (C_t) can also be seen as permutations on N items where N is the number tags. Thus, composing the query sequences with the following constant cyclic permutation π_j gives nonidentical shifted query sequences:

$$\pi_j(i) = i + r_j \bmod N, \quad \text{for random } r_j,\ j = 0, 1, \ldots, l,\ 0 < i \le N. \tag{11}$$

In other words, for all search items $\{P_0, P_1, \ldots, P_l\}$ we have the following general terms for the corresponding query sequences

$$C_{\pi_j(t)}, \quad \text{for } j = 0, 1, \ldots, l. \tag{12}$$

In fact, this modification corresponds to random selection of the starting point of the exhaustive search process. Since we use different query sequences for different tag searches, for any selected two tags the index difference will also be different. Therefore, timing attacks would fail in measuring the time differences.

6. Conclusion

It is shown that exhaustive search process is crucial in RFID authentication protocols. Although the protocol might satisfy the necessary security requirements of the RFID system specifications, careless deployments of database search mechanisms could jeopardize the security of the whole system. Therefore, it should not be left to user's choice and has to be described precisely in the system specifications. We believe our attempt would point out this salient missing link in RFID security protocols and address the potential pitfalls or side channels in realizations.

In order to support our observation through a careful analysis we give the minimum index difference of two selected tags in database such that the attacker succeeds. In addition, the success probability of the proposed attack model is derived in terms of the number of tags in the system, number of cryptographic operations carried out by the server, the computational power of the server, and the sensitivity of the attacker in timing.

As a countermeasure for the timing attack, we propose a dynamic search process which would fail in measuring the running time differences for different tag searches. We claim that choosing nonidentical random query sequences (C_t) for all search items gives the desired protection for the described attacks.

Acknowledgments

The authors would like to thank the anonymous reviewers for their constructive comments and suggestions on this work. Note that, G. Saldamli is partially funded by Bogazici University BAP project No: 5721.

References

[1] H. Y. Chien and C. H. Chen, "Mutual authentication protocol for RFID conforming to EPC class 1 generation 2 standards," *Computer Standards and Interfaces*, vol. 29, no. 2, pp. 254–259, 2007.

[2] M. Ohkubo, K. Suzuki, and S. Kinoshita, "Cryptographic approach to "privacy-friendly" tags," RFID Privacy Work-6 shop, MIT, MA, USA, 2003.

[3] K. Rhee, J. Kwak, S. Kim, and D. Won, "Challenge-response based on RFID authentication protocol for distributed database environment," in *Proceedings of the International Conference on Security in Pervasive Computing (SPC '05)*, D. Hutter and M. Ullmann, Eds., vol. 3450 of *Lecture Notes in Computer Science*, pp. 70–84, Springer, Berlin, Germany, 2005.

[4] D. Nguyen Duc, J. M. Park, H. R. Lee, and K. J. Kim, "Enhancing security of EPCglobal Gen-2 RFID tag against traceability and cloning," in *Proceedings of the Symposium on Cryptography and Information Security*, Hiroshima, Japan, January 2006.

[5] B. Song and C. J. Mitchell, "RFID authentication protocol for low-cost tags," in *Proceedings of the ACM Conference on Wireless Network Security (WiSec '08)*, V. D. Gligor, J. Hubaux, and R. Poovendran, Eds., pp. 140–147, ACM Press, Alexandria, Va, USA, 2008.

[6] T. Dimitriou, "A lightweight RFID protocol to protect against traceability and cloning attacks," in *Proceedings of the Conference on Security and Privacy for Emerging Areas in Communication Networks (SecureComm '05)*, IEEE, Athens, Greece, September 2005.

[7] D. Henrici and P. Müller, "Hash-based enhancement of location privacy for radio-frequency identification devices using varying identifiers," in *Proceedings of the International Workshop on Pervasive Computing and Communication Security (PerSec '04)*, pp. 149–153, IEEE Computer Society, Orlando, Fla, USA, March 2004.

[8] D. Molnar and D. Wagner, "Privacy and security in library RFID: issues, practices, and architectures," in *Proceedings of the ACM Conference on Computer and Communications Security (CCS '04)*, B. Pfitzmann and P. Liu, Eds., pp. 210–219, ACM Press, Washington, DC, USA, October 2004.

[9] J. C. Ha, J. H. Ha, S. J. Moon, J. M. Gonzalez Nieto, and C. Boyd, "Low-cost and strong-security RFID authentication protocol," in *Proceedings of the EUC Workshops*, vol. 4809 of *Lecture Notes in Computer Science*, pp. 795–807, Springer, Berlin, Germany, 2007.

[10] G. Tsudik, "A family of dunces: trivial RFID identification and authentication protocols," Cryptology ePrint Archive Report 2006/015, 2007.

[11] C. C. Tan, B. Sheng, and Q. Li, "Serverless search and authentication protocols for RFID," in *Proceedings of the 5th Annual IEEE International Conference on Pervasive Computing and Communications (PerCom '07)*, pp. 3–12, March 2007.

[12] A. Juels and S. Weis, "Defining strong privacy for RFID," in *Proceedings of the 5th Annual IEEE International Conference on Pervasive Computing and Communications (PerCom '07)*, pp. 342–347, 2007, Full version available at IACR ePrint Archive.

[13] S. Vaudenay, "On privacy models for RFID," in *Proceedings of the Advances in Cryptology (ASIACRYPT '07)*, vol. 4833 of *Lecture Notes in Computer Science*, pp. 68–87, Springer, Berlin, Germany, 2007.

[14] M. Burmester, T. V. Le, and B. D. Medeiros, "Provably secure ubiquitous systems: universally composable RFID authentication protocols," in *Proceedings of the Conference on Security and Privacy for Emerging Areas in Communication Networks (SecureComm '06)*, IEEE, 2006.

[15] I. Erguler, M. Akgun, and E. Anarim, "Cryptanalysis of a lightweight RFID authentication protocol—LRMAP," in *Proceedings of the Western European Workshop on Research in Cryptology (WeWORC '09)*, Graz, Austria, July 2009.

[16] I. Erguler and E. Anarim, "Scalability and security conflict for RFID authentication protocols," *Wireless Personal Communications*. In press.

[17] B. Song and C. J. Mitchell, "Scalable RFID pseudonym protocol," in *Proceedings of the 3rd International Conference on Network and System Security (NSS '09)*, pp. 216–224, IEEE Computer Society Press, October 2009.

[18] J. Ha, J. Ha, S. Moon, and C. Boyd, "LRMAP: lightweight and resynchronous mutual authentication protocol for RFID system," in *Proceedings of the International Conference on Ubiquitous Convergence Technology (ICUCT '07)*, F. Stajano, H.-J. Kim, J.-S. Chae, and S.-D. Kim, Eds., vol. 4412 of *Lecture Notes in Computer Science*, pp. 80–89, Springer, Berlin, Germany, 2007.

[19] G. Avoine, I. Coisel, and T. Martin, "Time measurement threatens privacyfriendly RFID authentication protocols," in *Proceedings of the Workshop on RFID Security (RFIDSec '10)*, Istanbul, Turkey, June 2010.

[20] S. Weis, S. Sarma, R. Rivest, and D. Engels, "Security and privacy aspects of low-cost radio frequency identification systems," in *Proceedings of the International Conference on Security in Pervasive Computing (SPC '03)*, D. Hutter, G. Mller, W. Stephan, and M. Ullmann, Eds., vol. 2802 of *Lecture Notes in Computer Science*, pp. 201–212, Springer, Berlin, Germany, 2003.

[21] Y. An and S. Oh, "RFID system for users privacy protection," in *Proceedings of Asia-Pacific Conference on Communications*, pp. 516–519, Perth, Australia, 2005.

[22] D. N. Duc, J. Park, H. Lee, and K. Kim, "Enhancing security of EPC global gen-2 RFID tag against traceability and cloning," in *Proceedings of the Symposium on Cryptography and Information Security (SCIS '06)*, The Institute of Electronics, Information and Communication Engineers, Hiroshima, Japan, 2006.

[23] K. Osaka, T. Takagi, K. Yamazaki, and O. Takahashi, "An efficient and secure RFID security method with ownership transfer," in *Proceedings of the Computational Intelligence and Security (CIS '06)*, Y. Wang, Y. Cheung, and H. Liu, Eds., vol. 4456 of *Lecture Notes in Computer Science*, pp. 778–787, Springer, Berlin, Germany, 2006.

[24] S. Lee, T. Asano, and K. Kim, "RFID mutual authentication scheme based on synchronized secret information," in *Proceedings of the Symposium on Cryptography and Information Security (SCIS '06)*, The Institute of Electronics, Information and Communication Engineers, Hiroshima, Japan, 2006.

[25] P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estevez-Tapiador, and A. Ribagorda, "An efficient authentication protocol for RFID systems resistant to active attacks," in *Proceedings of the Emerging Directions in Embedded and Ubiquitous Computing (EUC '07)*, vol. 4809 of *Lecture Notes in Computer Science*, pp. 781–794, Springer, Berlin, Germany, 2007.

[26] S. Fouladgar and H. Afifi, "A simple privacy protecting scheme enabling delegation and ownership transfer for RFID tags," *Journal of Communications*, vol. 2, no. 6, pp. 6–13, 2007.

[27] H. Y. Chien and C. W. Huang, "A lightweight RFID protocol using substring," in *Proceedings of the IFIP International Conference on Embedded and Ubiquitous Computing (EUC '07)*, vol. 4808 of *Lecture Notes in Computer Science*, pp. 422–431, Springer, Berlin, Germany, 2007.

[28] T. Lim, T. Li, and T. Gu, "Secure RFID identification and authentication with triggered hash chain variants," in *Proceedings of the 14th International Conference on Parallel and Distributed Systems (ICPADS '08)*, pp. 583–590, IEEE Computer Society, Melbourne, Australia, 2008.

[29] S. Cai, Y. Li, T. Li, and R. Deng, "Attacks and improvements to an RFID mutual authentication protocol and its extensions," in *Proceedings of the 2nd ACM Conference on Wireless Network Security (WiSec '09)*, pp. 51–58, ACM Press, Zurich, Switzerland, 2009.

[30] P. Kocher, "Timing attacks on implementations of Diffie-Hellman, RSA, DSS and other systems," in *Proceedings of the Advances in Cryptology (CRYPTO '96)*, N. Koblitz, Ed., vol. 1109 of *Lecture Notes in Computer Science*, pp. 104–113, Springer, Berlin, Germany, 1996.

[31] D. Brumley and D. Boneh, "Remote timing attacks are practical," in *Proceedings of the 12th Usenix Security Symposium (SECURITY '04)*, pp. 1–14, Washington DC, USA, 2004.

[32] T. van Deursen and S. Radomirović, "Security of RFID protocols—a case study," *Electronic Notes in Theoretical Computer Science*, vol. 244, pp. 41–52, 2009.

[33] G. Avoine, "Adversarial model for radio frequency identification," Cryptology ePrint Archieve Report 2005/049, 2005, http://eprint.iacr.org.

[34] T. Van Le, M. Burmester, and B. De Medeiros, "Universally composable and forward-secure RFID authentication and authenticated key exchange," in *Proceedings of the 2nd ACM Conference on Computer and Communications Security (CCS '07)*, pp. 242–252, Singapore, March 2007.

[35] T. V. Deursen, S. Mauw, and S. Radomirovic, "Untraceability of RFID protocols," in *Proceedings of the Information Security Theory and Practices. Smart Devices, Convergence and Next Generation Networks (WISTP '08)*, vol. 5019 of *Lecture Notes in Computer Science*, pp. 1–15, Springer, Berlin, Germany, 2008.

[36] G. Avoine, E. Dysli, and P. Oechslin, "Reducing time complexity in RFID systems," in *Proceedings of the Selected Areas in Cryptography (SAC '05)*, B. Preneel and S. Tavares, Eds., vol. 3897 of *Lecture Notes in Computer Science*, pp. 291–306, Springer, Berlin, Germany, 2005.