Pre vs Post State Update: Trading Privacy for Availability in RFID

Mike Burmester, Senior Member IEEE, and Jorge Munilla

***Abstract*—Designing lightweight RFID protocols that provide strong privacy is a major challenge. For anonymity tags must use pseudonyms that have to be refreshed with every interrogation (whether completed or not). For forward secrecy, the state of a tag must be updated and it must be hard to reverse updates. Since the interrogating reader can be adversarial, the adversary may control state updates. It follows that it may not be possible for a tag to maintain synchrony with authorized readers. In this letter we analyze a recently proposed RFID protocol and show that there is a fundamental trade-off between privacy and availability. We then prove that for lightweight RFID applications strong privacy cannot be achieved in the presence of a Byzantine adversary.**

***Index Terms*—RFID, privacy, unlinkability, forward secrecy, DoS, protocol failure.**

I. Introduction

RADIO Frequency Identification (RFID) is a well established wireless technology for inventory, retail and supply-chain management. Initial designs focused on performance with less attention paid to security. However, as early as 2002 privacy issues were raised. In 2003 the CASPIAN group raised concerns regarding the possible misuse of RFID technology. In 2008 the European Commission launched a public consultation on privacy issues for RFID, in particular regarding data protection and information security [1]. This resulted in pressure to legislate/regulate RFID technologies and protect access to personal information [2], [3]. Several RFID authentication protocols that address privacy have been proposed in the literature. We refer the reader to the on-line RFID repository of Avoine [4].

In this letter we investigate a recent paper by Sun and Zhong on hash-based RFID security [5]. This identifies vulnerabilities of an RFID protocol proposed earlier by Ha et al. [6], and describes a modification that addresses these vulnerabilities. However, as we shall see, the proposed solution raises other security concerns.

Our goal in this letter is to show that for lightweight RFID applications there is a fundamental trade-off between privacy and availability. Any solution that improves one of these features will have a negative impact on the other. This trade-off should be taken into account when designing RFID protocols that support privacy.

II. Background: RFID deployments, Threat model and Privacy Features

RFID Deployments. A typical RFID deployment involves three types of legitimate entities: *tags*, *readers* and *back-end servers* [7].

Tags. These are attached to, or embedded in, host objects to be identified. The most common low cost tags are passive tags that have no power of their own and get power from the radio waves of the reader. Such tags are unable to perform public key cryptographic operations, and are restricted to inexpensive conventional operations such as hash functions and symmetric key operations and a modest amount of computation.

Readers. These have resources at least comparable to those of a cellphone. Readers implement a radio interface to the tags (including an RF module, a control unit and a coupling element) and a high-level interface to a back-end server that eventually processes captured data.

Back-End Server. This is a trusted entity that maintains a database containing all information needed to identify tags, including their identification numbers. Since the integrity of an RFID system depends entirely on the proper behavior of the server, this should be physically secure. Servers and readers are sometimes treated as one entity. However replicating the security functionality on all readers is not practical (compromising of a single reader would undermine the security of the whole system) and poses a management nightmare (changing any security-related parameter would require modifying all readers).

Server-Reader Communication. Several readers may be assigned to a single server. These entities can implement sophisticated cryptographic protocols and therefore all communication between a server and readers is over private and authenticated channels.

Reader-Tag communication. Tags can only communicate with readers. These must be in wireless range. RFID wireless channels are particularly vulnerable because tags are restricted to lightweight protection [8].

Threat Model. We assume a Byzantine threat model for which all parties: the tags $\mathcal{T}$, the readers $\mathcal{R}$, the server and the adversary $\mathcal{A}$, are probabilistic polynomial-time (PPT) Turing machines. $\mathcal{A}$ controls the delivery schedule of all communication and may eavesdrop or modify contents, and attempt to perform impersonation, reflection, man-in-the middle, or other passive or active attacks [9].

Privacy Features. In the context of RFID, privacy

M. Burmester is with the Department of Computer Science, Florida State University, Tallahassee, FL, 30302. E-mail: burmeste@cs.fsu.edu

J. Munilla is with the Communication Engineering Department, Univ. de Málaga, Spain, 29071. E-mail: munilla@ic.uma.es

Manuscript received xxxx, 20xx; revised December xx, 20xx.

refers to anonymity, that is the inability of a passive or active adversary to obtain knowledge that can be used to identify (partly or wholly) a tag. An adversary that physically tracks a tag can determine which executions of the protocol are linked to this tag. This kind of tracking cannot be prevented. The concept of unlinkability is then related to the capability of an adversary to link interrogations once this physical action is temporarily interrputed.

Unlinkability is a feature of anonymity which protects past interrogations $\mathsf{int}_1, \mathsf{int}_2$ (partial or completed) of a tag from being linked by an adversary $\mathcal{A}^u$ that is allowed to interact arbitrarily with tags $\mathcal{T}$ and readers $\mathcal{R}$. We require that: $\forall$ PPT $\mathcal{T}$; $\forall$ PPT $\mathcal{R}$; $\forall$ pairs of interrogations $\mathsf{int}_1, \mathsf{int}_2$; $\nexists$ a PPT $\mathcal{A}^u$ that can decide with advantage

$$Adv_{\mathcal{A}^u} = |P_1 - P_2| \tag{1}$$

better than negligible whether the same tag $\mathcal{T}$ is involved in both interrogations. Here P_1 is the probability that $\mathcal{T}$ is involved in $\mathsf{int}_1, \mathsf{int}_2$ and P_2 the probability that $\mathcal{A}^u$ succeeds in detecting that $\mathcal{T}$ is involved in $\mathsf{int}_1, \mathsf{int}_2$.

Session unlinkability is a weak form of unlinkability for which we require additionally that either int_1 completed, or int_1, int_2 are separated by a completed interrogation involving the tag of int_1 [10].

Forward Secrecy is a strong form of anonymity which protects past interrogations int (partial or completed) of a tag $\mathcal{T}$ from being linked to $\mathcal{T}$ by an adversary $\mathcal{A}^{fs}$ that succeeds in compromising $\mathcal{T}$ (can access all private information stored in $\mathcal{T}$). We require that: $\forall$ PPT $\mathcal{T}$; $\forall$ PPT $\mathcal{R}$; $\forall$ interrogations int (partial or completed); $\nexists$ a PPT adversary $\mathcal{A}^{fs}$ that after first being allowed to interact arbitrarily with readers and tags and then given access to the state of $\mathcal{T}$, can decide with advantage

$$Adv_{\mathcal{A}^{fs}} = |P_1' - P_2'| \tag{2}$$

better than negligible whether $\mathcal{T}$ was involved in int. Here P_1' is the probability that $\mathcal{T}$ is involved in int and P_2' the probability that $\mathcal{A}^{fs}$ detects that $\mathcal{T}$ is involved in int.

Session forward secrecy is a weak form of forward secrecy. As with session unlinkability, we require that either int completed, or an intermediate interrogation involving $\mathcal{T}$ completed before $\mathcal{A}$ can access the private information of $\mathcal{T}$. That is, int and the compromise/capture of $\mathcal{T}$ are separated by a completed interrogation involving $\mathcal{T}$.

III. The LRMAP Protocol

We briefly describe the Lightweight and Resynchronous Mutual Authentication Protocol (LRMAP) proposed by Ha *et al.* [6], see Figure 1.

In this protocol the reader $\mathcal{R}$ and tag $\mathcal{T}$ share a cryptographic hash function H and a secret ID, which is updated at the end of each successful protocol execution. We call this *post-updating*. $\mathcal{R}$ keeps for each tag: the current value ID, the previously used value PID and, for efficiency, the hash of ID: $HID = H(ID)$. $\mathcal{T}$ uses a bit flag $Sync$ to check if the last protocol execution was successful ($Sync = 0$) or not ($Sync = 1$), to avoid sending the same identifying message.

$\mathcal{R}$ starts by sending $\mathcal{T}$ a nonce $N_\mathcal{R}$. Upon receiving it, $\mathcal{T}$ generates a nonce $N_\mathcal{T}$ and computes a message P that depends on the value of $Sync$: if $Sync = 0$ (successful previous execution) then $P = H(ID)$, while if $Sync = 1$ (unsuccessful previous execution) then $P = H(ID||N_\mathcal{T})$, where $||$ is concatenation. Then $\mathcal{T}$ sets $Sync$ to 1, computes $Q = H(ID||N_\mathcal{T}||N_\mathcal{R})$, and sends $\mathcal{R}$ the left part of this message $LP(Q)$ along with P and its nonce.

$\mathcal{R}$ uses P to identify $\mathcal{T}$ and $LP(Q)$ to authenticate $\mathcal{T}$. If $\mathcal{T}$ is authenticated, then R replies with the right part $RP(Q)$ of the message Q. $\mathcal{T}$ checks this for correctness, and if correct (*post*) updates ID to $H(ID||N_\mathcal{R})$ and sets $S \leftarrow 0$.

Ha *et al.* presented a formal security model for RFID location privacy and proved the security of LRMAP in this model [11]. Some flaws were later found in this security model [12] and an attack on LRMAP is described in [13]. This exploits the reader's response-time, but can easily be addressed by setting fixed response-times.

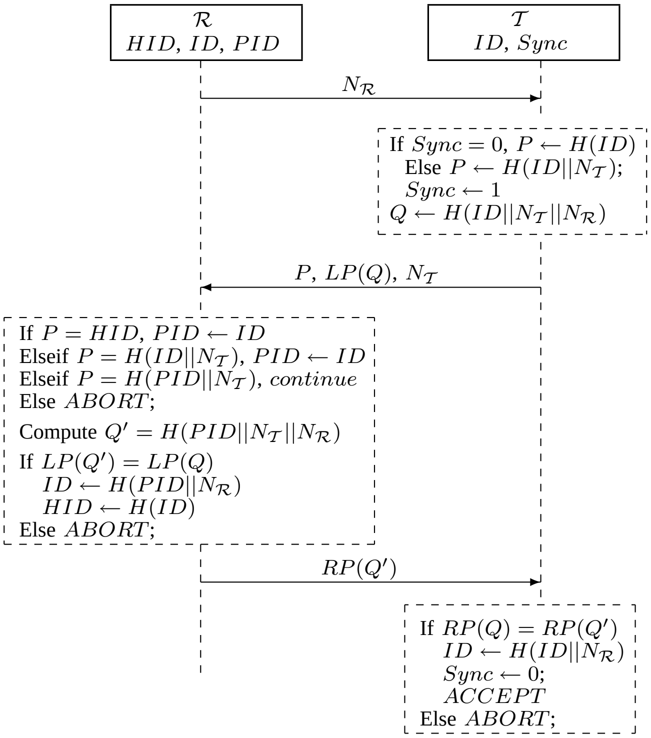
$\mathcal{R}$
HID, ID, PID
$\mathcal{T}$
$ID, Sync$
$N_\mathcal{R}$
If $Sync = 0$, $P \leftarrow H(ID)$
Else $P \leftarrow H(ID||N_\mathcal{T})$;
$Sync \leftarrow 1$
$Q \leftarrow H(ID||N_\mathcal{T}||N_\mathcal{R})$
P, $LP(Q)$, $N_\mathcal{T}$
If $P = HID$, $PID \leftarrow ID$
Elseif $P = H(ID||N_\mathcal{T})$, $PID \leftarrow ID$
Elseif $P = H(PID||N_\mathcal{T})$, $continue$
Else $ABORT$;
Compute $Q' = H(PID||N_\mathcal{T}||N_\mathcal{R})$
If $LP(Q') = LP(Q)$
$ID \leftarrow H(PID||N_\mathcal{R})$
$HID \leftarrow H(ID)$
Else $ABORT$;
$RP(Q')$
If $RP(Q) = RP(Q')$
$ID \leftarrow H(ID||N_\mathcal{R})$
$Sync \leftarrow 0$;
$ACCEPT$
Else $ABORT$;

Fig. 1. The LRMAP Protocol

IV. The Sun-Zhong LRMAP

Sun and Zhong use the value of $Sync$ in LRMAP and its relation to the completion of the previous interrogation to show that it is possible to distinguish a target tag from others [5], which violates forward secrecy. In particular, an attacker that eavesdrops on an interrogation of tag $\mathcal{T}$ can determine if this was successful or not. If $\mathcal{T}$ is compromised soon after, then it is possible to determine with non-negligible probability whether $\mathcal{T}$ was, or was not involved, by checking the value of $Sync$. Sun and Zhong proposed a new protocol that addresses this vulnerability, see Figure 2.

In the modified protocol ID is (*pre*) updated with every interrogation, no matter if it is successful or not. This value is then used as the secret part of the input of another hash function H_1 to obfuscate and authenticate the contents of the exchanged messages. The reader, in order to identify the tag $\mathcal{T}$ has to construct a hash chain $H(ID), \ldots, H^i(ID)$ (as in [14], [15], [16]) until the expected value is obtained, or a threshold t is reached.

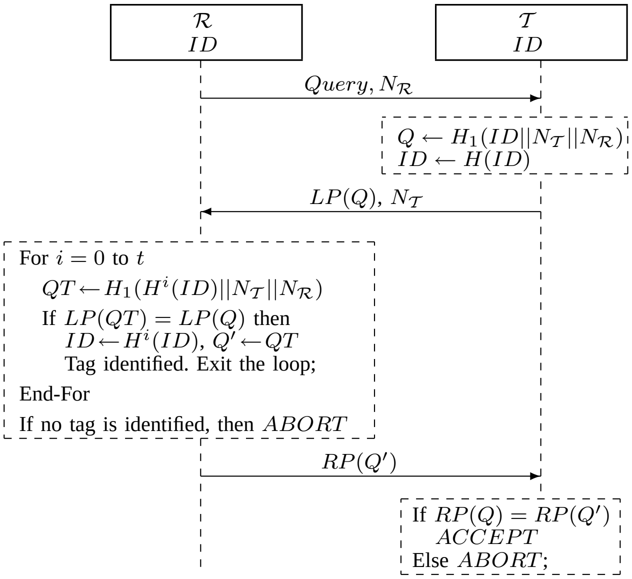
$\mathcal{R}$
ID
$\mathcal{T}$
ID
$Query, N_{\mathcal{R}}$
$Q \leftarrow H_1(ID||N_{\mathcal{T}}||N_{\mathcal{R}})$
$ID \leftarrow H(ID)$
$LP(Q), N_{\mathcal{T}}$
For $i = 0$ to t
$QT \leftarrow H_1(H^i(ID)||N_{\mathcal{T}}||N_{\mathcal{R}})$
If $LP(QT) = LP(Q)$ then
$ID \leftarrow H^i(ID)$, $Q' \leftarrow QT$
Tag identified. Exit the loop;
End-For
If no tag is identified, then $ABORT$
$RP(Q')$
If $RP(Q) = RP(Q')$
$ACCEPT$
Else $ABORT$;

Fig. 2. The SZ-LRMAP Protocol

V. Analysis of LRMAP and SZ-LRMAP

LRMAP. The Sun and Zhong [5] attack on LRMAP which uses the one bit leaked by $Sync$ when a tag $\mathcal{T}$ gets compromised/captured, is rather weak. To link an interrogation int to $\mathcal{T}$ the adversary must compromise/capture $\mathcal{T}$ just after it has eavesdropped on int, making certain that it has not been interrogated again and that its state is therefore the same as when eavesdropped. Furthermore, the attack only identifies the group of interrogated tags in which $\mathcal{T}$ belongs.

However there is another attack on forward secrecy. An adversary who eavesdrops on an interrupted interrogation obtains the values $LP(Q)$, $N_{\mathcal{R}}$ and $N_{\mathcal{T}}$. Later (not necessary just after) when the tag gets compromised the current value of ID will be disclosed. Thus the adversary only needs to compute $LP(ID||N_{\mathcal{T}}||N_{\mathcal{R}})$ and compare the result with $LP(Q)$. If no intermediate successful interrogation took place then these values match and the adversary will be able to identify the tag without ambiguity. That is, LRMAP provides only *session* forward secrecy protection (Section II).

SZ-LRMAP. By *pre-updating* ID instead of *post-updating* it, Sun and Zhong (Section IV) improve the privacy features but at a cost: the new protocol is subject to availability attacks. The protection offered by the threshold t that prevents the reader computing hash chains of indefinite length, is only a patch and does not solve the problem. In fact, if the value of t is low then a de-synchronization attack is possible on the tags. An adversary impersonating the reader $\mathcal{R}$ can query s times a tag $\mathcal{T}$, $s > t$: this will *permanently* de-synchronize $\{\mathcal{T}, \mathcal{R}\}$, and as a result $\mathcal{R}$ cannot identify $\mathcal{T}$ anymore ($\mathcal{T}$ is "killed"). By contrast if the value of t is high then the we have a DoS attack on $\mathcal{R}$. Let n be the number of tags in the database of $\mathcal{R}$. Then an adversary impersonating tags, can send junk messages to $\mathcal{R}$. $\mathcal{R}$ will need to perform $2nt$ computations (involving H and H_1) and nt checks before discarding each of these messages.

The proposed fix for LRMAP therefore replaces one vulnerability for another. In following sections we discuss this trade-off in a more formal way.

VI. Hash-Based Privacy Modes

Updating tag identities by using one-way hash functions is a natural way to get forward secrecy. Let N be a nonce. We distinguish three privacy modes that tags can use to get identified, which are listed below with the corresponding updates:

1) $H(ID)$; ID post-updated: $ID \leftarrow H(ID||N)$.
2) $H(ID||N)$; ID post-updated: $ID \leftarrow H(ID||N)$.
3) $H(H^i(ID)||N)$; ID pre-updated: $ID \leftarrow H(ID)$.

Each of these modes presents different characteristics which are summarized in Table I.

TABLE I
Identifying Modes

Identifying Value	Updating Mode	Reader Computations	Unlinkability	Forward-Secrecy
$H(ID)$	Post	$2nc$	Session	Session
$H(ID\|\|N)$	Post	$2n(h+c)$	Strong	Session
$H(H^i(ID)\|\|N)$	Pre	$tn(2h+c)$	Strong	Strong

In the first case the identifying information is stored in the database of the reader and therefore only one look-up is needed. However, because the reader does not know if the pseudonyms were updated successfully in the previous session, there are two possible values for each tag. As a result $2n$ checks (denoted as c in the table) are required. This mode only provides session unlinkability and session forward secrecy (Section II) since the identifying message and the stored ID only change when the protocol is successful.

In the second privacy mode, ID is still post-updated and therefore as before it provides session forward secrecy. We get session unlinkability by computing a different random identifying message $H(ID||N)$ each time a tag is queried. The price for this is that the reader has to carry out a hash computation (denoted as h in the table) and check the result for two possible values of ID (the current and the previous one) for all tags in its database before discarding the message: *i.e.* $2n(h+c)$ computations.

Finally, the last privacy mode captures both strong unlinkability and strong forward secrecy, by pre-updating ID every time the tag is interrogated. However in this mode the reader $\mathcal{R}$ may have to keep searching for a tag indefinitely, and therefore a threshold t is imposed on the number of computations carried out by $\mathcal{R}$. Thus $\mathcal{R}$ has to compute up to $2tn$ hash functions (each message needs two hashes) and check tn hashes before discarding a tag.

Note that LRMAP uses the first privacy mode—when the executions are not disrupted, and the second privacy mode—when the executions are disrupted, while SZ-LRMAP uses the third privacy mode.

VII. Tradeoff between Privacy and Availability

Pre-updating pseudonyms (third privacy mode) is the only way to get strong anonymity: that is (strong) unlinkability along with (strong) forward secrecy. However, for lightweight applications strong anonymity cannot be achieved in the Byzantine threat model.

Theorem. Hash-based (strong) forward-secrecy for RFID is not possible in the Byzantine threat model.

Proof. By contradiction. We show that any RFID system that supports (strong) forward secrecy does not support availability: a Byzantine adversary can de-synchronize any tag from the readers.

In the Byzantine threat model all RFID parties (tags $\mathcal{T}$, readers $\mathcal{R}$, the server and the adversary $\mathcal{A}^{fs}$) are PPT and the adversary controls the delivery schedule of all communication (Section II). For forward secrecy we require that: $\forall$ PPT $\mathcal{T}$; $\forall$ PPT $\mathcal{R}$; $\forall$ interrogations int (partial or completed); $\nexists$ a PPT adversary $\mathcal{A}^{fs}$ that after first being allowed to interact arbitrarily with the tags and readers and then given access to the state of $\mathcal{T}$, can decide with advantage $Adv_{\mathcal{A}^{fs}}$ better than negligible (Equation (2)) whether $\mathcal{T}$ was involved in int. This means that for forward secrecy, all tags $\mathcal{T}$ must pre-update (the interrogation int may have been interrupted), and all state updates must be one-way.

Let $\mathcal{R}$ be a PPT reader bounded by polynomial p_R. Then there is a PPT adversary $\mathcal{A}^{fs}$ bounded by polynomial p_A with $deg(p_A) > deg(p_R)$ that can force a tag $\mathcal{T}$ to update its state to a state beyond the bound of $\mathcal{R}$ (by querying it p_A times). Since the state updates are one-way, this permanently de-synchronizes $\{\mathcal{T}, \mathcal{R}\}$ and we lose availability. □

Note that the definition of forward secrecy requires protection against *any* PPT adversary: whatever the threshold value t and computation capability of the RFID reader ($tn(2h + c)$), there is a PPT adversary $\mathcal{A}^{fs}$ whose computational capability is higher. The theorem does not apply to session forward secrecy. Indeed LRMAP supports unlinkability and session forward secrecy (Section VII).

We conclude by observing that the impossibility of forward secrecy is not restricted to RFID systems: it extends to any lightweight authentication system. However public-key mechanisms *will* provide (strong) forward secrecy. Therefore the trade-off applies only to symmetric-key systems.

VIII. Conclusion

We have shown that there is a trade-off between privacy and availability for hash-based RFID authentication protocols. We compared the protocols LRMAP and the recently proposed SZ-LRMAP and found that the latter addresses a privacy issue of LRMAP at the expense of availability. In particular, an adversary can force a tag to update beyond a threshold t, causing permanent desynchronization. If the value of the threshold t is set very high, then the adversary can perform a DoS attack on the reader, by impersonating tags and sending junk queries. The reader will have to perform αt ($\alpha \geq 2$), hash operations before discarding each one of these. Finally we analyzed the different hash-based authentication modes and showed that for lightweight systems we cannot achieve strong anonymity (unlinkability and forward secrecy) in the Byzantine threat model.

References

[1] European Comission, “Commission launches consultation on Radio Frequency Identification (RFID).” http://ec.europa.eu/digital-agenda/en/news/commission-launches-consultation-radio-frequency-identification-rfid.

[2] The European Parliament and the Council of the European Union, “Directive 95/46/EC.”

[3] N. Gohring, “California Makes It a Crime to ’skim’ RFID Tags,” PC-World, October 2008.

[4] G. Avoine, http://www.avoine.net/rfid/, 2013.

[5] D.-Z. Sun and J.-D. Zhong, “A hash-based RFID security protocol for strong privacy protection.” *IEEE Trans. Consumer Electronics*, vol. 58, no. 4, pp. 1246–1252, 2012.

[6] J. Ha, J. Ha, S. Moon, and C. Boyd, “LRMAP: Lightweight and Resynchronous Mutual Authentication Protocol for RFID System,” *Proc. 1st Int. Conf. Ubiquitous Convergence Technology*, ICUCT’06. Springer-Verlag, 2007, pp. 80–89.

[7] K. Finkenzeller, *RFID Handbook : Fundamentals and Applications in Contactless Smart Cards and Identification*, 2nd ed. John Wiley & Sons, May 2003.

[8] International Organization for Standardization, “ISO/IEC 29192-1:Information Technology- Security Techniques - Lightweight cryptography - Part 1: General. ISO/IEC, 2012.”

[9] M. Burmester, T. van Le, and B. de Medeiros, “Provably Secure Ubiquitous Systems: Universally Composable RFID Authentication Protocols,” *Conf. Security and Privacy in Communication Networks (SecureComm)*. IEEE, 2006, pp. 1–9.

[10] M. Burmester and J. Munilla, “Lightweight RFID authentication with forward and backward security,” *ACM Transactions on Information and System Security*, vol. Vol. 14, 2011.

[11] J. Ha, S. Moon, J. Zhou, and J. Ha, “A new formal proof model for RFID location privacy,” *Proc. 13th European Symposium on Research in Computer Security: Computer Security*, ESORICS ’08. Springer-Verlag, 2008, pp. 267–281.

[12] T. van Deursen and S. Radomirovic, “On a new formal proof model for RFID location privacy,” *Information Processing Letters*, vol. 110, no. 2, pp. 57 – 61, 2009.

[13] I. Erguler, M. Akgun, and E. Anarim, “Cryptanalysis of a lightweight RFID authentication protocol- lrmap,” *Proc. Western European Workshop on Research in Cryptology*, WEWoRC’09, 2009.

[14] M. Ohkubo, K. Suzuki, and S. Kinoshita, “Cryptographic Approach to “Privacy-Friendly” Tags,” *RFID Privacy Workshop*, MIT, Massachusetts, USA, November 2003.

[15] T. Dimitriou, “A lightweight RFID protocol to protect against traceability and cloning attacks,” *Proc. IEEE Intern. Conf. on Security and Privacy in Communication Networks (SECURECOMM 2005)*. IEEE Press, 2005.

[16] G. Avoine, E. Dysli, and P. Oechslin, “Reducing Time Complexity in RFID Systems,” *Selected Areas in Cryptography – SAC 2005*, Lecture Notes in Computer Science, B. Preneel and S. Tavares, Eds., vol. 3897. Springer-Verlag, August 2005, pp. 291–306.