Privacy versus Scalability in Radio Frequency Identification Systems

August 6, 2010

Basel Alomair and Radha Poovendran
Network Security Lab
University of Washington-Seattle
{alomair,rp3}@uw.edu

Abstract

Embedding a Radio Frequency Identification (RFID) tag into individual items enables the unique identification of such items over the wireless medium, without the need for a line-of-sight path. One of the main challenges for the successful commercialization of the RFID technology is the efficient, yet private, identification of low-cost tags in the presence of adversaries attempting to illegally track users via tags in their possession. An RFID system consists of two functional components, namely, the interactive protocol between RFID reader-tag pairs and the reader-database information retrieval mechanism. Because of the large number of tags in a typical RFID system, the private identification of tags can be a challenging problem. In this paper, we investigate privacy-preserving RFID systems and classify them based on the computational efficiency of tag identification. We show the close relation between the degree of privacy achieved by the reader-tag interaction and the reader-database information retrieval complexity.

Keywords: Radio Frequency Identification (RFID), scalability, privacy, passive, active

1 Introduction

1.1 RFID systems

Typically, Radio Frequency Identification (RFID) systems are composed of three main components: tags, readers, and a database. An RFID tag is a small device that can be attached to products and allow for unique item identification and product description. RFID tags can be battery powered (active) or powerless (passive). Passive RFID tags (which are the main emphasis of this paper) are very cheap devices with limited memory and limited computational capabilities. An RFID reader, on the other hand, is a computationally powerful device with ability to interrogate tags and access the database, where information about individual tags and their corresponding items is stored.

When an RFID tag is within communication range of an RFID reader, the reader interrogates that tag (and powers it if it is passive). Upon interrogation, the tag responds with

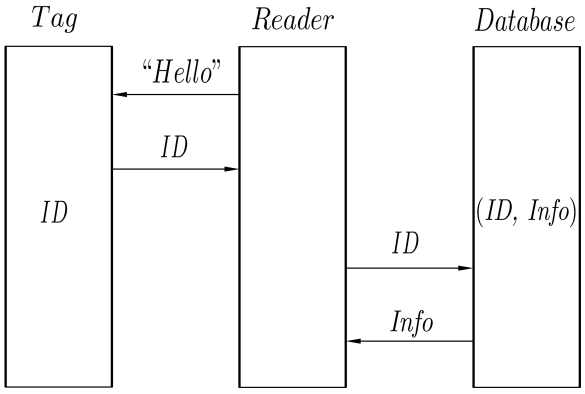

Figure 1: A simple identification protocol for reading the ID of an RFID tag within a communication range of an RFID reader. The RFID reader broadcasts a "$Hello$" message to announce. The RFID tag responds to the reader's request by transmitting its unique ID. The tag's unique ID is then used by the reader to lookup the database for information related to the item carrying the RFID tag.

a quantity that allows legitimate readers to access the database and carry out the identification process. If things work as planned, the reader should be able to uniquely identify the interrogated tag.

The specific details of the identification process can vary dramatically from one protocol to another, depending on the targeted applications and the security assumptions on the underlying environment. In its simplest form, identification can be as straightforward as sending unique identifiers in clear text. Figure 1 depicts an instance of a simple identification run. The RFID reader interrogates the tag by sending a "Hello" message. The tag responds with its unique identifier in clear text. The reader can then access the database to obtain information about the tag and the item carrying it.

1.2 The RFID Controversy

Compared to traditional means of identification, RFID tags possess a unique property that is making the deployment of RFID systems in everyday life a highly controversial issue. RFID tags respond to readers' queries via the wireless medium, no line-of-sight is required as in traditional identification processes (e.g., ID cards or barcodes). Consequently, the identity of RFID tags, and ultimately their owners, can be revealed to unauthorized parties without the owners' approval nor even their awareness.

Privacy activists have been concerned about the invasion of users' privacy by RFID tags, calling for the delay or even the abandonment of their deployment (naming RFID tags "the spy chips" [1]). In some cases, companies have been forced to repudiate their plans for RFID deployment in response to the threat of being boycotted [27]. Consequently, providing private identification for RFID systems has been an attractive problem for both academic and industrial researchers.

1.3 Private Identification

In addition to the primary objective of RFID systems, identification, there are two secondary goals that most RFID systems aim to satisfy: privacy and security. The distinction between the primary and secondary goals could explain the market failure of privacy friendly solutions for RFID systems.

The simple scheme in Figure 1 is clearly a violation of users' privacy. A person carrying an item equipped with an RFID tag, a watch or a jacket for example, can be tracked down by the tag he/she is carrying. A rogue reader interrogating a tag multiple times, and receiving the same identifier as in the basic scheme of Figure 1, will be able to correlate the tag's responses and ultimately identifying the person carrying the tag, without his/her awareness. Furthermore, the scheme of Figure 1 is also a security violation. Users listening to the same radio channel can record the identity of the tag and illegally impersonate that tag later. For instance, a tag used for access control can be easily cloned, granting access to unauthorized, possibly malicious, persons. Therefore, in a typical RFID system, tag authentication is also a basic requirement.

There are two complementing ends in any radio frequency identification system:

1. the interactive protocol between authorized RFID reader-tag pairs,

2. the interaction between RFID readers and the database for data retrieval.

The reader-tag interactive protocol usually involves, in addition to identification, tag authentication or mutual authentication, depending on the specification of the protocol. The data retrieval mechanism is nontrivial since, to satisfy the privacy requirement, tags' information cannot be transmitted in clear text. That is, in one hand, tags' responses must not reveal private information to unauthorized observers and, on the other hand, it must enable authorized readers to access the database and retrieve tags' information.

In most papers in the literature of private RFID systems, those two complementing ends have been treated independently. In fact, the majority of RFID papers study only the interaction between RFID reader-tag pairs, either by proposing new protocols or analyzing existing ones. This is not surprising because it deals with the most challenging issue in RFID designs. The fact that RFID tags are, in most applications, low-cost devices with stringent computational capabilities makes the use of sophisticated cryptographic primitives proven to achieve private identification and secure authentication impractical. In fact, several attempts have been made to come up with cryptographic primitives designed specifically for RFID tags (see, e.g., [10, 23, 25, 28, 68]).

However, the fact that both readers and the database are computationally powerful devices able to establish secure channels[1] does not make the reader-database interaction an easy one. As will be detailed in this work, the computational limitation of RFID tags, in addition to its direct implications on the reader-tag interaction, will also have a significant impact on the reader-database information retrieval process.

[1] Secure channels enable users to communicate confidentially while maintaining message integrity. Interested readers can find more about establishing secure channels in [8, 41].

1.4 Why is Private Identification Hard?

Individually, the three objectives of an RFID system, identification, privacy, and security, can be achieved relatively easily. Reaching all three objectives simultaneously, however, is a challenging task. For instance, identification, by itself, can be as easy as transmitting tags' identifiers in clear text, as in the basic scheme of Figure 1. If tags' privacy is of any importance, however, transmitting tags' identifiers in clear text is obviously unacceptable.

If tags are not to be traced by unauthorized observers, their responses must be randomized in a way that cannot be correlated to tags by *unauthorized users*. For computationally powerful devices, the literature of cryptography is rich with ready-to-implement solutions for the private identification problem [50, 72]. For instance, a tag can encrypt a randomized version of its identifier with the reader's public key, provided the existence of a public key infrastructure. Only an authorized reader, with knowledge of the private key, can decrypt the message and extract the tag's identity. In most practical implementations of RFID systems, however, passive tags have limited computational power and performing public key operations is considered beyond their computational capabilities. Consequently, most RFID systems are restricted to the use of symmetric-key cryptography to provide security and privacy.

Since RFID tags are not tamper-resistant, each tag must be loaded with a unique secret key (otherwise, the physical capture of a single tag can break the security of the entire system). Consequently, symmetric-key privacy-preserving RFID systems are faced with the following paradox. In one hand, private identification requires that tags disguise (encrypt) their identities with their *secret keys*. On the other hand, for the reader to extract (decrypt) the tag's identity, it must know the identity of the tag first (in order to retrieve the secret key required for decryption). Consequently, depending on the required degree of privacy and the specifics of the interactive protocol between reader-tag pairs, searching the database to identify tags based on their responses is a nontrivial task.

In this paper, we investigate the implications of the achieved degree of privacy and the specifics of the interactive reader-tag protocol on the identification process. In the literature, one can find good survey papers on RFID systems (see, e.g., [34, 61, 65, 69, 77, 81]). However, unlike previous work, we focus here on an important aspect of RFID systems that has not been discussed in detail previously. Namely, we provide an up-to-date study of RFID systems, emphasizing the effect of the reader-tag interaction protocol on the computational effort expended by the database. We give two classifications of existing symmetric-key RFID systems: one based on the reader-tag interaction protocols, and the other is based on the time complexity of identifying tags' responses. The reader-tag interaction protocol can be either stateful or stateless. The identification process can be accomplished in constant-time, linear-time, or logarithmic-time (as functions of the number of tags in the system). We provide a generic treatment of these classes to identify the properties that distinguish them from one another. As opposed to previous work, we do not study specific protocols. Instead, we study the general classes and give examples of protocols appeared in the literature in each class.

2 Degrees of Privacy

In this section, we define two degrees of privacy for RFID tags that are essential for our analysis of RFID systems. Tags' privacy is characterized by the ability of unauthorized users to trace RFID tags using their responses to readers' interrogations. Two notions of untraceability are

defined as follows:

Definition 1 (Universal Untraceability) *An RFID tag is said to be universally untraceable if two tag's responses, separated by a successful identification run with a valid reader, cannot be correlated with high confidence by unauthorized users.*

A stronger notion of privacy for RFID tags is the notion of existential untraceability, defined as follows.

Definition 2 (Existential Untraceability) *An RFID tag is said to be existentially untraceable if two tag's responses, not necessarily separated by a successful identification with a valid reader, cannot be correlated with high confidence by unauthorized users.*

The main difference between the two definitions is allowing the tag to complete a successful protocol run with an authorized reader. Another, and probably more intuitive, interpretation of Definition 1 is that it implies privacy against *passive adversaries* only. To see this, observe that an active adversary can interrogate the same tag twice in the absence of authorized readers, thus observing two responses that are not separated by a valid protocol run. On the other hand, Definition 2 extends tags' privacy to hold against *active adversaries* too.

The differentiation between the two definitions is important in analyzing RFID protocols. Intuitively, protecting the privacy of RFID tags against active adversaries is more difficult than providing privacy against passive adversaries. Whenever is convenient, we will refer to protocols that provide universal untraceability only as *passively private*, while protocols that provide existential untraceability as *actively private*.

There is yet another important notion of privacy in RFID systems called forward untraceability. Since RFID tags are typically not tamper-resistant, an adversary can capture a tag and expose its secret parameters. Informally, forward security requires that an adversary exposing a certain tag's secret parameters cannot correlate past instances of the tag's responses. Since the main purpose of this paper is to study the effect of privacy on identification complexity, and since forward untraceability does not contribute to our classification of RFID protocols, we omit the detailed discussion of forward untraceability in this paper (interested readers may refer to, e.g., [4, 9, 20, 42] for more discussion).

In what follows, we classify RFID protocols based on the reader-tag interaction into stateful and stateless protocols, and examine the characteristics of each class.

3 Stateful Protocols

As can be inferred from the names, tags in stateful protocols are identified via a unique state they possess (a temporary pseudonym or a position in a data structure, etc.). For successful identification, each tag's state must match the state stored at the database; otherwise, tags cannot be identified successfully. This means that tags must always be synchronized with the database for such protocols to function properly (stateful protocols can also be called "synchronous" protocols).

In what follows, we give a general description of stateful protocols. Obviously, different protocols have different properties, but we describe here the common characteristics of protocols belonging to this class.

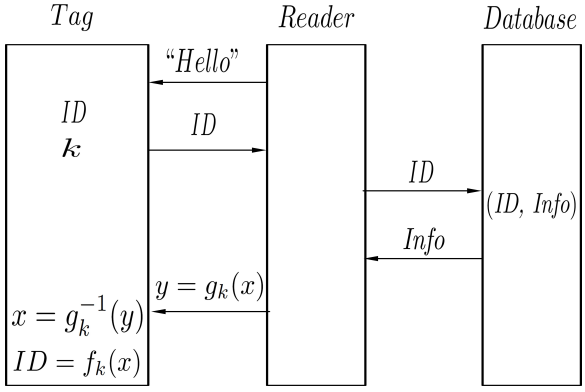

Figure 2: An instance of protocols based on updated identifiers. The reader interrogates the tag by sending a "*Hello*" message. The tag responds with its current identifier ID. Upon identifying the tag and obtaining its secret key, the reader generate a random number x and sends $g_k(x)$ to the tag, where g_k a function that is determined by the tag's secret key k. The tag inverts the function g_k to compute the random number x and then use it to update its identifier $ID = f_k(x)$.

3.1 General Description

Each tag is loaded with a unique state (call it ID) and a secret key k. Upon interrogation by a reader, the tag responds with its current ID. Given the tag's ID, the reader can access the database and obtain information about the tag, including the tag's secret key. Although the unique ID is sufficient for identification, the key is necessary for authentication. The main property that differentiates protocols of this class is how the authentication process is carried out. Since the main purpose of this work is the effect of the reader-tag interactive protocol on the identification process, we omit the details of the authentication process (interested readers may refer to [3, 22, 44, 59, 60, 62, 63, 67, 71, 80] for the detailed description of different authentication processes).

Now, with the tag's secret key obtained, the reader generates a random string, x, and evaluates $g_k(x)$, where g is an invertible function and k is the tag's secret key. The reader then transmits $g_k(x)$ to the tag. Given $y = g_k(x)$, the tag inverts y to obtain the value of x; i.e., $x = g_k^{-1}(y)$. With the value of x in hand, the tag and the database update the tag's identifier to $ID = f_k(x)$.

For the next protocol run, the tag can be identified with its updated state. Note that the choice of the functions g and f is an important factor in determining the security of the protocol and the amount of computational power performed by tags. Therefore, the choice of g and f is a nontrivial task in which different stateful protocols differ (see, e.g., [3, 22, 44, 59, 60, 62, 63, 67, 71, 73, 80]). An instance of such protocols is depicted in Figure 2.

3.2 Properties of Stateful Protocols

One of the most important advantages of stateful protocols, especially to this work, is the efficient tag identification. Since, from the standpoint of authorized readers, each tag has

only one possible response, its current state, tags can be identified in *constant-time*. Even if the length of tags' identifiers is sufficiently long (to prevent easy-to-implement attacks such as random guessing and exhaustive search), at which point it would be impractical to build a database that can accommodate all possible identifiers for direct access,[2] existing data structures such as hash tables can be used to allow for constant-time identification.

Observe, however, that tags in this class of protocols will not update their states in the absence of authorized readers. This is critically important to maintain the required synchronization between tags and the database. This implies, however, that tags implementing this class of protocols are only universally untraceable. To see this, note that an active adversary interrogating the same tag twice in the absence of authorized readers will receive the same state, thus, easily correlating the two responses. That is, this class of protocols protects tags' privacy against passive adversaries only.

There is another implication of the fact that tags update their states based on information transmitted by the reader. In most RFID applications, the identity of the interrogated tag must be authenticated. Additionally, in this class of protocols, the reader must also be authenticated. If reader authentication is not required, tags can be easily desynchronized by updating their states based on false information generated by malicious readers.

There is one more advantage of this class of protocols. Since tags are not required to randomize their states internally, there is no need for a random or pseudorandom number generator in the tag's side. This can be particularly important for RFID systems with very cheap tags. The generation of random numbers is critical for the security of most cryptographic protocols. However, standard methods for generating such numbers that have been proven to be secure for general cryptographic protocols are known to be expensive for RFID tags [24]. Consequently, significant efforts have been made to the design of random and pseudorandom generators that can be suitable for RFID tags (see, e.g., [11, 15, 31, 46, 49, 66, 75]). Even with the rapid advances in hardware technology, some designers believe in building tags that are as cheap as possible. Their justification is that, when RFID technology is to replace conventional barcodes for product identification, tags will contribute significantly to the cost of the overall production. For example, a 10-cent tag that can perform provably secure cryptography will add 20% to the price of 50-cent items, not to mention even cheaper items. When retailers are to choose between low cost and higher security, it seems inevitable that cheaper tags will win the race.

3.3 Non-Cryptographic Privacy Enhancements

As mentioned earlier, stateful protocols do not provide privacy against active adversaries. However, they are still attractive for two main reasons: their constant-time identification and their cost-efficient tag implementation. Consequently, significant efforts have been devoted to the design on non-cryptographic techniques to enhance the privacy of tags in stateful protocols.

In [36], Juels *et al.* proposed the use of a blocker tag that blocks readers' interrogation. The blocker tag can simulate many ordinary RFID tags simultaneously to enhance privacy. This can be done universally by simulating all tags, or selectively by simulating a subset of tags. A variant of the blocker tag, called soft-blocking, was also proposed in [35]. Unlike the

[2] If tags' identifiers are 128-bit long, for instance, one will need a database of size proportional to 4×10^{25} terabyte for direct addressing.

blocker tag, soft-blocking provides a weaker privacy enforcement but has the advantage of offering flexible privacy policies.

In [26], Floerkemeier *et al.* proposed the use of a watchdog tag to make users aware of their tags being interrogated. In [64], Rieback *et al.* proposed the use of an RFID guardian, a battery powered device that protects tags from being illegally scanned. In [37], Juels *et al.* proposed the use of an RFID Enhancer Proxy (REP). The REP is more computationally powerful than tags and can enforce more sophisticated privacy policies.

In the next section, we discuss protocols that are designed to enhance tags' privacy through cryptographic solutions.

4 Stateless Protocols

Unlike stateful protocols, tags in stateless protocols have the ability to randomize their responses internally. To enhance the tag's privacy, upon readers interrogations, the tag will generate a random string and respond with a quantity that should not be correlated to is previous response by unauthorized readers. However, although randomized responses can enhance tags' privacy, they make the identification process more complicated. As opposed to stateful protocols, each tag now has more than one possible response. In fact, in most systems, the set of all possible responses of each tag is exponential in the length of its response. Therefore, even for authorized readers, tag identification is not as simple as a constant-time lookup.

In this section, we address stateless RFID protocols. Depending on the computational effort expended at the database in order to identify tags' responses, we categorize the class of stateless protocols into two subclasses: linear-time and logarithmic-time identification protocols.

4.1 Linear-time Identification Protocols

We first give a general description of linear-time identification protocols and then discuss their properties.

4.1.1 General Description

In a typical protocol of this class, each tag possesses a unique ID and is equipped with a random (pseudorandom) number generator. When a tag is in the vicinity of an RFID reader, the reader interrogates the tag by sending a random nonce, r_1. Using its own random number source, the tag generates another nonce, r_2, and computes $h(ID, r_1, r_2)$, the hash of its identifier concatenated with r_1 and r_2. The tag responds with the concatenation of r_2 and the resulting hash value, i.e., $s = (r_2, h(ID, r_1, r_2))$.[3]

Upon receiving the response $s = (r_2, h(ID, r_1, r_2))$, the reader extracts r_2 and computes $h(ID, r_1, r_2)$ for the list of ID's in the system. The tag with the identifier that matches $h(ID, r_1, r_2)$ is the interrogated tag. Figure 3 depicts an instance of such protocols. The details of different protocols in this class vary according to the specifics of the protocol. Linear-time identification protocols include, but are not limited to, [17, 18, 30, 56, 57, 70, 78].

[3]Some protocols use pseudorandom functions instead of hash function, but this is the main idea of this class of protocols.

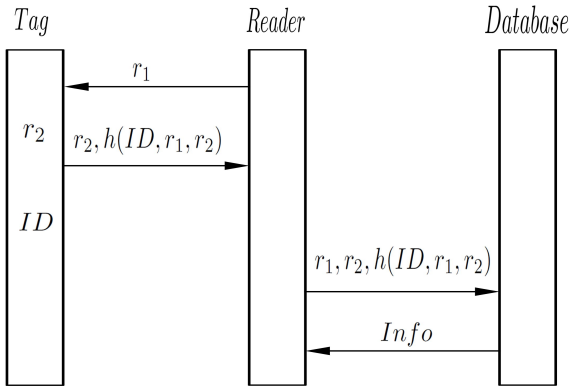

Figure 3: A schematic of an instance of the linear-time protocols. The reader sends a random number r_1 to the tag, which responds with the hash of its identifier concatenated with r_1 and r_2. The reader searches all possible ID's in the database for the one that matches the received response.

4.1.2 Properties of Linear-Time Identification Protocols

Identification inefficiency is the most important drawback of this class of protocols. Recall that each tag's response can take any value in the range of the hash function, as opposed to a single possible response per tag in the stateful protocols of Section 3. Observe also that, since cryptographic hash functions are noninvertible, the authorized reader has no means of identifying the tag except the exhaustive search. That is, to identify a single response, the database must hash the two nonces r_1 and r_2 with all possible ID's until a match is found. (Even if an invertible function such as an encryption algorithm is used, the reader must try to decrypt the response with all possible secret keys in the system until a match is found.) If N is the number of tags in the system, the database is expected to perform an average of $N/2$ hash operations before it can identify the tag. Thus, the complexity of the identification process is $O(N)$. We say that this class of protocols is a *linear-time* identification class.

The main purpose behind the introduction of such protocols is to overcome the privacy issue of stateful protocols. Observe that tags can randomize their responses without the help of authorized readers. Therefore, an adversary interrogating the same tag twice in the absence of valid readers will receive two responses that cannot be correlated. In other words, tags implementing this class of protocols are existentially untraceable.

Another property of this class of protocols is that there is no need for mutual authentication. This is a direct consequence of the tags' ability to randomize their responses internally. Observe that tags in such protocols do not use information generated by the reader to update their parameters, unlike stateful protocols. Furthermore, responding to malicious interrogation does not reveal tags' identities. Therefore, the tag can respond to any interrogation, whether valid or not, without authenticating the reader and without undermining the privacy of the tag nor the correctness of the protocol.

Typically, in a protocol of this class, tags' responses require one random number generation and one hash operation. Thus, a typical protocol of this class requires $O(1)$ communication and computational overhead on the tags' side, with $O(N)$ computational effort on the database (where N is the total number of tags in the system).

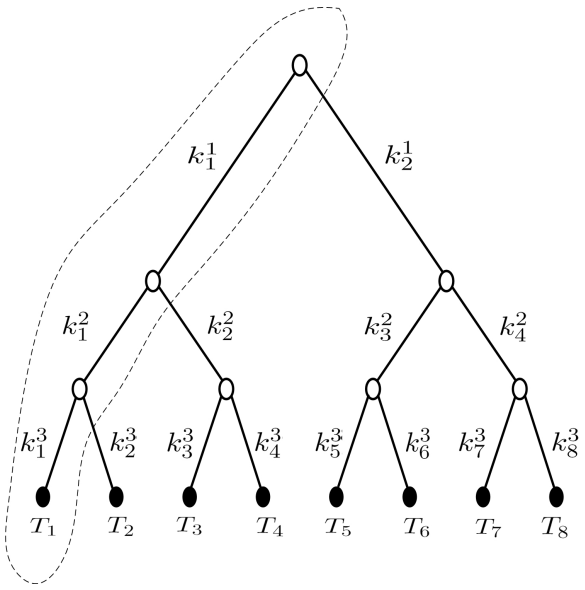

Figure 4: An example of a tree-based RFID system with eight tags. Each edge in the tree corresponds to a unique secret key and each tag corresponds to a unique leaf. Tags are identified via the set of secret keys corresponding to the path from the root of the tree to the tag's leaf.

There is yet another consequence of the fact that the database must perform a linear search to identify tags' responses. In a system with a large number of tags, an adversary can make the database busy searching for false responses. That is, linear-time protocols are vulnerable to denial of service attacks.

The second class of protocols is the *logarithmic-time* identification class, where tag identification requires searching time proportional to the logarithm of the number of tags in the system.

4.2 Logarithmic-time Identification Protocols

An important step in the direction of solving the scalability issue in RFID systems, while still protecting tags' privacy against active adversaries, was proposed by Molnar and Wagner in [52]. In what follows, we give a general description of this class of protocols and then discuss its properties.

4.2.1 General Description

In this class of protocols, tags are arranged in a tree data structure. Each edge of the tree corresponds to a secret key, and each tag corresponds to a unique leaf in the tree. The secret key of each tag is the set of keys corresponding to the path from the root of the tree to the tag's leaf. Figure 4 depicts an instance of a constructed tree with eight tags using a branching factor of two. In the example of Figure 4, tag T_1 possesses the set of keys $K = \{k_1^1, k_1^2, k_1^3\}$, and K becomes its unique identifier.

To identify a tag, say T_1, the reader sends a random number r_1. The tag responds with

$s = (r_2, h(k_1^1, r_1, r_2))$, where r_2 is a random number generated by the tag. Using the tag's response, the reader can determine whether the tag belongs to the left or right side of the tree by computing $h(k_1^1, r_1, r_2), h(k_2^1, r_1, r_2)$ and looking for a match to the tag's response (in this example, T_1 belongs to the left of the tree). By performing the same procedure for every level of the tree, the tag can be uniquely identified by the set of keys, K, it possesses (the $O(\log N)$ interactions can be combined into one interaction of size $O(\log N)$). Tree based protocols include, but are not limited to, [12, 13, 19, 47, 48, 51, 76].

4.2.2 Properties of Logarithmic-Time Identification Protocols

The main reason behind using the tree data structure is to solve the scalability issue of linear-time identification protocols. For a system of N tags and a branching factor b, tags can be identified in $O(\log_b N)$ time. This is a significant improvement over the $O(N)$ performance of the previous class of protocols. We say that this class of protocols is a *logarithmic-time* identification class.

Just like the linear-time class, protocols based on tree structures can provide the required privacy against active adversaries. This is a direct consequence of the fact that tags' responses can be randomized internally, without the need for readers assistance.

In general, tree based protocols as originally proposed in [52] do not require mutual authentication. Just like linear-time protocols, this is due to the fact that tags need not to update their states based on information delivered by the reader, while still maintaining privacy against active adversaries.

As can be inferred from the description of logarithmic-time protocols, the improvement in identification complexity came by trading-off communication and computational efficiency at the tags' side. That is, unlike linear-time protocols, logarithmic-time protocols requires $O(\log N)$ communication and computation overhead. Although increasing the computational effort on tags' side to $O(\log N)$ might not have a considerable effect, the increase in the communication overhead to $O(\log N)$ can have noticeable impact on the practicality of the system. In a typical RFID application, readers interrogate multiple tags simultaneously. Therefore, even in the linear-time protocols where the communication overhead is $O(1)$, collision avoidance and medium access control (MAC) are amongst the most challenging problems in practical RFID systems (see, e.g., [7, 33, 39, 40, 53]). Increasing tags' responses to $O(\log N)$ can only complicate collision avoidance and MAC even further.

Compared to constant-time and linear-time protocols, the logarithmic-time class of protocols possess a unique vulnerability to tag compromise attacks. Due to the importance of this vulnerability, we discuss the details of it in the following section.

4.2.3 The Tag Compromise Vulnerability

In [5], Avoine *et al.* discovered a new security threat in logarithmic-time protocols. The observation they made is that arranging tags in a tree, based on their secret keys, implies that different tags will share secret information depending on their positions in the tree. Since RFID tags are typically not tamper-resistant, compromising a subset of tags in the system will reveal secret information about other, uncompromised, tags.

Consider an adversary capturing T_1 in Figure 4, thus obtaining $\{k_1^1, k_1^2, k_1^3\}$. Consider now the adversary interrogating T_2 and observing its response and then interrogating T_5 and

Table 1: Performance comparison of the three classes of existing RFID protocols as a function of the number of tags in the system, N. Note that logarithmic-time protocols require mutual authentication and are vulnerable to desynchronization attacks if they employ key-updating mechanisms.

	Privacy	Identification	Key size	Communication	Authentication	Desynchronization	Tag capture
Constant	Passive	$O(1)$	$O(1)$	$O(1)$	Mutual	Vulnerable	Secure
Linear	Active	$O(N)$	$O(1)$	$O(1)$	Tag	Secure	Secure
Logarithmic	Active	$O(\log N)$	$O(\log N)$	$O(\log N)$	Tag/Mutual*	Secure/Vulnerable*	Vulnerable

observing its response, with the goal of determining whether they are the same tag or not. Since the first key of T_2 is k_1^1 and the first key of T_5 is k_2^1, the adversary can distinguish between them, even though neither T_2 nor T_5 have been compromised. In general, by compromising a single tag in the system, the adversary can always distinguish between two tags that belong to opposite halves of the tree. In fact, Avoine *et al.* analyzed the information leakage by tag compromise attacks in tree based scheme showing that, by compromising 20 tags in a system of 2^{20} tags, an adversary can trace uncompromised tags with an average probability close to one [5]. Quantifying the amount of information leakage is tree-based protocols has also been studied in, e.g., [32, 54].

Researchers who believe that the reduction in identification complexity is significant and cannot be overlooked as a result of the new vulnerability it introduces have proposed techniques to mitigate the effect of tag compromise attacks (see, e.g., [47, 48, 76]). The main concept shared by all such techniques is to employ dynamic key-updating mechanisms. Obviously, different protocols employ different key updating mechanisms and, hence, their effectiveness in mitigating tag compromise attacks vary. Note, however, that such updating mechanism imposes an extra requirement on such protocols. Namely, unlike standard tree-based approaches, the update procedure requires a mutual reader-tag authentication, and a proper protection against desynchronization attacks.

A comparison of different classes of identification protocols is summarized in Table 1. Before we conclude the paper, we will describe recently proposed protocols that have different properties than the general classes described previously.

5 Recent Advances

In this section, we discuss some recent advances in the design of RFID systems.

5.1 Advances in Stateful Protocols

In [38], Tsudik proposed a protocol based on monotonically increasing time stamps. The database maintains a periodically updated lookup table in which each row corresponds to a certain tag. Adopting the same idea of time stamps, the protocols in [14, 21, 74] further provided some improvements over the original scheme of [73]. Such protocols can achieve constant-time identification. However, time-stamp based protocols have been analyzed and shown to lack some security requirements (see, e.g., [38, 43, 45, 58]).

In [2], Alomair *et al.* proposed a new stateful protocol that can allow for constant-time identification while maintaining tags' privacy against active adversaries. The main concept

of their protocol is limiting the number of "consecutive" interrogations by an active adversary, say $1,000,000$ interrogations. Their justification for limiting the number of consecutive interrogations by the adversary is the following. Unlike general computer communication systems, a high number of successive interrogation can be difficult to achieve in practical RFID systems. A web server, for instance, is always online and can be attacked from long distances. For an RFID tag to be interrogated, on the other hand, the adversary must be in close proximity to the tag. However, if the adversary is always in close proximity to the tag, the tag can be tracked without interrogation, e.g., visually. Therefore, the typical goal of adversaries in RFID systems is to identify tags that have not been always in their vicinity. A brief description of the protocol is as follows.

Instead of incurring more communication overhead as in tree-based protocols, the protocol in [2] trades-off a larger storage and utilizes an offline computations of tags' responses. Using a three-tier hash table data structure, the database can identify interrogated tags in constant-time.

In a system with a million tags, the required size of the database is no more than 12 terabyte, while protecting tags' privacy against up to a million consecutive interrogations [2]. That is, for an active adversary to be able to identify a tag, the tag must be interrogated more than one million times not separated by a single interrogation by a valid reader. Obviously, whether that many consecutive runs grants reasonable security or not depends on the response time of tags, which, in turn, depends on the protocol and the computations performed by tags. A typical response time for a passive EPC tag is in the order of 4 milliseconds [82]. This implies that an active adversary must spend 67 minutes of consecutive interrogations with the same tag to correlate its responses. To enhance security even further, the tag can increase its response time as a function of the number of consecutive incomplete protocol runs.

5.2 Advances in Stateless Protocols

In an attempt to solve the scalability issue in linear-time RFID systems, few techniques have been proposed recently. In [6], Avoine and Oechslin proposed the use of a time-memory trade-off based on the work of Hellman [29] and Oechslin [55]. The goal of such trade-off techniques is to reduce the amount of computational effort needed to invert a one-way function. The main concept of such techniques is chaining almost all possible outputs of the one-way function using a reduction function that generates an arbitrary input of the one-way function using one of its outputs. By alternating the one-way function and the reduction function on a specific value, a chain of inputs and outputs of the one-way function is constructed. Given the generation of a sufficient number of chains, most outputs of the one-way function will appear at least once. To invert an output of the one-way function, a chain started with that output is built. The idea is to search the stored chains for a match, which can lead to finding the input corresponding to the desired output. By adapting the technique in [55], identification complexity can be reduced to $O(N^{2/3})$, while maintaining the same security and privacy levels of the linear-time protocols [6].

In [16] Cheon *et al.* proposed a meet-me-in-the-middle strategy to improve the efficiency of the identification process. The basic idea is to have two sets of keys $\mathcal{K}_1$ and $\mathcal{K}_2$, each of size $\sqrt{N}$ where N is the number of tags in the system. Each tag T_i is loaded with two keys k_1^i and k_2^i, where $k_1^i \in \mathcal{K}_1$ and $k_2^i \in \mathcal{K}_2$. When a tag is interrogated with a random string r, it responds with r' and $C = PRF_{k_1^i}(r, r') \oplus PRF_{k_2^i}(r, r')$, where PRF is a pseudorandom

function and r' is a random string generated by the tag.

With the received (r', C), the reader constructs a table with entries $\left(k_1^i, PRF_{k_1^i}(r, r')\right)$, for each $k_1^i \in \mathcal{K}_1$ and sorts the table. Then, for each $k_2^i \in \mathcal{K}_2$, the reader computes $C \oplus PRF_{k_2^i}(r, r')$ and searches for it in the table. The tag with a (k_1^i, k_2^i) that corresponds to the match is the interrogated tag. The complexity for identifying the tag is $O(\sqrt{N} \log N)$. Although the idea is novel, there are two concerning issues about the protocol. First, just like tree-based protocols, this protocol has a tag compromise vulnerability. In particular, compromising t tags in the system will enable the adversary to identify $t^2 - t$ uncompromised tags [16]. Second, with identification complexity of $O(\log N)$, tree-based protocols are more efficient than the $O(\sqrt{N} \log N)$ of this protocol.

In [79], Wu and Stinson proposed a privacy-preserving protocol based on polynomial operations over the finite field $\mathbb{F}_{2^\ell}$. The security of the protocol is based on the difficulty of reconstructing a polynomial with noisy data. To be able to perform the required computations, tags can be designed with about $10,000$ gates, whereas ECC based public-key computations might require around $20,000$ gates [79].

What is more relevant to this paper, however, is the identification complexity. In the protocol of [79], the database needs to solve mb polynomials of degree k, where m, b, and k are predefined security parameters. Therefore, although asymptotically constant, the identification operation can be cumbersome. Typical numbers of m and b are 16 and 8, respectively [79]. Consequently, the database will perform 128 operations to identify a tag's response. This is equivalent to the required computational effort in a tree-based protocol with a system of 2^{128} tags and a branching factor of 2.

6 Conclusion

In this paper, we studied symmetric-key, privacy-preserving RFID systems. Existing RFID systems are classified based on the amount of computational overhead incurred on the database to identify tags' responses. Identifying tags while preserving their privacy against passive adversaries is shown to be achieved in constant-time, independent of the number of tags in the system. Identifying tags, while preserving their privacy against active adversaries, however, is a more difficult task. Based on their efficiency on identifying tags' responses, actively private protocols are categorized into linear-time and logarithmic-time protocols. Linear-time identification protocols are the ones requiring database search time proportional to the number of tags in the system. They are shown to be secure but impractical for large-scale systems. Logarithmic-time protocols are those requiring database search time proportional to the logarithm of the number of tags in the system. Obviously, logarithmic-time protocols better suit large-scale systems but are vulnerable to tag compromise attacks, where capturing tags in the system undermines the security of uncompromised tags.

Active research problems include the design of non-cryptographic solutions to enhance the privacy of stateful protocols, the search for solutions to the tag compromise vulnerability in the tree based protocols, and the design of actively private protocols that can beat the linear-time complexity without introducing new vulnerabilities.

References

[1] K. Albrecht and L. McIntyre. Consumers Against Supermarket Privacy Invasion and Numbering – CASPIAN. `http://www.spychips.com/`.

[2] B. Alomair, A. Clark, J. Cuellar, and R. Poovendran. Scalable RFID Systems: a Privacy-Preserving Protocol with Constant-Time Identification. In *the 40th Annual IEEE/IFIP International Conference on Dependable Systems and Networks – DSN'10*, Chicago, Illinois, USA, June 2010. IEEE.

[3] B. Alomair, L. Lazos, and R. Poovendran. Securing Low-cost RFID Systems: an Unconditionally Secure Approach. In *Journal of Computer Security - Special Issue on RFID System Security*, 2010.

[4] G. Avoine. Adversary Model for Radio Frequency Identification. Technical Report LASEC-REPORT-2005-001, Swiss Federal Institute of Technology (EPFL), Security and Cryptography Laboratory (LASEC), Lausanne, Switzerland, September 2005.

[5] G. Avoine, E. Dysli, and P. Oechslin. Reducing time complexity in RFID systems. *Selected Areas in Cryptography–SAC*, 3897:291–306, 2005.

[6] G. Avoine and P. Oechslin. A Scalable and Provably Secure Hash Based RFID Protocol. In *International Workshop on Pervasive Computing and Communication Security – PerSec 2005*, pages 110–114, Kauai Island, Hawaii, USA, March 2005. IEEE, IEEE Computer Society Press.

[7] G. Avoine and P. Oechslin. RFID Traceability: A Multilayer Problem. In A. Patrick and M. Yung, editors, *Financial Cryptography – FC'05*, volume 3570 of *Lecture Notes in Computer Science*, pages 125–140, Roseau, The Commonwealth Of Dominica, February–March 2005. IFCA, Springer-Verlag.

[8] M. Bellare and C. Namprempre. Authenticated Encryption: Relations among Notions and Analysis of the Generic Composition Paradigm. *Proceedings of the 6th International Conference on the Theory and Application of Cryptology and Information Security: Advances in Cryptology–ASIACRYPT*, pages 531–545, 2000.

[9] C. Berbain, O. Billet, J. Etrog, and H. Gilbert. An Efficient Forward Private RFID Protocol. In E. Al-Shaer, S. Jha, and A. D. Keromytis, editors, *16th ACM Conference on Computer and Communications Security – CCS'09*, pages 43–53, Chicago, Illinois, USA, November 2009. ACM, ACM Press.

[10] A. Bogdanov, L. R. Knudsen, G. Leander, C. Paar, A. Poschmann, M. J. Robshaw, Y. Seurin, and C. Vikkelsoe. PRESENT: An Ultra-Lightweight Block Cipher. In P. Paillier and I. Verbauwhede, editors, *Workshop on Cryptographic Hardware and Embedded Systems – CHES 2007*, volume 4727 of *Lecture Notes in Computer Science*, pages 450–466, Vienna, Austria, September 2007. Springer-Verlag.

[11] L. Bolotnyy and G. Robins. Randomized pseudo-random function tree walking algorithm for secure radio-frequency identification. In *Proc. IEEE Workshop on Automatic Identification Advanced Technologies (Auto-ID)*, pages 43–48, 2005.

[12] J. Bringer, H. Chabanne, and T. Icart. Improved Privacy of the Tree-Based Hash Protocols Using Physically Unclonable Function. In R. Ostrovsky, R. D. Prisco, and I. Visconti, editors, *Proceedings of the 6th International Conference on Security and Cryptography for Networks – SCN'08*, volume 5229 of *Lecture Notes in Computer Science*, pages 77–91, Amalfi, Italy, August 2008. Springer.

[13] L. Buttyán, T. Holczer, and I. Vajda. Optimal Key-Trees for Tree-Based Private Authentication. In *Workshop on Privacy Enhancing Technologies - PET 2006*, Cambridge, UK, June 2006.

[14] C. Chatmon, T. van Le, and M. Burmester. Secure anonymous RFID authentication protocols. *Florida State University, Department of Computer Science, Tech. Rep*, 2006.

[15] W. Che, H. Deng, X. Tan, and J. Wang. Scheme of truly random number generator application in rfid tag. *Networked RFID Systems and Lightweight Cryptography Raising Barriers to Product Counterfeiting, Springer*, 2007.

[16] J. H. Cheon, J. Hong, and G. Tsudik. Reducing RFID Reader Load with the Meet-in-the-Middle Strategy. Cryptology ePrint Archive, Report 2009/092, 2009.

[17] 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, December 2007.

[18] H.-Y. Chien and C.-W. Huang. A Lightweight RFID Protocol Using Substring. In *Embedded and Ubiquitous Computing – EUC'07*, volume 4808 of *Lecture Notes in Computer Science*, pages 422–431, Taipei, Taiwan, December 2007. Springer.

[19] W. Choi and B.-h. Roh. Backward Channel Protection Method for RFID Security Schemes Based on Tree-Walking Algorithms. In M. L. Gavrilova, O. Gervasi, V. Kumar, C. J. K. Tan, D. Taniar, A. Lagan, Y. Mun, and H. Choo, editors, *International Conference on Computational Science and its Applications - ICCSA 2006, Proceedings, Part IV*, volume 3983 of *Lecture Notes in Computer Science*, pages 279–287, Glasgow, Scotland, May 2006. Springer-Verlag.

[20] M. Conti, R. Di Pietro, L. V. Mancini, and A. Spognardi. FastRIPP: RFID Privacy Preserving protocol with Forward Secrecy and Fast Resynchronization. In *33th Annual Conference of the IEEE Industrial Electronics Society (IEEE IECON 07)*, pages 52–57, Taipei, Taiwan, November 2007. IEEE, IEEE Computer Society.

[21] M. Conti, R. D. Pietro, L. V. Mancini, and A. Spognardi. RIPP-FS: an RFID Identification, Privacy Preserving Protocol with Forward Secrecy. In *International Workshop on Pervasive Computing and Communication Security – PerSec 2007*, pages 229–234, New York City, New York, USA, March 2007. IEEE, IEEE Computer Society Press.

[22] T. Dimitriou. A Lightweight RFID Protocol to protect against Traceability and Cloning attacks. In *Conference on Security and Privacy for Emerging Areas in Communication Networks – SecureComm*, Athens, Greece, September 2005. IEEE.

[23] M. Feldhofer, S. Dominikus, and J. Wolkerstorfer. Strong Authentication for RFID Systems using the AES Algorithm. In M. Joye and J.-J. Quisquater, editors, *Workshop on Cryptographic Hardware and Embedded Systems – CHES 2004*, volume 3156 of *Lecture Notes in Computer Science*, pages 357–370, Boston, Massachusetts, USA, August 2004. IACR, Springer-Verlag.

[24] M. Feldhofer, S. Dominikus, and J. Wolkerstorfer. Strong authentication for RFID systems using the AES algorithm. In M. Joye and J.-J. Quisquater, editors, *Workshop on Cryptographic Hardware and Embedded Systems – CHES 2004*, volume 3156 of *Lecture Notes in Computer Science*, pages 357–370, Boston, Massachusetts, USA, August 2004. IACR, Springer-Verlag.

[25] M. Feldhofer, J. Wolkerstorfer, and V. Rijmen. AES Implementation on a Grain of Sand. *IEE Proceedings - Information Security*, 152(1):13–20, October 2005.

[26] C. Floerkemeier, R. Schneider, and M. Langheinrich. Scanning with a Purpose-Supporting the Fair Information Principles in RFID Protocols. *2nd International Symposium on Ubiquitous Computing Systems, Tokyo, Japan*, 2004.

[27] S. Garfinkel, A. Juels, and R. Pappu. RFID Privacy: An Overview of Problems and Proposed Solutions. *IEEE SECURITY & PRIVACY*, pages 34–43, 2005.

[28] F. Gosset, F. Standaert, and J. Quisquater. FPGA implementation of SQUASH. In *Proceedings of the 29th Symposium on Information Theory in the Benelux*, 2008.

[29] M. Hellman. A cryptanalytic time-memory trade-off. *IEEE transactions on Information Theory*, 26(4):401–406, 1980.

[30] D. Henrici and P. Müller. Hash-Based Enhancement of Location Privacy for Radio-Frequency Identification Devices Using Varying Identifiers. In R. Sandhu and R. Thomas, editors, *International Workshop on Pervasive Computing and Communication Security – PerSec 2004*, pages 149–153, Orlando, Florida, USA, March 2004. IEEE, IEEE Computer Society.

[31] D. Holcom, W. Burleson, and K. Fu. Initial SRAM state as a Fingerprint and Source of True Random Numbers for RFID Tags. In *Workshop on RFID Security – RFIDSec'07*, Malaga, Spain, July 2007.

[32] X. Huang. Quantifying Information Leakage in RFID Systems. In *10th International Conference on Advanced Communication Technology*. Citeseer, 2008.

[33] S. Jain and S. Das. Collision avoidance in a dense RFID network. In *Proceedings of the 1st international workshop on Wireless network testbeds, experimental evaluation & characterization*, page 56. ACM, 2006.

[34] A. Juels. RFID Security and Privacy: A Research Survey. *IEEE Journal on Selected Areas in Communications*, 24(2):381–394, February 2006.

[35] A. Juels and J. Brainard. Soft Blocking: Flexible Blocker Tags on the Cheap. In S. De Capitani di Vimercati and P. Syverson, editors, *Workshop on Privacy in the Electronic Society – WPES*, pages 1–7, Washington, DC, USA, October 2004. ACM, ACM Press.

[36] A. Juels, R. Rivest, and M. Szydlo. The blocker tag: Selective blocking of RFID tags for consumer privacy. In *Proceedings of the 10th ACM conference on Computer and communications security*, pages 103–111. ACM New York, NY, USA, 2003.

[37] A. Juels, P. Syverson, and D. Bailey. High-power proxies for enhancing RFID privacy and utility. *Proc. of the 5th Workshop on Privacy Enhancing Technologies*, 2005.

[38] A. Juels and S. Weis. Defining Strong Privacy for RFID. In *International Conference on Pervasive Computing and Communications – PerCom 2007*, pages 342–347, New York City, New York, USA, March 2007. IEEE, IEEE Computer Society Press.

[39] G. Khandelwal, K. Lee, A. Yener, and S. Serbetli. ASAP: a MAC protocol for dense and time-constrained RFID systems. *EURASIP Journal on Wireless Communications and Networking*, 2007.

[40] M. Kodialam and T. Nandagopal. Fast and reliable estimation schemes in RFID systems. In *Proceedings of the 12th annual international conference on Mobile computing and networking, MobiCom*. ACM, 2006.

[41] H. Krawczyk. The order of encryption and authentication for protecting communications(or: How secure is SSL?). *Advances in Cryptology–CRYPTO 2001*, pages 310–331, 2001.

[42] T. V. Le, M. Burmester, and B. de Medeiros. Universally Composable and Forward-secure RFID Authentication and Authenticated Key Exchange. In F. Bao and S. Miller, editors, *ACM Symposium on Information, Computer and Communications Security – ASIACCS 2007*, pages 242–252, Singapore, Republic of Singapore, March 2007. ACM, ACM Press.

[43] M. Lehtonen, T. Staake, F. Michahelles, and E. Fleisch. From Identification to Authentication - A Review of RFID Product Authentication Techniques. In *Workshop on RFID Security – RFIDSec'06*, Graz, Austria, July 2006. Ecrypt.

[44] Y. Li and X. Ding. Protecting RFID communications in supply chains. In *Proceedings of the 2nd ACM symposium on Information, computer and communications security*, pages 234–241. ACM New York, NY, USA, 2007.

[45] T. Lim, T. Li, and Y. Li. A Security and Performance Evaluation of Hash-Based RFID Protocols. *4th International Conferences on Information Security and Cryptology-Inscrypt'08*, pages 406–424, 2008.

[46] Z. Liu and D. Peng. True Random Number Generator in RFID Systems Against Traceability. In *IEEE Consumer Communications and Networking Conference – CCNS*, volume 1, pages 620–624, Las Vegas, Nevada, USA, January 2006. IEEE, IEEE.

[47] L. Lu, J. Han, L. Hu, Y. Liu, and L. Ni. Dynamic Key-Updating: Privacy-Preserving Authentication for RFID Systems. *Proceedings of the Fifth IEEE International Conference on Pervasive Computing and Communications*, pages 13–22, 2007.

[48] L. Lu, J. Han, R. Xiao, and Y. Liu. ACTION: Breaking the Privacy Barrier for RFID Systems. *INFOCOM 2009. The 28th IEEE Conference on Computer Communications.*, 2009.

[49] J. Melia-Segui, J. Garcia-Alfaro, and J. Herrera-Joancomarti. Analysis and Improvement of a Pseudorandom Number Generator for EPC Gen2 Tags. In *1st International Workshop on Lightweight Cryptography for Resource-Constrained Devices – WLC'10*, Lecture Notes in Computer Science, Tenerife, Canary Islands, Spain, January 2010. Springer.

[50] A. Menezes, P. Van Oorschot, and S. Vanstone. *Handbook of applied cryptography*. CRC press, 1997.

[51] D. Molnar, A. Soppera, and D. Wagner. A Scalable, Delegatable Pseudonym Protocol Enabling Ownership Transfer of RFID Tags. In B. Preneel and S. Tavares, editors, *Selected Areas in Cryptography – SAC 2005*, volume 3897 of *Lecture Notes in Computer Science*, pages 276–290, Kingston, Canada, August 2005. Springer-Verlag.

[52] D. Molnar and D. Wagner. Privacy and security in library RFID: issues, practices, and architectures. *Proceedings of the 11th ACM conference on Computer and communications security*, pages 210–219, 2004.

[53] J. Myung and W. Lee. An adaptive memoryless tag anti-collision protocol for RFID networks. In *IEEE INFOCOM*, 2005.

[54] K. Nohl and D. Evans. Quantifying Information Leakage in Tree-Based Hash Protocols. In *Conference on Information and Communications Security – ICICS'06*, volume 4307 of *Lecture Notes in Computer Science*, pages 228–237, Raleigh, North Carolina, USA, December 2006. Springer-Verlag.

[55] P. Oechslin. Making a faster cryptanalytic time-memory trade-off. In *Advances in Cryptology–CRYPTO*, volume 3, pages 617–630. Springer, 2003.

[56] M. Ohkubo, K. Suzuki, and S. Kinoshita. Cryptographic Approach to "Privacy-Friendly" Tags. In *RFID Privacy Workshop*, MIT, MA, USA, November 2003.

[57] M. Ohkubo, K. Suzuki, and S. Kinoshita. Efficient Hash-Chain Based RFID Privacy Protection Scheme. In *International Conference on Ubiquitous Computing – Ubicomp, Workshop Privacy: Current Status and Future Directions*, Nottingham, England, September 2004.

[58] R.-I. Paise and S. Vaudenay. Mutual Authentication in RFID: Security and Privacy. In *Proceedings of the 3rd ACM Symposium on Information, Computer and Communications Security – ASIACCS'08*, pages 292–299, Tokyo, Japan, 2008. ACM Press.

[59] P. Peris-Lopez, J. C. Hernandez-Castro, J. Estevez-Tapiador, and A. Ribagorda. LMAP: A real lightweight mutual authentication protocol for low-cost RFID tags. Printed handout of Workshop on RFID Security – RFIDSec 06, July 2006.

[60] P. Peris-Lopez, J. C. Hernandez-Castro, J. Estevez-Tapiador, and A. Ribagorda. M2AP: A minimalist mutual-authentication protocol for low-cost RFID tags. In *International*

Conference on Ubiquitous Intelligence and Computing – UIC06, volume 4159 of *Lecture Notes in Computer Science*, pages 912–923. Springer-Verlag, September 2006.

[61] P. Peris-Lopez, J. C. Hernandez-Castro, J. Estevez-Tapiador, and A. Ribagorda. RFID Systems: A Survey on Security Threats and Proposed Solutions. In *11th IFIP International Conference on Personal Wireless Communications – PWC'06*, volume 4217 of *Lecture Notes in Computer Science*, pages 159–170, Albacete, Spain, September 2006. Springer-Verlag.

[62] 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 *OTM Federated Conferences and Workshop: IS Workshop – IS'06*, volume 4277 of *Lecture Notes in Computer Science*, pages 352–361. Springer-Verlag, November 2006.

[63] 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 *Workshop on Information Security Applications*, Lecture Notes in Computer Science, Jeju Island, Korea, September 2008. Springer-Verlag.

[64] M. Rieback, B. Crispo, and A. Tanenbaum. RFID Guardian: A battery-powered mobile device for RFID privacy management. *Proc. 10th Australasian Conf. on Information Security and Privacy (ACISP 2005*, 3574:184–194, 2004.

[65] S. Sarma, S. Weis, and D. Engels. Radio-Frequency Identification: Security Risks and Challenges. *Cryptobytes, RSA Laboratories*, 6(1):2–9, Spring 2003.

[66] N. Saxena and J. Voris. We Can Remember It for You Wholesale: Implications of Data Remanence on the Use of RAM for True Random Number Generation on RFID Tags. In *Proc. Workshop on Radio Frequency Identification Security (RFIDsec'09), July*, 2009.

[67] Y. Seo and K. Kim. Scalable and Untraceable Authentication Protocol for RFID. In *International Workshop on Security in Ubiquitous Computing Systems – SecUbiq 2006*, Lecture Notes in Computer Science, Seoul, Korea, August 2006. Springer-Verlag.

[68] A. Shamir. Squash: A new one-way hash function with provable security properties for highly constrained devices such as RFID tags. In *International Workshop on RFID Security (RFIDSec'07)*, 2007.

[69] D. Shih, C. Lin, and B. Lin. RFID tags: privacy and security aspects. *International Journal of Mobile Communications*, 3(3):214–230, 2005.

[70] B. Song and C. J. Mitchell. RFID Authentication Protocol for Low-cost Tags. In V. D. Gligor, J. Hubaux, and R. Poovendran, editors, *ACM Conference on Wireless Network Security, WiSec'08*, pages 140–147, Alexandria, Virginia, USA, April 2008. ACM Press.

[71] B. Song and C. J. Mitchell. Scalable RFID Authentication Protocol. In *3rd International Conference on Network & System Security — NSS 2009*, pages 216–224, Gold Coast, Australia, October 2009. IEEE Computer Society.

[72] D. Stinson. *Cryptography: Theory and Practice*. CRC Press, 2002.

[73] G. Tsudik. YA-TRAP: Yet Another Trivial RFID Authentication Protocol. In *International Conference on Pervasive Computing and Communications – PerCom 2006*, Pisa, Italy, March 2006. IEEE, IEEE Computer Society Press.

[74] G. Tsudik. A family of dunces: Trivial RFID identification and authentication protocols. In *Privacy Enhancing Technologies 7th International Symposium, PET 2007, Ottawa, Canada, June 20-22, 2007: Revised Selected Papers*, volume 4776, page 45. Springer, 2007.

[75] J. Voris and N. Saxena. We Can Remember It for You Wholesale: Implications of Data Remanence on the Use of RAM for True Random Number Generation on RFID Tags. In *Workshop on RFID Security – RFIDSec'09*, Leuven, Belgium, July 2009.

[76] W. Wang, Y. Li, L. Hu, and L. Lu. Storage-Awareness: RFID Private Authentication based on Sparse Tree. In *Security, Privacy and Trust in Pervasive and Ubiquitous Computing, 2007. SECPerU 2007. Third International Workshop on*, pages 61–66, 2007.

[77] S. Weis, S. Sarma, R. Rivest, and D. Engels. Security and Privacy Aspects of Low-Cost Radio Frequency Identification Systems. In D. Hutter, G. Müller, W. Stephan, and M. Ullmann, editors, *International Conference on Security in Pervasive Computing – SPC 2003*, volume 2802 of *Lecture Notes in Computer Science*, pages 454–469, Boppard, Germany, March 2003. Springer-Verlag.

[78] S. Weis, S. Sarma, R. Rivest, and D. Engels. Security and privacy aspects of low-cost radio frequency identification systems. In D. Hutter, G. Müller, W. Stephan, and M. Ullmann, editors, *International Conference on Security in Pervasive Computing – SPC 2003*, volume 2802 of *Lecture Notes in Computer Science*, pages 454–469, Boppard, Germany, March 2003. Springer-Verlag.

[79] J. Wu and D. R. Stinson. A Highly Scalable RFID Authentication Protocol. In *Proceedings of the 14th Austrialasian Conference on Information Security and Privacy – ACISP'09*, Brisbane, Australia, July 2009.

[80] Q. Yao, Y. Qi, J. Han, J. Zhao, X. Li, and Y. Liu. Randomizing RFID private authentication. In *IEEE International Conference on Pervasive Computing and Communications-PerCom'09*, pages 1–10. IEEE Computer Society, 2009.

[81] Y. Yousuf and V. Potdar. A Survey of RFID Authentication Protocols. *22nd International Conference on Advanced Information Networking and Applications - Workshops, 2008. AINAW 2008*, pages 1346–1350, March 2008.

[82] D. Zanetti, B. Danev, and S. Čapkun. Physical-layer identification of uhf rfid tags. In *16th ACM Conference on Mobile Computing and Networking – MobiCom'10*. ACM, 2010.