Pers Ubiquit Comput (2012) 16:351–365
DOI 10.1007/s00779-011-0396-y

ORIGINAL ARTICLE

A secure distance-based RFID identification protocol with an off-line back-end database

Pedro Peris-Lopez · Agustin Orfila · Esther Palomar · Julio C. Hernandez-Castro

Received: 15 October 2010 / Accepted: 4 April 2011 / Published online: 3 June 2011

Abstract The design of a secure RFID identification scheme is a thought-provoking challenge, and this paper deals with this problem adopting a groundbreaking approach. The proposed protocol, called Noent, is based on cryptographic puzzles to avoid the indiscriminate disclose of the confidential information stored on tags and on an innovative role reversal distance-bounding protocol to distinguish between honest and rogue readers. The protocol provides moderate privacy protection (data and location) to single tags but its effectiveness increases hugely when it is used to protect a large population of tags (e.g., protection against inventory disclosure). Moreover, in comparison with classical approaches, Noent does not require an on-line database, which facilitates key updating and mitigates desynchronization attacks.

P. Peris-Lopez (✉)
Faculty of Electrical Engineering, Mathematics, and Computer Science (EEMCS), Information Security and Privacy Lab, Delft University of Technology (TU-Delft), P.O. Box 5031, 2600 GA Delft, The Netherlands
e-mail: P.PerisLopez@tudelft.nl

A. Orfila · E. Palomar
Department of Computer Science, Carlos III University of Madrid, Avda. Universidad, 30, 28911 Madrid, Spain
e-mail: adiaz@inf.uc3m.es

E. Palomar
e-mail: epalomar@inf.uc3m.es

J. C. Hernandez-Castro
School of Computing, University of Portsmouth, Buckingham Building 1.17, Lion Terrace, Portsmouth PO1 3HE, UK
e-mail: Julio.Hernandez-Castro@port.ac.uk

Keywords RFID security · WSBC · Cryptographic puzzles · Distance-bounding protocols · Privacy · Traceability

1 Introduction

The market penetration of RFID technology is mainly being delayed because of its cost in comparison with other cheaper and widely used identification technologies such as barcodes and as a result of its associated security risk [23, 41, 52]. One of the main drawbacks of many RFID applications is that tags answer indiscriminately to reader queries, compromising the privacy of tag's holder. To understand this better, let us consider a simple example. Suppose that Bob tags every belonging of his house and a system raises an alarm whether the absence of one of these is detected. Basically, Bob has a real-time inventory of the content of his house, what can be very useful. Nevertheless, such a system has negative implications too. A burglar can scan the content of Bob's house and steal depending on Bob's inventory. Similarly, RFID technology is being used for inventory control, stock security and quality management by manufactures in the food industry, textile industry, etc. In these cases, an adversary can disclose the state of the manufacturer's stock that represents a private and commercially valuable information. As shown in these two illustrative examples, the disclosure of the content of a large population of tags represents a serious threat that puts private information at risk and slows down the deployment of RFID systems.

The need to guarantee the authenticity of the parties involved in an RFID identification process and the critical nature of the information that is at stake are encouraging the use of standard cryptography despite the severe

hardware limitations of low-cost RFID tags [1]. Yet, the level of security strength of a certain protocol does not depend exclusively on the cryptographic primitives used, but sometimes on whether an adversary can successfully (in time and from a given distance) break the system. For instance, in the RFID context if an adversary can run a brute force attack to disclose the static identifier for a population of tags but she consumes an excessive amount of time, the attack becomes impractical. Moreover, the reader should note that the key length used in many RFID applications is shorter than what we can find in standard cryptographic applications [29].

1.1 Contribution

The contribution of this paper is twofold. First, an identification protocol based on a cryptographic proof-of-work is introduced. Secondly, we twist the above protocol combining it with a distance-bounding protocol. Nevertheless, we do not use the classical approach—used in all RFID distance-bounding protocols to the best of our knowledge—in which the reader infers an upper bound of the distance to the tag. On the contrary, in our scheme, the tag deduces the distance to the reader. We emphasize here that it is the first time that this innovative technique is proposed.

1.1.1 Cryptographic puzzles

We introduce the use of cryptographic proof-of-work protocols [26] to discourage misbehavior in RFID systems (i.e., the indiscriminate disclosure of the tags' memory content). In a basic (completely insecure) identification scheme, first the reader sends a $\{Request\}$ message to the tag and then the tag backscatters its static identifier $\{ID\}$ to the reader. As an alternative, we present a method based on a simple concept, as follows:

$$\begin{aligned} &Reader \rightarrow Tag : Request \\ &Tag \rightarrow Reader : Puzzle\,(ID) \end{aligned} \tag{1}$$

The idea is that RFID readers that do not devote the required time and computational effort to solve the puzzle will not access any relevant identification material. Tags will generate puzzles that readers must solve in order to identify tags. After performing this operation, readers will have access to the information, previously encrypted and anonymized, i.e., the tag identifier. However, in this straightforward solution, rogue readers and honest readers would need to make the same effort to solve the cryptographic puzzle.

1.1.2 Distance-bounding protocols

As a main assumption, we theorize that legitimate readers are in close proximity and dishonest readers are often distant. There are many scenarios in which this assumption defend. Nevertheless, in those others in which a rogue reader is as close as it wants to the tags, we do not recommend the usage of our proposal. If we pick up the example of Bob's house again, a honest reader—Bob inside the house and equipped with an RFID reader—is very close to the tags (his belongings) but a dishonest reader—a burglar at the door of the house and equipped with and RFID reader—is far away from the tags attached to Bob's stock.

To the best of our knowledge, in every RFID distance-bounding protocols reported in scientific literature, the verifier (reader) infers an upper bound of the distance the prover (tag) is away from it. We propose a role reversal for the reader and tag, which offers a completely new perspective. In this new scenario, the confidence the tag (verifier) has is a function of its distance to the reader (prover). If we combine the use of distance checking and cryptographic puzzles, the tag can therefore fix the hardness of the puzzle and thus the time/computation associated with its solution depending on distance measures.

The only remaining question is how tags can estimate their distance to readers. A direct approach is to measure the time between challenges and responses in a rapid bit exchange. As low-cost tags do not possess an on-chip clock, a capacitor's discharge time [77] can be enough for a rough estimate of the round trip time (distance). Alternatively, a clock recovery circuit based on a phase-locked loop can be used, as shown in the prototype designed by Bo et al. [7]. Independently of the approach followed, in our proposal a certain degree of inaccuracy (e_i) regarding distance (d_i) does not represent a major security risk. That is, the tag can distinguish between honest readers and rogue readers accurately, which is our main objective, if:

$$\frac{d_2 \pm e_2}{d_1 \pm e_1} \gg 1$$

where d_1 (d_2) is the distance to honest (rogue) readers and e_1 (e_2) is the error in each distance measurement, respectively.

In summary, we propose a secure RFID identification scheme without the need of an online back-end database that is based on a proof-of-work and a distance checking. The goal of the protocol is to preserve the private information of a large number of tags providing the untraceability property as well. An adversary may compromise the privacy of an specific tag—after a huge computational work—but she would fail when a large population of tags is her target. In fact, the proposed scheme is a simple, yet effective, countermeasure against massive inventory disclosure. Moreover, it can be a useful deterrent against counterfeiting, which currently is one of the main concerns for many manufacturers (e.g., clothing or drug).

A preliminary version of our protocol can be found in [66]. In this article, the protocol specification is enhanced and further analyzed in terms of the computational overheads and security properties achieved.

1.2 Organization

The remainder of the article is organized as follows. Section 2 outlines the main proof-of-work-based approaches found in the literature. Moreover, an overview of distance-bounding protocols is presented in Sect. 3. In Sect. 4, we propose a secure RFID Identification protocol based on cryptographic puzzles and the application of a distance-bounding protocol that does not need an online back-end database. Performance and security analysis are presented in Sect. 5. Finally, in Sect. 6, we draw the main conclusions.

2 On proof-of-work mechanisms and cryptographic puzzle systems

2.1 Some definitions and properties

The idea of demonstrating a computational cost performed in a specified interval of time, i.e., the well-known *proof-of-work* (POW) system [26], is still being the basis of a number of recent security protocols. Basically, two entities are involved in such a process, most in the way of a *challenge-response* protocol, in which usually one party (the *verifier*) asks the other (the *prover*) to complete a simple test before granting access or a certain service. Provers cannot obtain the requested material without expending a minimal amount of computational resource and showing the expected evidence. We illustrate a basic interactive scheme, as follows:

$prover \rightarrow verifier$:*Request*
$\hookrightarrow$:Generates or chooses a puzzle.
$verifier \rightarrow prover$:*Puzzle*
$\hookrightarrow$:Commits resources into solving it.
$prover \rightarrow verifier$:*Response*
$\hookrightarrow$:Checks Response.

In the non-interactive POW approach, a number of puzzles are first computed in bulk and then centrally stored together. Provers will select their own challenges or, in other cases, a random start value, as we explain below. This fact means that there is only one round of communication from the prover.

On the other hand, perhaps the most interesting property of recent POW-based approaches is the puzzle's difficulty, i.e., the complexity of the computational cryptographic operation. Several proposals address the challenge of establishing such a cost dynamically and according to different parameters such as the quality of service demonstrated in past interactions.

2.1.1 Trapdoors

Regarding the definition of the computational cost of solving a puzzle, a new concept appears in the mid-1970s, namely the trapdoor [73]. We say F is a trapdoor function if there exists some secret information k, such that given $F(x)$ and k it is easy to compute x and otherwise not. A value is committed that cannot be discovered until the committer reveals either the value (or some other secret) or performs a private computation. Various types of commitment functions have been proposed that maintain a secret for a predictable time delay or until a moderate and predictable amount of computation has occurred.

A bit commitment is a means of requiring an entity to commit to a value, while keeping it hidden until revealing its value at a later point. We use the same example described in [73] to introduce this concept. Alice generates two random-bit strings $\{R_1, R_2\}$ and commits to a message M by computing $h(R_1||R_2||M)$ and sending $\{R_1, h(R_1||R_2||M)\}$ to Bob. When she wants to reveal M to Bob, she sends $\{R_2||M\}$. By the properties of the hash functions: (1) Bob cannot determine M from the first message Alice sent; (2) Alice cannot find a different pair $\{R_2^{'}, M'\}$ such that $h(R_1||R_2||M) = h(R_1||R_2||M')$. Weakly Secret Bit commitment (WSBC) functions work on the same principle, but with the noticeable difference that the secrecy of the bit commitment is breakable after an acceptable predefined limit in terms of time and/or computation. 2nd preimage resistance and weak-preimage resistance are the general properties that a WSBC function $\omega()$ should have. Additionally, collision resistance and near-preimage resistance [47] may be required, depending on the specific application. Interested readers can find an excellent survey on bit commitments in [33].

In addition, we can find several similarities between the common "rational exchange" requirements and the way of fairly delivering trapdoors, indeed. As we further detail, our proposal encounters that challenge in an elegant way.

2.1.2 The first puzzle-based system

The Puzzle System was first conceived in 1974 by Ralph Merkle [51] to ensure two parties can communicate securely over an insecure channel. The two parties will agree on a shared secret by exchanging messages. A *Merkle's puzzle* consists of a bulk of puzzles in the form of an encrypted message with an unknown key. Puzzles use one-way encryption functions whereas the key must be

short enough to allow a brute force attack. Let A and B agree on an encryption function, G, also known by eavesdroppers. G is a common encryption function of two arguments:(1) the message it must encrypt and (2) the encryption key. Figure 1 illustrates the protocol's messages and both parties interactions.

Contrary to the approaches described in sections below, this first cryptographic puzzle found in the literature was not designed as a proof of work, but as an early construction for a public key cryptosystem. Instead, early POW approaches concentrated on finding mechanisms to throttle nodes' selfish behavior by *CPU-bound cost–functions*, which parameterize the amount of work needed to gain some resource. Moreover, different primitives have been applied as a defense against spam and connection depletion, among others. We review the main approaches chronologically in sections below (see Fig. 2 for a schematic summary).

2.2 Early POW approaches

2.2.1 Limiting access

Dwork and Naor in 1992 [26] first formalized the aforementioned ideas to combat junk mailers by requiring senders to demonstrate that they have expended processing time in solving a moderately hard function of the message, called *pricing function*. Authors suggest three candidates

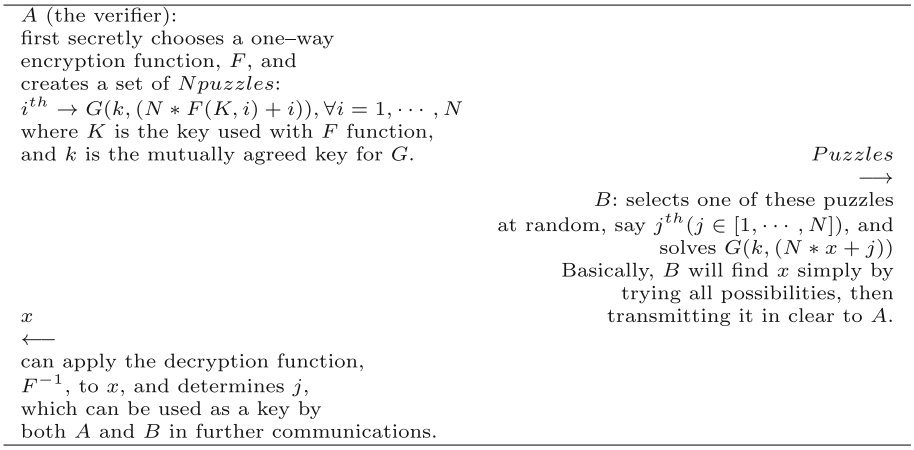


Fig. 1 The Puzzles System first conceived in 1974 by Ralph Merkle [51]

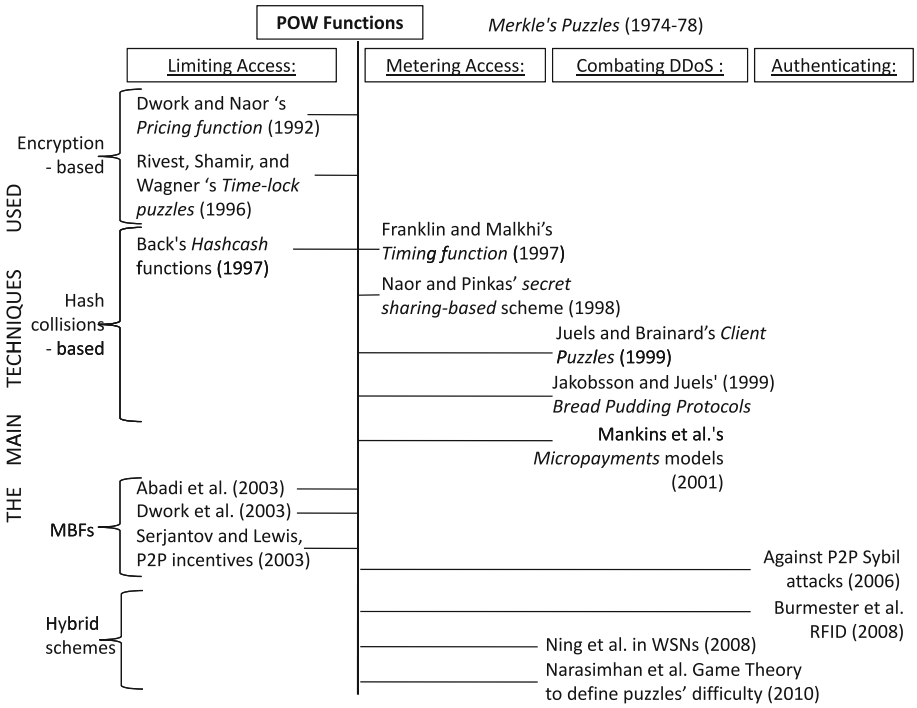


Fig. 2 A chronological classification of the main POW functions according to their objectives and to the cryptographic primitives used

for pricing functions: extracting Square Roots, the Fiat-Shamir-based scheme, and the Ong-Schnorr-Shamir-based scheme. For more details, we refer the reader to the original work. By definition, the function is expensive to solve, while staying comparatively cheap on the verifier side. On the one hand, the global aim is to limit the abilities (resources and time) of adversaries since spammers (even using botnets) cannot compile unlimited amounts of processing time at their disposal. On the other hand, the pricing function should be easy to evaluate given some sort of secret *trapdoor* or *shortcut*. Thus, obtaining a trapdoor function relies on the family of cryptographic functions used.

In 1996, Rivest, Shamir and Wagner [68] thwarted the related problem of an attacker capable of solving puzzles in parallel by means of *time-lock puzzles*. A time-lock puzzle presents a computational problem that cannot be solved without continuously running for a precise amount of time, i.e., encrypting some material (e.g., an encryption key) with the result of repeatedly squaring a value with respect to a composite module. Therefore, these puzzles can be used to implement delays, i.e., by setting the amount of computation, being the amount of delay controlled. Time-lock puzzles use fixed-cost functions, based on superencryption in RSA, and trapdoors.

Furthermore, Back's Hashcash system [5], first announced in 1997, mainly deals with email spam, and denial of service (DoS) attacks as well, by applying a trapdoor-free non-interactive POW system. A cost-function based on finding partial SHA-1 hash collisions assists in filtering email clients. Implementations (developed in several programming languages) modify the message's header by adding a hashcash token, which consists of several recipient-related data such as an address, a timestamp and a random nonce. The main difference between the hashcash cost-function and the pricing function mentioned before is that the former is trapdoor free in order to avoid that senders can cheaply mint tokens to others. However, senders will also have to expend a required processing time to produce a valid header.

2.2.2 Metering access

POWs have also been suggested for metering access. Franklin and Malkhi proposed a secure web metering system, first published in 1997 [30] and later patented in 2000, to monitor Website accesses by involving users in computing the *timing function* for a given input. The timing function is based on one-way hash functions and a unique seed generated for each visit. Later, and due to the importance of web advertising, in 1998 Naor and Pinkas [58] presented an efficient metering system that relies on the polynomial secret sharing scheme of Shamir. Essentially, an audit agency generates a challenge, i.e., a large computational task, and sends it to the server through a secure channel. Each client will be asked to perform a small part of this task, whose final completion proves the visit of k clients using their responses during the time frame. Note that an additional intermediate role, i.e., the auditory entity, modifies the interactive challenge-response protocol and therefore special attention should be paid to communication efficiency and clients collaboration.

2.2.3 Combating DoS

There have been several works that use cryptographic puzzles as an elegant solution to combat denial of service attacks. The Client Puzzle protocol introduced by Juels and Brainard in 1999 [42] uses cryptographic *client puzzles* for preventing a communication protocol such as TCP and SSL from connection depletion by rate-limiting TCP connections. Client puzzles apply a bounded probabilistic cost function. Since the client is expected to search some key space for a known solution, the size of the key space imposes an upper bound on the cost of finding the solution. Authors also use brute-force reversal of a one-way hash function h, as follows. To create a new puzzle, the server generates a random nonce n_S and the difficulty level $k > 0$ of the puzzle (i.e., the k first bits of the hash set as null). Both information is sent to the clients. To solve the puzzle, the client ID_C generates a random nonce n_C and solves $\mathcal{X}$ (i.e., the expected solution) from the following equation by brute force:

$$h(ID_C, n_S, n_C, \mathcal{X}) = \overbrace{000\ldots000}\,\mathcal{Y}$$

where $\mathcal{Y}$ are the rest of the hash bits and represent any bit pattern. The prover is expected to try 2^k possible solutions before finding the right one. Note that n_S must be changed periodically in order to prevent attackers from pre-computing solutions. To this regard, Jakobsson and Juels' *Bread Pudding Protocols*, presented in 1999 [40], extend the Client Puzzle protocol to perform an otherwise useful computation, i.e., re-used the computation waste for another purpose. Their proposal has been recently patented.

2.3 Recent approaches

Now, the idea of using cryptographic POW to increase the cost of sending email and make sending spam unprofitable is being extended to other emerging research areas. It is the turn of new resource-constrained platforms like Wireless Sensor Networks (WSNs) and RFID systems and also peer-to-peer (P2P) systems. Similarly, there are several publications on mitigating DoS and free-riding attacks as well as providing solutions for routing and authentication in such

domains. To deal with free riding, the underlying idea is that sharing can be encouraged by imposing a cost on the downloads, but ensuring that those who share more freely do not incur this cost, and this way dealing with user self-interest. In Mankins et al.'s *micropayment* scheme [50], published in 2001, users are given an incentive to work together toward a common goal through the introduction of these puzzles. Authors present and evaluate several types of micropayment variants.

In the meantime, Abadi et al.'s contribution in 2003 [2] presents an alternative computational POW approach based on memory latency, since memory latency varies much less than that of CPU speeds. The authors called this invention: *Memory-Bound Functions* (MBFs). Before MBFs, the more powerful participants may be able to solve puzzles faster than others. Now, the puzzles are processor or memory-bound computations and can be used to ensure that every node will spend approximately the same amount of critical resource. The analysis and evaluation of several MBFs presented by Dwork et al. in 2003 [27] result in a constant performance across different machines. As a consequence of such validations, many works based on MBFs were presented in different research areas. For example, Serjantov and Lewis [70] consider using client puzzles to provide incentives in P2P systems. Controversy exists, however, about the appropriateness of imposing such an effort to every node in the system, no matter what node's behavior is. This fact is discussed in various papers, such as Laurie and Clayton's publication in 2004 [48], which account the additional burden unfair and counter-productive to honest participants.

On the other hand, Borisov's proposal, presented in 2006 [9], imposes a computational cost on occupying a position within the overlay in order to secure its participation in the P2P network, thus avoiding Sybil attacks where attackers with a large amount of computational resources can get a huge range of node IDs. Furthermore, Ning et al.'s weak authentication mechanism, proposed in 2008 [61], uses one-way key chains to ensure authentication of the broadcast packets in WSNs. Likewise, POWs are recently being adapted to RFID systems by applying lightweight functions. For example, in Burmester et al.'s [13] challenge-response protocol, the RFID tag can obfuscate its identifier, but only the back-end server can disambiguate it using a trapdoor that only it possesses. Authors use public-key one-way functions.

Finally, it is still a challenge to consistently establish what the difficulty of a puzzle to be sent must be. In fact, Narasimhan et al. [59] in 2010 apply Game Theory to formally analyze a Client Puzzle protocol and state that the difficulty of the puzzle should not be determined without a minimal number of computations. Simulations conducted were based on existing puzzles like hash-reversal and time-lock puzzles.

A note on our proposal. Our proposal is designed to control the work on the reader side, bounding the time necessary to reveal the tags' identifiers. The idea is that RFID readers that do not devote the required time and computational effort to solve the puzzle cannot access any relevant identification material. Tags will generate puzzles (a symmetrically encrypted cryptogram and a trapdoor, i.e., some bits of the secret key used in encryption), while readers must solve them in order to identify tags. After this situation takes place, readers will possess the information previously encrypted and anonymized, i.e., the tag identifier. As we describe below, tags establish the hardness of the puzzle, and thus the time/computation associated with its solution, depending on distance measures toward the reader. Moreover, note that the experimental puzzles (initially proved) lie on the exhaustive search on the key space as the fastest method to recover the secret key used in the construction of the symmetric cryptosystem.

3 On distance-bounding protocols

3.1 Preliminaries

Desmedt et al. introduced at Crypto'87 a new attack called the *mafia fraud* [22] that defeats any authentication protocol. This attack allows an adversary to successfully pass the authentication by relaying the messages between a verifier and a legitimate prover and is based on the *chess grandmaster* problem [19]. In 1987, the *mafia fraud* appeared somehow unrealistic because the legitimate prover is required to be involved in the execution of the protocol without being aware of the fraud. However, the great deployment of passive RFID systems turned this attack into a real threat [25, 37]. In a *mafia fraud* scenario, both the reader R and the tag T are honest, but a malicious adversary is performing a man-in-the-middle attack between the reader and the tag by using both a rogue tag $\overline{T}$ and rogue reader $\overline{R}$. The adversary makes $\overline{T}$ interact with the honest reader R and makes the rogue reader $\overline{R}$ interact with the honest tag T. In addition, both rogue parties act as proxies forwarding each other all the messages they receive. As a result, the reader R will authenticate the presence of a rogue token ($\overline{T}$), that is typically further away than T, while R and T remain unaware of the manoeuver. A plot of the attack scenario is shown in Fig. 3. As a practical example of this attack, let us imagine that an adversary wants to open a car fraudulently. He has to place the rogue card near the vehicle while an accomplice with a rogue reader places himself near the owner. The legitimate reader of the car

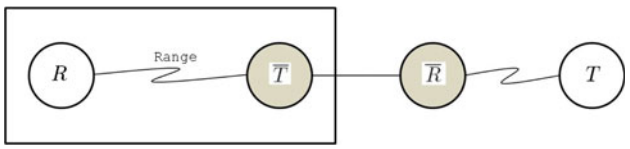


Fig. 3 Mafia fraud attack

powers the rogue card and the rogue reader powers up the owners card and the messages are forwarded. The electronic protection is thus breached remaining both genuine parties unaware. The reason why they remain unaware is that RFID tags are simple devices that automatically answer to any authentication query from a reader without alerting its holder. As a consequence, the reader has no way to decide whether the tag holder agreed to authenticate, being the presence of the tag in the close environment of the reader an implicit authentication agreement from its holder. Due to the maximum reader–tag communication, distance cannot exceed a few decimeters with cryptography-compliant tags, providing the reader with a means to decide whether the distance to the tag is less than a given threshold is of utmost importance.

3.2 Distance-bounding protocols and RFID systems

Distance-bounding protocols (DBP) emerged as a countermeasure against *mafia fraud*. Desmedt et al. introduced the distance-bounding concept [6] based on the measurement of the round trip time (RTT) of exchanged messages. Then, in 1993, Brands and Chaum designed the first distance-bounding protocol [10] based on Desmedt et al. ideas. It was not until 2005 that Hancke and Kunh proposed the first distance-bounding protocol specifically designed for RFID devices [36]. Since then, many proposals have appeared that tries to improve these protocols.

There are three types of attacks related with the distance between the reader and the tag. First, the *distance fraud attack* [10] where a dishonest tag may claim to be closer than it really is. Second, the already exposed *mafia fraud attack* and finally the so-called *terrorist attack* [21] that is also a relay attack but considers an scenario where a dishonest tag colludes with an adversary to trick a reader of its physical proximity. The tag motivation could be either that it is fraudulent or that it is honest but being coerced by the attacker. This attack is depicted in Fig. 4. Here, the behavior of tag T is not legitimate and collaborates with the rogue tag $\overline{T}$. $\overline{T}$ uses T to convince the reader that T is close, when in fact it is not. It is important to note that in a *terrorist attack* is assumed that $\overline{T}$ does not have any advantage for future attacks [32]. Amongst these three types of attacks, *mafia fraud* is the most serious since it can be mounted without the notice of neither the reader nor the tag.

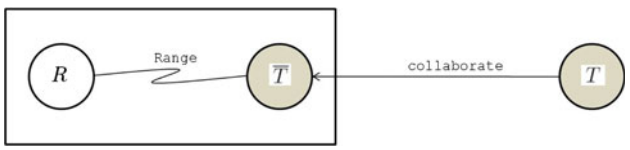


Fig. 4 Terrorist fraud attack

Relay attacks cannot be prevented by cryptographic protocols that operate at the application layer of an RFID protocol stack. At this layer, information about arrival times of messages has already been blurred substantially by the many synchronization, collision avoidance, demodulation, symbol-detection, error-detection and retransmission mechanisms that are implemented in the lower layers [36]. The defense shall be integrated into the physical layer of the communication protocol, so as to obtain high-resolution timing information about the arrival of individual data bits.

3.3 Main approaches

There are several solutions to estimate the distance between two devices. For instance, the use of the global positioning system (GPS), perform multi-channels communication or the received signal strength indication (RSSI) [29, 32]. However, a GPS receiver is too expensive to be added to a low-cost RFID, the physical layer is too simple to allow multi-channels communication and the RSSI can be amplified by the adversary. Accordingly, the solution adopted to design distance-bounding protocols for RFID systems is to measure the RTT in order to estimate an upper bound on the distance between the reader and the tag. This solution only requires a single trusted clock on the reader side and no hardware modification for the tag.

Brands and Chaum's distance-bounding protocol [10] consists of a fast bit exchange phase where the reader sends out one bit (i.e., a challenge) and starts a timer. Then, the tag responds to the reader with one bit (i.e., a response) that stops the timer. The reader uses the RTT to extract the propagation time. This process is repeated n rounds for security reasons and the reader decides whether the tag is within the limitation of the distance. The processing time of the tag must be as short and invariant as possible in order to extract the propagation time accurately.

The are two main approaches for the communication method. First, we can use an ordinary communication channel without any correction mechanism. The main drawback of this approach is the unwanted errors prompted by noise and multi-path effects in the channel. Alternatively, ultra-wide band (UWB) channels seem an interesting approach to combat errors in the channel and to achieve

a resolution of 30 cm or less. Unfortunately, UWB channels have not been practically implemented in the RFID environment. We urge the interested reader to consult Sect. 4 in [35], where a detailed description of the possible communication channels for distance bounding is included.

The popular Hancke and Kuhn's protocol [36] also uses the expensive and complex UWB channel. Reid et al. [67] try to overcome this problem and propose a low-cost protocol that removes this extra link taking advantage of the generally undesirable side channel effects. Later on, Munilla and Peinado [56] proposed another low-cost protocol after pointing out certain problems in the estimation of the timing resolution in Reid et al.'s work [67].

Among the existing distance-bounding protocols based on RTT, two main types can be distinguished depending on the necessity of a final signed message to end the protocol. On the one hand, those that need a signature were introduced by Brands and Chaum [10]. The final signature can be computed on the challenges and the responses only or on some other information like the nonces exchanged. On the other hand, those were based on Hancke and Kuhn protocol [36] in which there is no need of a final signature finishing when the measurement of the RTTs is done. Both protocols can be implemented using symmetric key cryptography. Brands and Chaum's protocol needs three phases: the first and final ones are slow phases while the second one is fast. The RTT is measured n times during the fast phase, while the slow phases include all the time-consuming operations; in particular, the final slow phase is used to complete the authentication. On the contrary, Hancke and Kuhn's protocol consists of a single slow phase followed by a fast one with n RTTs measured. In this case, the fast phase allows the verifier to check both authentication and distance. Most of existing works derive from either Brands and Chaum's protocol [45, 54, 57, 71] or Hancke and Kuhn's protocol [3, 43, 60, 67, 74]. The proposal in [77] is from this perspective an "hybrid" protocol.

3.3.1 Recent variants

Distance-bounding protocols mitigate *mafia fraud attack*, but the adversary always has a certain probability of success. This is the reason why these protocols are compared in terms of this probablity. Munilla and Peinado [54] proposed a protocol in which the success probability of an adversary to accomplish a *mafia fraud* attack is reduced. However, the feasibility of the scheme is questionable since it requires three physical states {0, 1, void}. Singelee and Preneel [71] proposed a distance-bounding protocol, which uses an error correction code to facilitate the corrections of errors (in noisy channels) during the rapid bit exchange. Nevertheless, this scheme's security and implementation cost on RFID tags is questioned in [55]. Finally, the above-mentioned protocols do not address *terrorist fraud* attacks. In 2007, Tu and Piramuthu [77] addressed both *terrorist* and *mafia fraud* attacks and proposed an enhanced scheme. The authors used ideas previously presented in [67] to prevent *terrorist* attacks. Nevertheless, Kim et al. [45] noted that Tu and Piramuthu's protocol is vulnerable to a simple active attack and proposed a new protocol attempting to correct the defenses of all its predecessors.

Kim et al. [45] fulfilled a comparison of the main DBP on several points of view: *mafia* and *terrorist fraud* attack resistance, error resistance, privacy preservation, mutual authentication and computational overhead inside the tag. Regarding the security against *mafia fraud* attack, the false acceptance ratio (M-FAR), i.e., the probability that an adversary succeeds, is computed.

Mitrokotsa et al. [53] proved that the M-FAR of Reid et al. is $(3/4)^n$ and not $(7/8)^n$ as reported in [45]. Accordingly, we have changed this value in Table 1 that reproduces the comparison provided in [45]. As can be seen, the

Table 1 Comparison of distance-bounding protocols [45]

	Mafia	M-FAR	Terrorist	T-FAR	ER	Privacy	MA	CO
BC [10]	Yes	$(1/2)^n$	No	–	No	–	No	2
HK [36]	Yes	$(3/4)^n$	No	–	Yes	–	No	1
R. et al. [67]	Yes	$(3/4)^n$	Yes	$(3/4)^v$	Yes	No	No	2
SP [71]	Yes	$(1/2)^n$	No	–	Yes	–	No	1+ECC
C. et al. [14]	Yes	$(1/2)^n$	No	–	No	–	Yes	4
NV [60]	Yes	$(1/2)^n$	No	–	No	–	No	2k
K. et al. (MA) [45]	Yes	$(1/2)^n$	Yes	$(3/4)^n$	Yes	Yes	Yes	3
K. et al. (no MA) [45]	Yes	$(1/2)^n$	Yes	$(3/4)^n$	Yes	Yes	No	2

ER error resistance, *MA* mutual authentication, *CO* computation overhead

security against *terrorist fraud* attack and its success probability for an adversary (T-FAR) are compared in a similar way. The resilience against channel errors is pretty important for protocol's robustness, as fast bit exchanges are typically sensitive to channel errors. Reid et al.'s protocol discloses identities in cleartext during protocol execution and is thus not privacy preserving. Most of the other protocols assume that the reader knows the identity and secret key of the tag before starting distance-bounding protocol; hence, ignoring the privacy issue or assuming a single secret is shared by all tags. Kim et al.'s protocol allows the reader to learn the tag's identity during execution, although the corresponding overhead is pretty high due to the need of an exhaustive search among all possible keys. Finally, the amount of computation needed in the tag is measured as the required number of computation of pseudo-random functions such as hash functions, symmetric key encryptions, etc.

4 NOENT: a secure RFID identification protocol with an off-line back-end database

In this Section, a new protocol based on the combined use of cryptographic puzzles (proof-of-work) and distance-bounding protocols (distance checking) is introduced. Before introducing the details, the protocol objectives are described—see Sect. 5.2 for a detailed analysis of the security properties. The protocol has to offer privacy protection of the confidential information stored in the tag, that is, the static identifier (*ID*). This identifier allows the unequivocal identification of the tag and can be used as a search index to allocate all the additional information linked to the tag and stored in the off-line back-end database. The anonymization of messages is crucial to avoid traceability and replay attacks. The protocol offers 'moderate' protection concerning privacy and traceability when a single tag is considered. We claim that the security level is 'moderate' because an adversary—located far away from the tag and equipped with a reader—after a highly consuming computation can disclose its secret information, putting at risk its privacy. Nevertheless, the mechanism results very effective when a larger population of tags is considered and in such a case the task of the adversary becomes unaffordable. As a consequence, the protocol results very useful to face interesting real problems such as inventory disclosure or cloning a population of tags (e.g., massive counterfeiting).

4.1 Notation

$\mathcal{R}$ and $\mathcal{T}$ denote the two protocol parties, reader and tag, respectively. As commonly assumed, readers and tags use a non-secure communication channel. That is, the forward (reader-to-tag) and backward (tag-to-reader) channels are insecure. We also assume that the *ID* is the information these two entities would like to exchange securely, where *ID* symbolizes the unique identification number of the tag. Moreover, $enc_k(x)$ is a symmetric key algorithm (e.g., the block cipher AES [28] or TEA [39]) that encrypts message x under key k. The concatenation of variables is denoted by $\|$. Let $\varsigma_j = enc_k(n_1||ID||\alpha_j||n_1||j)$ represent the cryptographic puzzle sent by $\mathcal{T}$ at the j-th protocol instance, where n_1 is a random number and α_j represents the challenge bits in the low-level distance-bounding exchange. The combined use of this nonce and the encryption algorithm facilitates tag identification, providing anonymity and privacy protection, as shown in Sect. 5.

Likewise, $\omega_j^{\pi}(k)$ represents a WSBC function, i.e., a trapdoor (refer to Sect. 2 for further information). Specifically, the WSBC we suggest for the *ID* is simply $\langle \varsigma_j, \omega_j^{\pi}(k) \rangle$. Solutions such as time-lock puzzles, which encrypt k with the result of repeatedly squaring a value with respect to a composite module, may be a very natural implementation for $\omega()$. However, we are forced to choose a much simpler solution due to the severe restrictions of low/moderate-cost tags, which are the most suitable for these kind of solutions [64]. Specifically, in our proposal, the tag randomly selects l bits of k, and this collection of bits forms $\omega_j^{\pi}(k)$. However, for high-cost tags that possess superior computational capabilities, we find more convenient the use of strong solutions such as those outlined previously.

Finally, $h(a\|b)$ is a hash function whose input is the concatenation of a and b. Specifically, $\upsilon_j = h(j \| n_1 \| ID \| \alpha_j \| n_2)$ is the pseudonym sent by the tag at the j-th identification process that mainly has the role of allowing the verification of the puzzle solution after the reader completes the operation. Generally, a pseudonym transmits the static identifier of a tag with the guarantee of keeping confidential information secret and ensuring the untraceability of tag responses [75].

4.2 Noent protocol

In this Section, the Noent protocol is presented. First, an initialization phase is introduced, and then the protocol is described.

4.2.1 Initial preparations

Each tag has an unique identifier number (*ID*) and a secret key (k), which are set in the initialization process. The *ID* and some additional information linked to the tag are stored in an off-line back-end database. Legitimate readers have access to the information stored in this database after a successful authentication.

4.2.2 Identification protocol

In this Section, a secure Identification Protocol between a tag an a reader is introduced. We note here that the back-end database is not involved in this process. So, the reader may run the protocol several times to identify a population of tags. Then, once the static identifiers of these tags are disclosed, the reader may establish a secure connection (i.e., SSL) and check all these values in a single step, avoiding a permanent connection to the back-end database. This approach is not completely new; similar solutions appear in [34, 46].

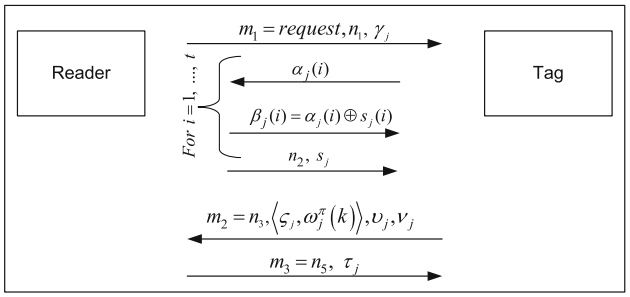
The steps of the protocol for a tag identification are described below (see Fig. 5).

Weak Secret Bit Commitment + Authentication (Steps 1–3–4)

1. $\mathcal{R} \rightarrow \mathcal{T} : m_1 = request, n_1, \gamma_j$
   $\mathcal{R}$ generates two nonces $\{n_1, n_2\}$, a t-bit s_j random value and commits this value by sending random number n_1 and message γ_j ($\gamma_j = h(n_1 \| n_2 \| s_j)$).

Distance-Bounding Protocol (Step 2)

2. $\mathcal{T}$ and $\mathcal{R}$ start a low-level distance-bounding exchange. Due to the practical infeasibility of using UWB channels, we opt for using the normal communication channel without any corrections techniques or packet delimiters—to minimize latency. α_j and s_j symbolize a t-bit random value generated by the tag and the reader, respectively. The bit at the i-*th* position is denoted by subindex (i).
   - The following steps are repeated t times, for $i = \{1, \ldots, t\}$:
     - $\mathcal{T}$ sends bit $\alpha_j(i)$ to $\mathcal{R}$.
     - $\mathcal{R}$ sends bit $\beta_j(i) = \alpha_j(i) \oplus s_j(i)$ to $\mathcal{T}$ immediately after the reception of $\alpha_j(i)$.
   - After completion of the rapid bit exchange, R opens the commitment of the hidden value s_j by sending $\{n_2, s_j\}$.
   - $\mathcal{T}$ can determine an upper bound on the $\{d_{rt}\}$ distance to $\mathcal{R}$ using the maximum of the delay times (RTT) between sending out bit $\{\alpha_j(i)\}$ and receiving bit $\{\beta_j(i)\}$ back. As we have t rounds of exchanging bits without correcting techniques and due to the presence of noise in the channel—real communication channel—an acceptable threshold for the number errors have to be set. This problem was sketchily introduced in [44], and recently, a formal framework was proposed in [24].
3. $\mathcal{T} \rightarrow \mathcal{R} : m_2 = n_3, \langle \varsigma_j, \omega_j^{\pi}(k) \rangle, \upsilon_j, \nu_j$
   $\mathcal{T}$ generates two new nonces $\{n_3, n_4\}$ and computes a WSBC $\langle \varsigma_j, \omega_j^{\pi}(k) \rangle$ which depends on the distance (d_{rt}) that separates the tag and the reader. Specifically, the l variable of $\omega_j^{\pi}(k)$ is conditioned by the distance $\{l = f(d_{rt})\}$. Note that the tag sends its static identifier (ID) in an anonymized and privacy protection way by enclosing this value in the cryptographic puzzle. Finally, message m_2 is ended by an authentication message $v_j = enc_k$ ($j \| n_4 \| ID \| n_1$)).
4. $\mathcal{R} \rightarrow \mathcal{T} : m_3 = n_5, \tau_j$
   $\mathcal{R}$ sends to $\mathcal{T}$ the nonce n_5 and the encrypted message τ_j ($\tau_j = enc_k(j \| n_5 \| ID + 1 \| \alpha_j \| \beta_j \| n_4 \| n_1)$) which has a double purpose: (1) the tag can authenticate the reader and (2) the tag is able to check that the messages (challenges and responses) in the rapid bit exchange have not been altered by an adversary.



1. $\mathcal{R} \rightarrow \mathcal{T}$: $m_1 = request, n_1, \gamma_j$
2. Distance-bounding protocol
   2.0 For $i = 1, \ldots t$
   $\mathcal{T} \rightarrow \mathcal{R}$: $\alpha_j(i)$
   $\mathcal{R} \rightarrow \mathcal{T}$: $\beta_j(i) = \alpha_j(i) \oplus s_j(i)$
   2.1 $\mathcal{R} \rightarrow \mathcal{T}$: n_2, s_j
3. $\mathcal{T} \rightarrow \mathcal{R}$: $m_2 = n_3, \langle \varsigma_j, \omega_j^{\pi}(k) \rangle, \upsilon_j, \nu_j$
4. $\mathcal{R} \rightarrow \mathcal{T}$: $m_3 = n_5, \tau_j$

where $\{n_i\}_{i=0}^{5}$ are different *nonces*
$\gamma_j = h(n_1 || n_2 || s_j)$
$\varsigma_j = enc_k(n_1 || ID || \alpha_j || n_1 || j)$
$\omega_j^{\pi}(k) = \{k_{\pi(0)}, k_{\pi(1)}, \ldots, k_{\pi(l-1)}\}$ is a *l-bit WSBC function*, $\pi()$ is a given permutation and $l = f(d_{rt})$
$\upsilon_j = h(j || n_1 || ID || \alpha_j || n_3)$
and $\nu_j = enc_k(j || n_4 || ID || n_1)$
$\tau_j = enc_k(j || n_5 || ID + 1 || \alpha_j || \beta_j || n_4 || n_1)$

Fig. 5 WSBC + Distance-Bounding Authentication Scheme (Noent approach)

5 Analysis

In this section, we first evaluate the use of cryptographic puzzles. Then, the security properties of Noent scheme are scrutinized.

5.1 Performance analysis

In this subsection, we estimate the computational effort required by readers to solve the cryptographic puzzle enclosed in the WSBCs. As RFID readers are much more powerful than tags in terms of computation and storage capability, we focus on the time consumed in each identification. Optimization of this factor is one of the main objectives for any identification system. A trade-off between security (i.e., inventory protection) and system performance is thus necessary.

For the experiments conducted, we considered two block ciphers as the basis for cryptographic puzzles, AES-128 and TEA [20, 69]. Both were coded in C, compiled with Microsoft Visual C++, and run on an AMD ATHLON(tm)2600 2.09GHz processor, with 1GB RAM under Windows XP SP2. The reader should note that these values are easily extrapolated to the values obtained with commercial RFID readers. First, in many scenarios, RFID readers are directly connected to computers, benefiting of all its processing capabilities. Secondly, in those scenarios in which a handheld RFID readers are used, we should realize that nowadays the computing and memory capabilities of these mobile devices are very similar to PCs (e.g., Motion CV5 at 1.2 GHz and 1 GB RAM [17]).

A factor contributing to complexity is the cost of executing several decryptions, for testing each candidate key. Specifically, the reader receives $\langle \varsigma_j, \omega_j^{\pi}(k) \rangle$, where ς_j and $\omega_j^{\pi}(k)$ represent the cryptographic puzzle and the output of the WSBC function, respectively. The reader then starts an exhaustive search; it probes all possible keys and benefits from the knowledge of l key bits for each. The above process is repeated until the correct key is found.

We have carried out 1000 experiments for different values of (n–l)-bits and also randomly varying the challenges and key used. We consider that more than 32 hidden bits would be impractical. Results are shown in Table 2, for a key length of $n = 128$. For each case, as the l value increases, the number of candidate keys obviously decreases, so the exploration time too. For practical considerations, the

Table 2 Average computational effort made (in 1000 experiments) by each reader for varying amounts of known bits

$n - l$ bits	l bits	Average time required (s)	
		AES-128	TEA
32	96	5495	761
28	100	544	47
24	104	15	0.22
20	108	0.01	0.01

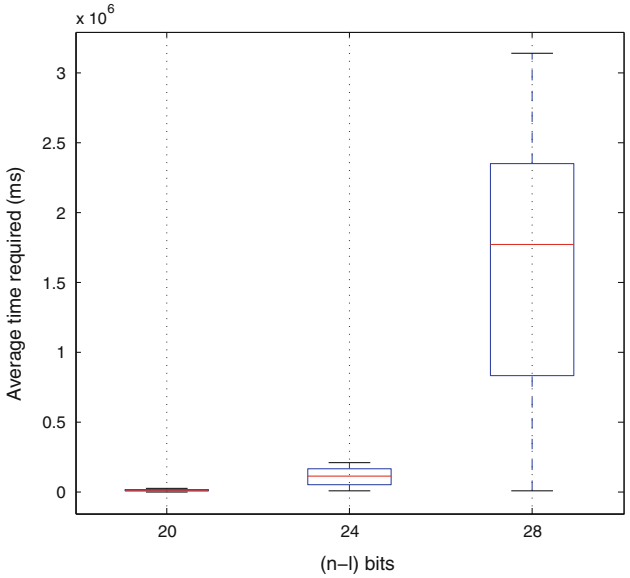


Fig. 6 Average time required (ms) in TEA experiments for different values of ($n - l$)-bits and also randomly varying the challenges and key used

requirements of the application in which the protocol is used will determine the choice of the l value.

Figure 6 provides a characterization of the distribution of our results using 128-key TEA encryption and 20, 24 and 28-hidden bits trapdoors. The purpose of this figure is to show the most extreme values in the experimental data set (maximum and minimum values, which we can consider them as outermost situations or potential outliers), the lower and upper quartiles, and the median. The results are displayed as boxplots, where the box represents the interquartile range, and whiskers extend to the 25th and 75th percentile of the simulations. Around the median, we have an equal number of values above and below, forming the interquartile range upon which we can consider an associated probability of occurrence.

Finally, we review the mandatory hardware demands for implementation of the proposed scheme. An encryption function is employed for the generation of the cryptographic puzzle and the authentication tokens. For that, we find symmetric cryptography convenient rather than asymmetric cryptography, which may be appropriate for high-cost RFID tags (e.g., e-passports [4, 62]). To facilitate verification of the correct solution to the puzzle, an anonymized version of the tag's identifier is used. Specifically, for pseudonym generation, we opt for the use of a hash function, one of the most common solutions in the literature [15, 16, 18, 49, 72, 75, 76]. The commitment scheme is also based on the use of a hash function. Generation of random numbers is necessary to avoid traceability and replay attacks. Additionally, random numbers are also used for selection of the l bits that constitute WSBC function

Table 3 Practical considerations about implementation

	Low-cost RFID tags	Moderate-cost RFID tags
Encryption (cryptographic puzzle + authentication tokens)	TEA [39]	AES-128 [28]
Hash function (pseudonym + commitment scheme)	H-PRESENT-128 [8]	SHA-1 [63]
PRNG (anonymity + WSBC function)	LAMED [65]	Grain [38]
Total GE	4-7K GE	8-12K GE

output. As many different primitives can be selected for these purposes, we suggest various options in Table 3. We make a rough distinction between low-cost RFID tags and moderate-cost RFID tags [64]. To clarify this distinction, we include the approximate number of Gates Equivalents (GE) for each of these alternatives at the bottom of the table.

5.2 Security analysis

The two main objectives of our protocol are privacy protection and untraceability. Regarding privacy, the static identifier of the tag is never sent in clear on the channel. Specifically, an encrypted version $\varsigma_j = enc_k(n_1||ID||\alpha_j||n_1||j)$ of the *ID* that requires the knowledge of the secret key k for its computation is used for puzzle generation. The puzzle is accompanied by a pseudonym $\upsilon_j = h(j||n_1||ID||\alpha_j||n_3)$ of the tag's *ID*, which is used for puzzle verification without compromising any confidential information. Additionally, part of the secret key $\omega_j^{\pi}(k)$ is delegated from $\mathcal{T}$ to $\mathcal{R}$. An attacker cannot exploit this as the number of bits delegated is conditioned by the distance that separate the tag and the reader. Rogue readers are often far way from tags or its presence would be detected easily by modern detection techniques [12] or simple visual inspection. Furthermore, different l bits of the key are randomly selected and employed in each iteration. Specifically, upon determining $l = f(d_{rt})$, the tag randomly picks up one of the WSBCs' possible $C(n, l)$ combinations, where n and l are the bit lengths of the key and the WSBC function, respectively. Where the $C(n, l)$ value may be considered poor in terms of security (e.g., $<2^{32}$ for low-cost RFID tags and $<2^{64}$ for moderate-cost tags [64]), we recommend updating of the key. To offer traceability protection, the freshness of the exchanged messages is provided by the nonces generated by the reader and the tag and used in each sub-message generation $\{\gamma_j, \varsigma_j, \upsilon_j, \nu_j, \tau_j\}$. An attacker cannot distinguish between the answers from different tags, thus guaranteeing users' location privacy.

Confidential information and location information are delegated to readers once a cryptographic puzzle is solved. After that, the reader can acquire the private information linked to the tag (i.e., $\{ID, k\}$). So, the responses of this tag can be uniquely identified using the captured information, compromising the tag holder's privacy location. Does it represent a data privacy invasion? Note that it is commonly assumed that rogue readers are far way from tags and legitimate readers are close. So, first of all there is a significant difference in the effort made—consumed time—by an honest reader and a dishonest reader. Secondly, the main application of our protocol is protection against the revelation of the contents of a great number of tags (e.g., a clothing manufacturer's inventory or the stock of books in a library). So, a rogue reader can compromise the privacy of a specific tag after consuming a significant amount of time. However, this task results unfeasible—the consumed time is excessive—when it deals with a group of tags. If private information is not compromised, tracking this group of tags is in vain as the rogue reader cannot distinguish between responses made by different tags.

Regarding performance, it is questionable if the proposed protocol is efficient enough. If we compare Noent with a simple and non-secure RFID identification scheme in which tags backscatter its static identifier indiscriminately, it is not. Nevertheless, if we compare the consuming time of solving a puzzle—identify a tag in our scheme—with the slow reading rate of barcode technology, Noent is very effective, efficient and reliable. For instance, using Noent, we can check the stock of a warehouse in few hours and this task would take several days when barcode labels are used.

One of the most important advantages of the proposed scheme is the absence of an on-line back-end database. The reader may connect to the database at regular or arbitrary time intervals and check a collection of tags identifers, all at once. Moreover, taking advantage that the tag's secret key is neither stored in database nor shared with the reader(s), the key of the tag can be updated (i.e., $k^n = h(k^{n-1})$) each time a tag is read. More precisely, the reader discloses the *ID*, and the secret key K of the tag by solving a cryptographic puzzle and the tag is authenticated by checking ν_j. The mutual authentication is completed when the tag validates τ_j. There are two alternatives for the precise moment when the tag updating is executed. Basically, the tag can update its key after sending m_2 or this updating may depend on a successful reader authentication. The second option is appropriate when we need a permanent connection to the database. Nevertheless, we opt for the first alternative because it prevents to a greater extent from the possibilities of a successful attack by an adversary listening on the channel. In virtue of this updating, the scheme provides backward security [31]. That is, past communications are protected even when the content of the tag is revealed.

Another important aspect regarding the usage of RFID tags is resilience to cloning attacks. The proposed scheme can be viewed as a countermeasure against them. An attacker can clone a particular tag after solving the cryptographic puzzle sent by it. However, the success ratio of this attack is zero when the number of tags is increased because of the excessive time consumed in solving all the associated puzzles. It may seem to readers of this paper that honest readers would suffer from the same problem. Nevertheless, our proposal is based on the idea that the hardness of the puzzle depends on the distance between a tag and a reader. As honest readers are close to tags, they would receive much simpler puzzles than those received by rogue readers that are far away from tags. So, honest readers are able to identify a voluminous population of tags. The only existing possibility is the placing of the rogue reader close to tags but its presence would be easily detected by using modern detection techniques such as the rogue RFID detector proposed in [12] or classical visual inspection.

Regarding protection against relay attacks, the proposed scheme is inspired on the most recent results in this research area. *Distance fraud* attack is possible when there is no relationship between the challenge bits and the response bits exchanged during the distance verification. In our scheme, the use of commitments prevents dishonest readers from sending their answer before reception of the challenges and response values (i.e., $\beta_j(i) = \alpha_j(i) \oplus s_j(i)$) depend on the challenge bits (i.e., $\alpha_j(i)$). Accordingly, the success probability for an adversary to conduct a *distance fraud* attack is at most $(\frac{1}{2})^t$ [11], where t is the number of bits exchanged in the rapid bit phase. To avoid *mafia fraud* attacks, two further precautions are taken. First, a fast bit exchange between the tag and the reader is performed. Second, once the distance-bounding exchange is completed, the reader sends to the tag the encrypted message τ_j, which includes all the random values $\{\alpha_j, \beta_j\}$ passed in the fast phase by the two entities. If the encryption scheme is secure and the reader is not in physical proximity to the tag, an adversary has a success probability upper bounded by the ideal $(\frac{1}{2})^t$ [11, 44]. Finally, we emphasize here that *terrorist attacks* were discarded by design because in the protocol assumptions, dishonest/terrorist readers are considered to be far way from tags. So, the collaboration between a dishonest reader and a terrorist reader would result in vain.

6 Conclusions

In this paper, a cutting-edge scheme, which tackles the design of a secure RFID identification protocol, is introduced. The proposal results innovative due to the approach followed. To the best of our knowledge, it is the first time to study the use of cryptographic puzzles and a role reversal distance-bounding protocol (tag(verifier); reader (prover)) is proposed in the RFID context. Specifically, Noent protocol provides privacy and offers protection against traceability attacks, which are two of the main security risks linked with RFID technology. In addition, it results a very effective protection technique when the privacy—information and location—of a large population of tags is at stake. Finally, Noent does not need an on-line back-end database what makes it very advantageous and facilitates its use in real scenarios.

Acknowledgments Our sincere thanks to the anonymous reviewers for their useful comments which assisted us in the improvement of this article.

References

1. O'Connor MC (2009) Bridge researchers demo highly secure EPC Gen-2 RFID. RFID J. http://www.rfidjournal.com/article/view/5074
2. Abadi M, Burrows M, Manasse M, Wobber T (2003) Moderately hard, memory-bound functions. In: Proceedings of the 10th annual network and distributed system security symposium, pp 25–39
3. Avoine G, Tchamkerten A (2009) An efficient distance bounding RFID authentication protocol : balancing false acceptance rate and memory requirement. In: Proceedings of the information security conference (ISC'09), pp 250–261
4. Avoine G, Kalach K, Quisquater JJ (2008) ePassport: securing international contacts with contactless chips. In: Financial cryptography and data security – FC'08, vol 5143, Springer, Berlin, Lecture Notes in Computer Science, pp 141–155
5. Back A (2002) Hashcash. A denial of service counter-measure. Technical report. http://www.hashcash.org/hashcash.pdf
6. Bengio S, Grassard G, Desmedt YG, Goutier C, Quisquater JJ (1991) Secure implementation of identification systems. J Cryptol 4:175–183
7. Bo F, Yujie D, Xiaoxing Z, Yingjie L (2009) Low power clock recovery circuit for passive hf rfid tag. Analog Int Circuits Signal Process 59:207–214
8. Bogdanov A, Leander G, Paar C, Poschmann A, Robshaw MJ, Seurin Y (2008) Hash functions and RFID tags : mind the gap. In: Proceedings of the 10th international workshop cryptographic hardware and embedded systems, vol 5154, Springer, Berlin, Lecture Notes in Computer Science
9. Borisov N (2006) Computational puzzles as sybil defenses. In: Proceedings of the 6th international conference on Peer-to-Peer computing, IEEE, pp 171–176
10. Brands S, Chaum D (1993) Distance-bounding Protocols. In: Advances in cryptology (EUROCYPT'93). Springer, New York, pp 344–359
11. Brands S, Chaum D (1994) Distance-bounding protocols. In: Proceedings of EUROCRYPT '93: workshop on the theory

and application of cryptographic techniques on advances in cryptology
12. Broer DA (2010) Rogue RFID detector. United States, Patent Number 20100148964. http://www.freepatentsonline.com/y2010/0148964.html
13. Burmester M, de Medeiros B, Motta R (2008) Robust, anonymous rfid authentication with constant key-lookup. In: Proceedings of the 2008 ACM symposium on Information, computer and communications security, ACM, pp 283–291
14. Capkun S, Buttyan L, Hubaux JP (2003) SECTOR: Secure tracking of node encounters in multi-hop wireless networks. In: Proceedings. of the 1st ACM worskhop on security of Ad Hoc and sensor networks, p 12
15. Chang JC, Wu HL (2009) A hybrid RFID protocol against tracking attacks. In: International conference on intelligent information hiding and multimedia signal processing, vol 0, IEEE Computer Society, Los Alamitos, pp 865–868
16. Choi EY, Lee SM, Lee DH (2005) Efficient RFID authentication protocol for ubiquitous computing environment. In: International workshop on security in ubiquitous computing systems – SecUbiq 2005, vol 3823 Springer, Lecture Notes in Computer Science, pp 945–954
17. Computing M (Consulted on February, 2011) Motion C5v Tablet PC. http://www.motioncomputing.com/products/tablet_pc_c5.asp
18. Conti M, Pietro RD, Mancini LV, Spognardi A (2007) RIPP-FS: an RFID identification, privacy preserving protocol with forward secrecy. In: International workshop on pervasive computing and communication security – PerSec 2007, IEEE Computer Society Press, pp 229–234
19. Conway JH (2000) On numbers and games, 2nd edn. AK Peters, Ltd
20. Daemen J, Rijmen V (2002) The design of Rijndael: AES the advanced encryption standard. Springer, Berlin
21. Desmedt Y (1988) Major security problems with the 'unforgeable' (Feige)-Fiat-Shamir proofs of identity and how to overcome them. In: Proceedings of SecuriCom 88, pp 7–15
22. Desmedt Y, Goutier C, Bengio S (1987) Special uses and abuses of the Fiat-Shamir passport protocol. In: Proceedings of advances in cryptology—CRYPTO '87, pp 21–39
23. Deursen Tv, Radomirovic S (2008) Attacks on RFID protocols. Cryptology ePrint Archive, Report 2008/310, http://eprint.iacr.org/
24. Dimitrakakis C, Mitrokotsa A (2010) Expected loss analysis of thresholded authentication protocols in noisy conditions. CoRR abs/1009.0278
25. Drimer S, Murdoch SJ (2007) Keep your enemies close: distance bounding against smartcard relay attacks. In: Proceedings of 16th USENIX security symposium, USENIX Association, Berkeley, CA, USA, pp 1–16
26. Dwork C, Naor M (1992) Pricing via processing or combatting junk mail. In: Proceedings of the 12th annual international cryptology conference on advances in cryptology, Springer, Berlin, pp 139–147
27. Dwork C, Goldberg A, Naor M (2003) On memory-bound functions for fighting spam. In: Proceedings of advances in cryptology, Springer, Berlin, pp 426–444
28. Feldhofer M, Wolkerstorfer J, Rijmen V (2005) AES implementation on a grain of sand. IEE Proc- Inform Secur 152(1):13–20
29. Fishkin K, Roy S (2003) Enhancing RFID privacy via antenna energy analysis. Technical Report IRS-TR-03-012, ECRYPT
30. Franklin M, Malkhi D (1997) Auditable metering with lightweight security. In: Proceedings of financial cryptography, vol 1318, Springer, Berlin, pp 151–160
31. Garcia FD, van Rossum P (2009) Modeling privacy for Off-line RFID systems. In: Workshop on RFID security – RFIDSec'09
32. Gildas A, Bingöl MA, Kardaş S, Lauradoux C, Martin B (2010) A framework for analyzing RFID distance bounding protocols. J Comput Secur 19(2):289–317 (Special Issue on RFID System Security)
33. Goldschlag D, Stubblebine S, Syverson P (2010) Temporarily hidden bit commitment and lottery applications. Int J Inform Secur 9:33–50
34. Han TS, Dillon S, Chang E (2007) Anonymous mutual authentication protocol for RFID tag without back-end database. In: Proceedings of the 3rd international conference on Mobile ad-hoc and sensor networks, MSN'07, pp 623–632
35. Hancke G (2010) Design of a secure distance-bounding channel for RFID. J Network Comput Appl 34(3):877–887
36. Hancke G, Kuhn M (2005) An RFID distance bounding protocol. In: Proceedings of the first international conference on security and privacy for emerging areas in communications networks (SECURECOMM'05), IEEE, pp 67–73
37. Hancke G, Mayes K, Markantonakis K (2009) Confidence in smart token proximity: relay attacks revisited. Comput Secur 28(7):615–627
38. Hell M, Johansson T, Maximov A, Meier W (2006) A stream cipher proposal: grain-128, pp 1614–1618. doi:10.1109/ISIT.2006.261549
39. Israsena P (2006) Securing Ubiquitous and Low-cost RFID Using Tiny Encryption Algorithm. In: Proceedings of international symposium on wireless pervasive computing
40. Jakobsson M, Juels A (1999) Proofs of work and bread pudding protocols. In: Proceedings of the IFIP TC6/TC11 joint working conference on secure information networks, Kluwer, Dordrecht, pp 258–272
41. Juels A (2006) Rfid security and privacy: a research survey. IEEE J Selected Areas in Commun 24(2):381–394
42. Juels A, Brainard J (1999) Client puzzles: a cryptographic defense against connection depletion attacks. In: Proceedings of the networks and distributed security systems, California, pp 151–165
43. Kim CH, Avoine G (2009) RFID distance bounding protocol with mixed challenges to prevent relay attackss. In: Proceedings of the conference on cryptology and network security (CANS'09), pp 119–113
44. Kim CH, Avoine G, Koeune F, Standaert FX, Pereira O (2008a) The Swiss-Knife RFID distance bounding protocol. In: Proceedings of international conference on information security and cryptology – ICISC, Springer, Berlin, LNCS
45. Kim CH, Avoine G, Pereira O (2008b) The Swiss-Knife RFID Distance Bounding Protocol. In: Proceedings of the conference on information security and cryptology (ICISC 2008), pp 98–115
46. Kim S, Lee K, Kim S, Won D (2009) Security analysis on anonymous mutual authentication protocol for RFID tag without back-end database and its improvement. World Acad Sci Eng Technol 460–464
47. Knudsen L, Muller F (2005) Some attacks against a double length hash proposal. In: ASIACRYPT, vol 3788, Springer, Berlin, Lecture Notes in Computer Science, pp 462–473
48. Laurie B, Clayton R (2004) Proof-of-work proves not to work. In: Proceedings of the 3rd workshop on the economics of information security
49. Lee S, Asano T, Kim K (2006) RFID mutual authentication scheme based on synchronized secret information. In Symposium on cryptography and information security
50. Mankins D, Krishnan R, Boyd C, Zao J, Frentz M (2001) Mitigating distributed denial of service attacks with dynamic resource pricing. In: Proceeidngs of the 17th annual conference on computer security applications
51. Merkle RC (1978) Secure communications over insecure channels. Commun ACM 21(4):294–299

52. Mitrokotsa A, Rieback MR, Tanenbaum AS (2008) Classification of RFID attacks. In: Proceedings of the 2nd international workshop on RFID technology
53. Mitrokotsa A, Dimitrakakis C, Peris-Lopez P, Hernandez-Castro JC (2010) Reid et al.'s distance bounding protocol and mafia fraud attacks over noisy channels. IEEE Commun Lett 14(2): 121–123
54. Munilla J, Peinado A (2008) Distance bounding protocols for RFID enhanced by using void-challenges and analysis in noisy channels. Wirel Commun Mob Comput J 8(9):1227–1232
55. Munilla J, Peinado A (2010) Attacks on a distance bounding protocol. Comput Commun 33(7):884–889
56. Munilla J, Peinado A (2010) Enhanced low-cost RFID protocol to detect relay attacks. Wirel Commun Mob Comput J 10(3):361–371
57. Munilla J, Ortiz A, Peinado A (2006) Distance bounding protocols with void-challenges for RFID. In: Workshop on RFID security, RFIDSec2006
58. Naor M, Pinkas B (1998) Secure and efficient metering. In: Proceedings of advances in cryptology (EUROCRYPT'98), vol 1403, Springer, Berlin, pp 576–590
59. Narasimhan H, Varadarajan V, Rangan C (2010) Game theoretic resistance to denial of service attacks using hidden difficulty puzzles. In: Proceedings of information security, practice and experience, vol 6047, Springer, Berlin, Lecture Notes in Computer Science, pp 359–376
60. Nikov V, Vauclair M (2008) Yet Another Secure Distance-Bounding Protocol. In: Proceeidngs of the conference on security and cryptography (SECRYPT 2008), pp 218–221
61. Ning P, Liu A, Du W (2008) Mitigating dos attacks against broadcast authentication in wireless sensor networks. ACM Trans Sensor Networks 4(1):1–35
62. Nithyanand R (2009) The evolution of cryptographic protocols in electronic passports. Cryptology ePrint archive, Report 2009/200, http://eprint.iacr.org/
63. O'Neill (McLoone) M (2008) Low-Cost SHA-1 Hash Function Architecture for RFID Tags. In: Proceedings of conference on RFID security
64. Peris-Lopez P (2008) Lightweight cryptography in radio frequency identification (RFID) systems. PhD thesis, Computer Science Department, Carlos III University of Madrid
65. Peris-Lopez P, Hernandez-Castro JC, Estevez-Tapiador JM, Ribagorda A (2009) LAMED – A PRNG for EPC class-1 generation-2 RFID specification. Comput Stand Interfaces 31(1):88–97. doi:10.1016/j.csi.2007.11.013
66. Peris-Lopez P, Hernandez-Castro JC, Tapiador JME, Palomar E, van der Lubbe JC (2010) Cryptographic puzzles and distance-bounding protocols: practical tools for RFID security. In: In IEEE international conference on RFID, IEEE, IEEE Computer Society, pp 45–52
67. Reid J, Gonzalez Nieto J, Tang T, Senadji B (2007) Detecting relay attacks with timing based protocols. In: Proceedings of 2nd ACM symposium on information, computer, and communications security, pp 204–213
68. Rivest RL, Shamir A, Wagner DA (1996) Technical report mit/lcs/tr-684. time-lock puzzles and timed-release crypto. Technical report
69. Russell M (2004) Tinyness: an overview of tea and related ciphers. http://www.users.cs.york.ac.uk/~matthew/TEA/TEA.html
70. Serjantov A, Lewis S (2003) Puzzles in p2p systems. In: Proceedings of the 8th CaberNet Radicals Workshop
71. Singelee D, Preneel B (2007) Distance bounding in noisy environments. In: Proceedings of the European conference on security and privacy in Ad-Hoc and sensor networks (ESAS), vol 3, pp 101–115
72. Song B, Mitchell CJ (2008) RFID Authentication protocol for Low-cost Tags. In: ACM conference on wireless network security, WiSec'08, ACM Press, pp 140–147
73. Syverson P (1998) Weakly secret bit commitment: applications to lotteries and fair exchange. In: Proceedings of the 11th IEEE computer security foundations workshop, pp 2–13
74. Trujillo-Rasua R, Martin B, Avoine G (2010) The poulidor distance-bounding protocol. In: Workshop on RFID security, RFIDSec2010
75. Weis S, Sarma S, Rivest R, Engels D (2003) Security and privacy aspects of low-cost radio frequency identification systems. In: International conference on security in pervasive computing – SPC 2003, vol 2802, Springer, Berlin, Lecture Notes in Computer Science, pp 454–469
76. Yang J, Park J, Lee H, Ren K, Kim K (2005) Mutual authentication protocol for low-cost RFID. Handout of the Ecrypt Workshop on RFID and Lightweight Crypto
77. Yu-ju T, Piramuthu S (2007) RFID Distance bounding protocols. In: Proceedings of the first international EURASIP workshop on RFID technology