Expert Systems

Article

DOI: 10.1111/j.1468-0394.2012.00652.x

A genetic tango attack against the David–Prasad RFID ultra-lightweight authentication protocol

David F. Barrero,[1] Julio César Hernández-Castro,[2]
Pedro Peris-Lopez,[3] David Camacho[4] and María D. R-Moreno[1]

(1) Departamento de Automática, Universidad de Alcalá, Spain
Email: david@aut.uah.es; mdolores@aut.uah.es
(2) School of Computing, University of Portsmouth, UK
Email: Julio.Hernandez-Castro@port.ac.uk
(3) Faculty of EEMCS, Security Lab, Delft University of Technology, The Netherlands
Email: P.PerisLopez@tudelft.nl
(4) Universidad Autónoma de Madrid, Escuela Politécnica Superior, Spain
Email: david.camacho@uam.es

Abstract: *Radio frequency identification (RFID) is a powerful technology that enables wireless information storage and control in an economical way. These properties have generated a wide range of applications in different areas. Due to economic and technological constrains, RFID devices are seriously limited, having small or even tiny computational capabilities. This issue is particularly challenging from the security point of view. Security protocols in RFID environments have to deal with strong computational limitations, and classical protocols cannot be used in this context. There have been several attempts to overcome these limitations in the form of new lightweight security protocols designed to be used in very constrained (sometimes called ultra-lightweight) RFID environments. One of these proposals is the David–Prasad ultra-lightweight authentication protocol. This protocol was successfully attacked using a cryptanalysis technique named Tango attack. The capacity of the attack depends on a set of boolean approximations. In this paper, we present an enhanced version of the Tango attack, named Genetic Tango attack, that uses Genetic Programming to design those approximations, easing the generation of automatic cryptanalysis and improving its power compared to a manually designed attack. Experimental results are given to illustrate the effectiveness of this new attack.*

Keywords: RFID, ultra-lightweight cryptography, genetic programming, security, tango attack

1. Introduction

For the last 40 years, *barcode technology* has played an almost universal role in object identification. Its simplicity, low cost and robustness has made it an universal technology in certain sectors such as supermarket checkout or logistics. However, after all these decades of barcode hegemony, two technologies are changing its position: QR codes and *radio frequency identification*, or simply RFID (Brown, 2007) tags.

RFID is a technology that enables remote identification of objects, much the same as barcodes. However, it has some features that notably increase the range of applications in comparison to barcodes or QR codes. The use of radio avoids the need of visual contact to access the information, making the identification process much more reliable and faster even from a longer operational distance. This small, but important, difference gives RFID an impressive range of new applications. It can be used, of course, in supermarket checkout and logistics (Oh & Park, 2008), but also access control (Chien, 2006), military (Landt, 2005), people orientation in buildings (R-Moreno et al., 2011), counterfeit detection and much more.

The ubiquity of RFID technology, as well as its wireless nature, the sensitivity of some applications such as the use of RFID in passports, and the constrains imposed by the hardware, generates a set of new stimulating problems that are still wide open. One of the main problems of RFID technology is security. Since the communication is done through radio frequency, it is relativity easy to access the device, or eavesdrop the communication. It is the same problem found, for instance, in Wi-Fi. Nonetheless, there is a critical difference: the computational resources available for RFID devices are very scarce, in some cases extremely scarce. Consequently, it is not possible to apply the solutions found in classical cryptography because they require computational resources that simply are not available in RFID, then new methods must be developed.

RFID security is a hot research area. In particular, the design and implementation of strong authentication protocols that overcome the intrinsic limitations of RFID devices is an important field with strong implications in practical issues. Actually, a new branch of Cryptography, named Lightweight Cryptography (Cole & Ranasinghe, 2008), has recently emerged. In this sense, intense research is being done, and several authors have proposed many authentication protocols.

One of these protocols was proposed by David and Prasad (David & Prasad, 2010) claiming it achieved strong authentication. However, this claim was questioned by (Hernández-Castro et al., 2010). The authors successfully performed an analytical attack, but they also introduced a new passive cryptanalysis method called Tango attack. In this paper, we

propose a major improvement over the Tango attack called Genetic Tango attack. Our proposal is based on the use of Genetic Programming (GP) to automatically seek better approximations that play a key concept in the original Tango attack. Additionally, we provide empirical data showing that GP is able to find much better approximations than a team of human experts. These evolved approximations lead to more powerful attacks against the David–Prasad authentication protocol than those presented in the original Tango attack.

The rest of the paper is organized as follows. We first introduce the related work. After that, a description of the David–Prasad authentication protocol is presented, followed in section 4 by a description of the Tango attack. Then, section 5 shows the Genetic Tango attack and presents experimental data of a Genetic Tango attack against the David–Prasad authentication protocol. Finally, the paper ends with some conclusions and future work.

2. Related work

Lightweight cryptography deals with the problem of providing security with a minimal use of computational resources. Actually, it uses very simple operations such as boolean operators and rotations, aiming to require a minimal footprint. Fortunately, the amount of information that has to be transmitted is limited, often we are only interested in identifying the tag, and thus the problem can be stated as an authentication problem. We should take into account that the reader has to authenticate the tag, but sometimes the tag also has to authenticate the reader to avoid unauthorized accesses (mutual authentication). Unlike tags, computational resources available for the reader are not as limited as in the tags. So the communication between the reader and the database can be done using classical cryptographic techniques. Subsequently, we can assume that this communication is secure. The challenge for lightweight cryptography is the reader-to-tag (forward) and the tag-to-reader (backward) channels.

In the last years, several authors have proposed different solutions. In 2006, the UMAP family of protocols [e.g. LMAP (Peris-López et al., 2006b) and M^{2}AP (Peris-López et al., 2006a)] was introduced and attracted certain attention of the research community. After some rounds of cryptanalysis of these schemes, many (if not all) of its security objectives were circumvented, for example with active attacks (Chien & Huang, 2007; Li & Wang, 2007) and later with passive (Báráz et al., 2007a, 2007b). They served, however, as interesting thought-provoking proposals that influenced later ultra-lightweight RFID designs. In 2007, Chien proposed SASI (Chien, 2007), which aims to provide a better security margin and requires only a tiny footprint. The main contribution was the addition of the bitwise rotations to the set of operations supported on a tag. Despite this twist in the design of the protocol, some attacks were subsequently published (Castro et al., 2009; D'Arco and De Santis, 2008; Cao et al., 2009; Phan, 2009).

In 2008, (Peris-López et al., 2009) introduced a new protocol, called Gossamer and inspired by both the UMAP family and SASI. The operations on tags are limited in this case to bitwise xor, addition and left rotation. A key factor in the design of Gossamer is the inclusion of the MixBits function. This is a very lightweight function with highly non-linear relations between inputs and outputs. A desynchronization attack (Yeh & Lo, 2010) conducted by an active attacker is, to the best of our knowledge, the only attack to date proposed against Gossamer. As an alternative to Gossamer, (Yeh et al., 2010) recently presented a new ultra-lightweight authentication protocol. The protocol is claimed to provide strong security, and to optimize the use of the tag memory in comparison with Gossamer. Contrary to Yeh *et al.*'s claims, the scheme suffers from traceability and full disclosure attacks as shown by Peris-López et al. (2010).

3. David–Prasad authentication protocol

In 2010, David and Prasad proposed an ultra-lightweight authentication protocol claiming strong security suitable for low-cost RFID tags (David & Prasad, 2010). The protocol provided mutual authentication between the tag and the reader using a limited collection of binary operations: and ($\wedge$), or ($\vee$), exclusive or ($\oplus$) and negation($\neg$). Due to the simplicity of these operations, they can be implemented using a low number of logic gates.

David–Prasad protocol keeps two shared secrets, K_1 and K_2 that are initially stored in the tag and in the database as well. Additionally, the tag stores its own static identifier ID. In order to protect the protocol against a traceability attack, tags are not identified by its ID, but rather by a pseudonym P_{ID2} that changes after each session. The old pseudonym (P_{ID1}) is also stored in the tag memory.

Reader and tag exchange a sequence of messages to authenticate each other, this process is named `session`. An authentication session with David–Prasad happens as follows:

1. The reader requests ($C_{request}$) the database a certificate C. The database verifies whether the reader is authorized and in that case it returns the certificate. This step is not required if the reader already has the certificate.
2. The reader sends an $ID_{request}$ to the tag and it replies with its current pseudonym, P_{ID2}.
3. In order to obtain the secrets associated to the tag, the reader sends the tuple $\{P_{ID2}, C\}$ to the database. It assesses the certificate and tries to match the pseudonym with its entries. If it finds a valid entry associated to the tag and authorizes the reader to access the information, it replies with the tuple $\{K_1, K_2\}$. In case that the database could not find an entry for P_{ID2}, the reader repeats the previous step using the old pseudonym P_{ID1} instead of P_{ID2}.
4. Upon receiving the secrets K_1 and K_2, the reader generates two nonces $\{n_1, n_2\}$, calculates the messages $\{A, B, D\}$ and sends them to the tag. The messages are calculated as follows:

$$\begin{aligned} A &= (P_{ID2} \wedge K_1 \wedge K_2) \oplus n_1 \\ B &= (\neg P_{ID2} \wedge K_2 \wedge K_1) \oplus n_2 \\ D &= (K_1 \wedge n_2) \oplus (K_2 \wedge n_1) \end{aligned} \tag{1}$$

5. The tag, using the messages A and B, infers the value of the nonces:

$$\begin{aligned} n_1 &= A \oplus (P_{ID2} \wedge K_1 \wedge K_2) \\ n_2 &= B \oplus (\neg P_{ID2} \wedge K_2 \wedge K_1) \end{aligned} \tag{2}$$

then it tries to authenticate the reader calculating a local version D' of D. If both values coincide, the reader is authenticated and then the tag sends back messages E and F:

$$\begin{aligned} E &= (K_1 \oplus n_1 \oplus ID) \oplus (K_2 \wedge n_2) \\ F &= (K_1 \wedge n_1) \oplus (K_2 \wedge n_2) \end{aligned} \tag{3}$$

6. Once E and F are sent, the tag updates its pseudonym:

$$\begin{aligned} P_{ID1} &= P_{ID2} \\ P_{ID2} &= P_{ID2} \oplus n_1 \oplus n_2 \end{aligned} \tag{4}$$

7. The reader authenticates the tag calculating a local version of F, F'. If $F = F'$, the tag is authenticated and the reader executes the next step, otherwise the authentication fails.
8. The reader extracts the ID from message E:

$$ID = E \oplus (K_2 \wedge n_2) \oplus K_1 \oplus n_1 \tag{5}$$

9. The reader updates the pseudonyms using equation (4) and sends the tuple $\{C, P_{ID1}, P_{ID2}\}$ to the database.
10. Finally, the database updates the entry using the information provided by the reader.

A summary of the protocol can be seen in Figure 1. Despite the authors claims about the secure properties of this protocol, it has been shown that David–Prasad is vulnerable to several kinds of cryptanalysis techniques. One of those techniques, and probably the more powerful, is the Tango attack.

4. The Tango attack

The Tango attack is a passive and efficient attack introduced in Hernández-Castro et al. (2010). It uses the poor diffusion properties of boolean operators to guess the secrets values by observing the messages of the protocol transmitted over the insecure radio channel. Using a Tango attack, it is possible to discover the tag ID using about 30 eavesdropped sessions with negligible computational resources. This attack was also used to attack the YLW (Yeh & Lo, 2010) authentication protocol, and, although the properties of this protocol makes it more secure than David–Prasad, the attack with Tango was devastating: it was possible to discover the tag ID with only a few numbers of eavesdropped authentication sessions.

Some definitions before the presentation of the attack may clarify the explanation. We consider variables as vectors in an m-dimensional space instead of just numerical variables. Secrets in RFID usually consist of 96 bits, so in general we consider that they are vectors in a 96-dimensional space. Since the secrets are kept in binary format, we can represent them as a vector of coefficients a_i that are the values of the vector Z in each dimension.

$$\begin{aligned} z &= \sum_{i=0}^{m-1} a_i \ldots 2^i, \ a_i \in \{0, 1\} \\ Z &= [a_0 \ a_1 \ \ldots \ a_{m-1}] \end{aligned} \tag{6}$$

To emphasize the vectorial nature of the secrets, we will denote them as $\overline{X} \in \{ID, K_1, K_2\}$. The Tango attack uses a set of *approximations*, that is functions $f : Z_2^m \times \cdots \times Z_2^m \to Z_2^m$ that, given the messages exchanged in an eavesdropped session, it computes a vector that eventually would be close to the secret we want to discover. Randomly trying different approximations or with a brute force search, eventually some of them would be able to correctly recover some parts of the secret. These approximations that provide information about the secret are called *good approximations*.

Good approximations might provide, on average, information about all parts of the secret, but usually, some good approximations tend to extract more information in some positions than others. Therefore, using several good approximations in a Tango attack usually yields better results. The price to pay is a slightly higher complexity in the algorithm.

Let us suppose there are a set of good approximations $\mathcal{F}$. Each function f_i in the set $\mathcal{F}$ takes an eavesdropped session $s \in \mathcal{S}$ as an argument and returns a 96-dimensional vector that is likely to approximate the value of $\overline{X}$, so let us name it $\overline{X}^i_{approx}$. The Tango attack with multiple approximations is as follows.

First, it eavesdrops the messages of the first session, s_0, and computes an approximated value of the secret using all the approximations.

$$\overline{X}^i_{approx} = f_i(s_0) \tag{7}$$

Then, the coefficients $\overline{X}^i_{approx} = [a^i_0 \ a^i_1 \ \ldots \ a^i_{m-1}]$ are added to obtain an aggregated approximation of $\overline{X}$, $\overline{X}_{approx}$.

$$\overline{X}_{approx} = \sum_{i=0}^{r-1} [a^i_0 \ a^i_1 \ \ldots \ a^i_{m-1}] \tag{8}$$

This procedure is repeated for all the $|\mathcal{S}|$ eavesdropped sessions, given that $\mathcal{S}$ is the set of all the eavesdropped sessions, and $|\mathcal{S}|$ its size. In this way $\overline{X}_{approx}$ becomes a vector $Z^m_{\mathbb{N}}$ whose coefficients contain the hamming weight of each approximation $\overline{X}^i_{approx}$, that is the sum of ones of each $\overline{X}^i_{approx}$. Of course, the coefficients of $\overline{X}_{approx}$ contain, to some extent, random values corresponding to approximations that were not able to correctly extract the secret. Nonetheless, in average, good approximations will add the correct coefficient to $\overline{X}_{approx}$, meaning that, in average, a_i will contain higher values if the coefficient i of the secret is 1, and lower values otherwise. It leads to a simple heuristic to construct a conjecture about $\overline{X}$: those coefficients in $\overline{X}_{approx}$ whose values exceed a certain threshold γ are set in the secret to one, otherwise they are set to zero.

$$\overline{X}_{conj} = [X^0_{conj} \ X^1_{conj} \ \ldots \ X^{m-1}_{conj}] \tag{9}$$

where

$$X^i_{conj} = \begin{cases} 1 & \text{if} \quad a_i > \gamma \\ 0 & \text{if} \quad a_i < \gamma \end{cases} \tag{10}$$

It was found empirically that a good value for γ is the half of the number of approximations calculated:

$$\gamma = \frac{|\mathcal{F}| \cdot |\mathcal{S}|}{2} \tag{11}$$

The key point in a Tango attack is the selection of the approximations. This selection was initially done with a manual brute-force search as shown in Table 1. This method, despite its simplicity and effectiveness, has a remarkable drawback, it only finds very simple approximations because only a tiny proportion of the search space is explored. This is the motivation of the Genetic Tango attack.

Table 1: *Best approximations calculated by a human. These were proposed in the original Tango attack*

Approximation	*Secret*	$hd(\overline{X}_{conj}, \overline{X})$
$D \oplus E \oplus F$	ID	31.1 ± 3.6
$A \oplus E \oplus F$	ID	22.2 ± 1.7
$B \oplus D \oplus E$	ID	34.0 ± 3.8
D	K_1	34.0 ± 1.9
F	K_1	36.1 ± 3.4
$A \oplus B \oplus D$	K_1	37.6 ± 5.8
$A \oplus B \oplus F$	K_2	36.3 ± 3.0
D	K_2	35.1 ± 3.9
$A \oplus B \oplus D$	K_2	36.8 ± 2.4

5. The Genetic Tango attack

The *Genetic Tango attack* follows the same algorithm described in the previous section, but it introduces GP to find good approximations. GP has been widely used to derive boolean expressions (Koza, 1992), and actually boolean problems have been usually included in test suites for experimental research in GP (Daida et al., 2003). Nonetheless GP has been successfully used in many other fields, achieving human-competitive results (Koza, 2010; Tsai, 2011). If we also consider the representation of the potential solutions in form of trees, classical tree-based GP seems to be a reasonable choice to enhance the method to find approximations.

We should stress that the Genetic Tango attack introduces a new method to find good approximations. There is a need to run this algorithm until the good approximations are found, then, those approximations can be used each time that the protocol is attacked. It is what Eiben and Smith called a design domain (Eiben & Smith, 2009). Therefore, once the approximations have been found, the Tango attack can be performed using them, and hence there is no need to run again the evolutionary algorithm (EA).

The design of our proposed EA is rather classical, following to some extent the *de facto* Koza's standard algorithm. The set of terminals is composed by the public messages in the protocol, $\mathcal{T} = \{A, B, D, E, F, P_{ID1}, P_{ID2}\}$, while the function set is limited to the boolean operations used by the protocol, $\mathcal{F} = \{and, or, xor\}$. The negation operator was removed from the function set because the exploratory experiments showed that it tended to vanish from the population in early generations; so it did not help to find better approximations and increased the search space. The only non-standard element in the algorithm is the use of a bloat control mechanism.

A serious problem usually found in GP is bloating, that is the increase of the tree size without an improvement of the fitness (Luke & Panait, 2006). In order to fight bloating, we used a lexicographic tournament selection (Luke & Panait, 2002) instead of a more classical one. This selection method works similarly to the tournament selection, with one important difference: when it finds two or more individuals with the same fitness, it takes into account their size, selecting the smallest one. In this way, the selection mechanism introduces a parsimony pressure without a need to modify the fitness function or changing the algorithm, like most of the bloat control techniques do (Luke & Panait, 2006).

The use of lexicographic selection presents an interesting side effect: When the population has converged, the lexicographic selection removes longer trees from the population while conserving the population fitness. In practical terms, it means that the lexicographic tournament simplifies the resulting expression. This property is specially interesting since we would like to perform a semantic interpretation of the approximation, which is quite difficult with large individuals.

The rest of the parameters are rather common in GP, the most important ones are summarized in Table 2. The initialization of the population is ramped half-and-half with a depth between 1 and 3. We used these low values to benefit exploration of small individuals in early generations. The course of the evolution favoured bigger individuals, and therefore search space regions associated to bigger individuals are later explored. Exploratory experiments showed that in this way it was more likely to find high-fitness individuals.

New individuals in a generation are obtained using Koza crossover with probability 0.9 and reproduction with probability 0.1. No mutation has been used. Lexicographic tournament is used with a hard limit to the tree depth that crossover can produce (see Table 2). The maximum depth of crossover imposes the maximum size that any individual in the population can reach, and also has a major impact in the search space size. We tried to keep this value as low as possible. For

Table 2: *Main parameters used to obtain the approximations for secrets ID, K_1 and K_2 in the Genetic Tango attack against David–Prasad authentication protocol*

Parameter	ID	K_1	K_2
Population	500	500	500
Generations	10	20	20
Terminal set	$A, B, D, E, F, P_{ID1}, P_{ID2}$	$A, B, D, E, F, P_{ID1}, P_{ID2}$	$A, B, D, E, F, P_{ID1}, P_{ID2}$
Function set	And, or, xor	And, or, xor	And, or, xor
Fitness	Hamming distance to secret	Hamming distance to secret	Hamming distance to secret
Fitness tags	5	5	5
Fitness sessions	100	100	100
Min. depth	1	1	1
Max. depth	3	5	5
Selection	Lexicographic tournament	Lexicographic tournament	Lexicographic tournament
Tournament size	4	4	4
Crossover	0.9	0.9	0.9
Reproduction	0.1	0.1	0.1
Elitism size	1	1	1
Terminals	0.1	0.1	0.1
Non-terminals	0.9	0.9	0.9

Table 3: *Best approximations using GP. The average Hamming distance between the conjecture computed with the approximation and the secret, over 10000 eavesdropped sessions, is also shown*

Approximation	*Secret*	$hd(X_{conj}, X)$
$(D \vee A) \oplus F \oplus E$	ID	12.5
$(((D \wedge F) \vee (B \oplus A)) \wedge ((D \wedge F) \vee (PID2 \wedge PID1))) \vee ((A \wedge F) \vee (D \wedge B))$	K_1	18.2
$(((A \wedge B) \oplus (B \oplus D)) \wedge (F \oplus A)) \vee ((PID1 \oplus A) \oplus (B \oplus PID2))$	K_2	11.3

this reason, for each secret, we first run several times the algorithm with a very small value of the maximum depth, then this value was increased in successive runs until no improvement in the fitness was observed.

Individuals in the population are assessed calculating the Hamming distance (Hamming, 1986) between the secret and the conjecture. In this form, the problem becomes a minimization problem where any individual that achieves a fit of 0 represents an ideal approximation able to correctly recover all the secrets. In an analytical form, the fitness function is given by

$$\mathfrak{F} = hd(\overline{X}_{conj}, \overline{X}) \tag{12}$$

where $\overline{X} \in \{ID, P_{ID1}, P_{ID2}\}$ is the secret, $\overline{X}_{conj}$ is the conjecture given by the approximation and $hd(\overline{A}, \overline{B})$ is the Hamming distance between $\overline{A}$ and $\overline{B}$.

We should mention that this assessment method only considers the distance between the conjecture and the secret. Nonetheless, it does not provide a full characterization of the solution quality in terms of its success in a Tango, there are other factors involved. The Hamming distance only provides a partial picture, the amount of the parts of the secret that the approximation is able to extract, in average terms. However, the location of the discovered parts of the secret also plays a mayor role. That is the reason because the Tango attack uses several approximations, to maximize the probability of recovering parts of the secret placed on different locations. Despite this fact, we have based the fitness function on the Hamming because it simplifies the implementation and hence it requires less computational resources.

So far we have been concerned with the description of the Genetic Tango attack, now we move towards to run the algorithm in order to find good approximations to the David–Prasad authentication protocol.

5.1. Evolutionary search for good approximations

In this section, we report details about how good approximations were found to attack the David–Prasad protocol. For each secret $X \in \{ID, P_{ID1}, P_{ID2}\}$, the algorithm was run 30 times with the configuration shown in Table 2, and we kept the best run. In order to provide a training set of tags to the algorithm, we created it simulating 10000 sessions with 10 random tags. In order to save computational resources and reduce variability, the simulated sessions were stored and used as input of the EA. (All the code, datasets and scripts needed to reproduce the experiments can be found on http://atc1.aut.uah.es/~david/es2012.)

The best evolutionary approximations that we found are reported in Table 3. The quality of the approximations is surprisingly good, the approximation to the ID yields an average Hamming distance of 12.5, while the approximations to K_1 and K_2 yield, respectively, 18.2 and 11.3. These results are more amazing if we compare them with the approximations made by human experts shown in Table 1. The improvement is evident, the best approximation obtained by a human to the ID is 22.2, while GP obtained an approximation with a distance of 12.5, that is GP has reduced the distance around 10 bits, which is a notable difference. But GP has reduced the distance even more for K_2, from 35.1 to only 11.3. It is an improvement of more than 23 bits.

We should mention that the Hamming distance only provides a clue about the real quality of the approximation. A small Hamming distance suggests that the approximation is good, however, depending on the position of the recovered bits, its extraction capabilities in a Tango attack might be different. For this reason, the Hamming distances reported in Table 3 should be interpreted in the context of the EA: it was used to guide the evolutionary process. In order to assess their performance in a Tango attack and its behaviour with unseen data, some additional experimentation is required. Subsection 5.2 deals with this issue.

A representative example of how GP evolves an approximation can be found in Figure 2. It represents the mean and the best-of-run fitness of a run that seeks a good approximation to K_2. The remarkable difference between best and mean fitness found in the first generation suggests that just a random search would perform well to find the approximation, although in further generations we can appreciate that the algorithm presents a significant evolvability [ability to increase above-average fitness (Altenberg, 1994)].

Figure 3 shows the average depth and the average number of nodes of the same run. It clearly shows the presence of code bloat and the effect of the lexicographic selection. There is a clear correlation between the number of nodes and the tree depth, which is something we could expect because all non-terminal nodes are binary, and thus the increase of the number of nodes is achieved through deeper trees. Although Figure 3 shows an increment of the average tree depth and

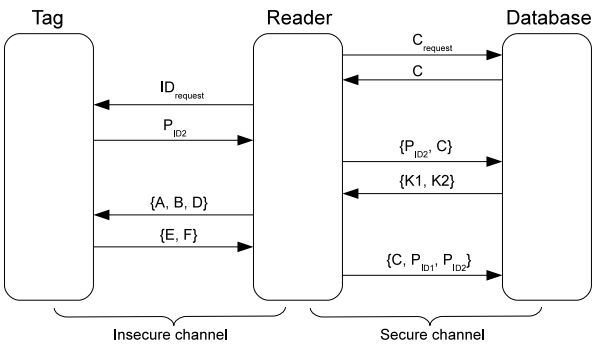

Figure 1: *Summary of the messages exchanged in David–Prasad authentication protocol when all the steps are correctly executed. Lightweight Cryptography only cares about the messages between the tag and the reader.*

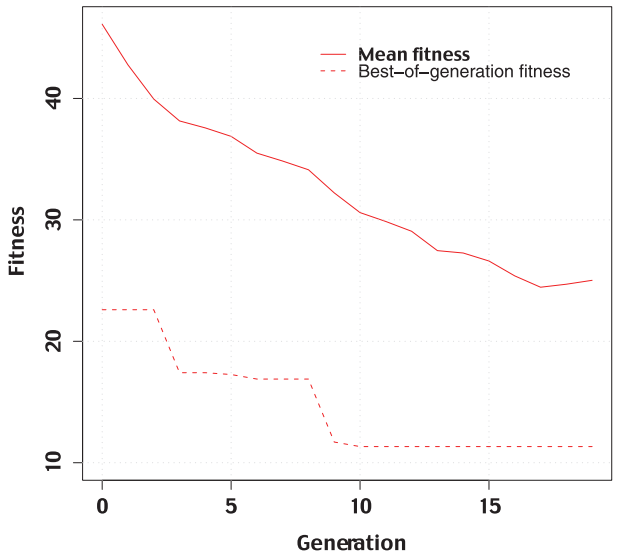

Figure 2: *Best fitness (upper) and average tree depth (bottom) in the evolution of a K_2 good approximation. These curves represents the best run.*

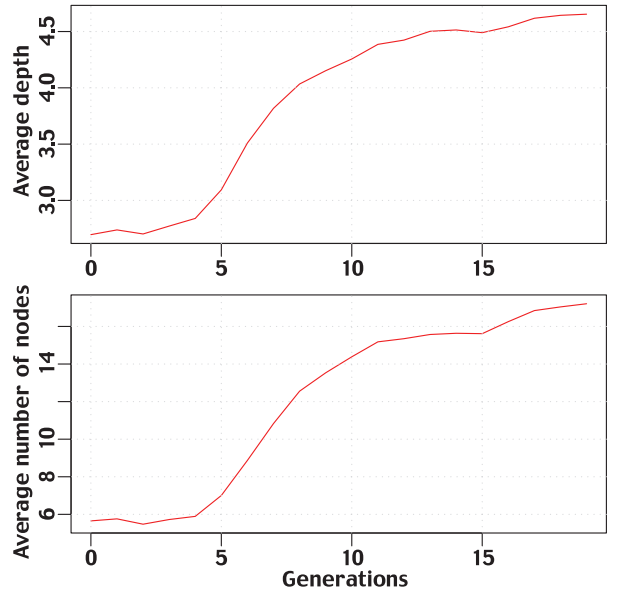

Figure 3: *Average tree depth (upper) and average number of nodes (bottom) in the evolution of a K_2 good approximation. These are representatives curves corresponding to one run.*

number of nodes, there are some regions where this grown is stopped, if not inverted. Generations 2 and 15 are examples of this fact.

The good approximations shown in Table 3 are supposed to reveal information about the secrets, and provide a way to attack the David–Prasad authentication protocol. However, in order to prove this hypothesis, we have to check if those approximations are able to extract information about the secrets. At the same time, it is not clear yet if the small Hamming distance obtained by the approximation correlates well with good secret extraction capabilities and if the approximations are able to perform well with different datasets. We try to provide an answer to these two questions in the next subsection.

5.2. Tango attack against David–Prasad with evolved approximations

Once the EA has provided some good approximations, it is necessary to assess its effectiveness in a Tango attack. To this end, in this section we compare the extraction capabilities of the manual and evolved approximations. In order to make a fair comparison, and avoid bias induced by the data, all the experiments reported in the following have used different datasets to the ones used to feed the EA. To better appreciate the approximation properties and their differences, we have performed the Tango attack using a single approximation for each secret.

Figure 4 shows a boxplot of the number of bits correctly recovered using four different approximations, including the one obtained with GP. For each plot, we have simulated the authentication of 10000 random tags. Looking at Figure 4, we observe that GP generates an approximation that clearly outperforms the best manual approximations, and in fact the result of the attack is dramatic. With only five sessions, in average it is possible to discover 88 bits, with five sessions more, this value grows to 93 bits. Almost all the bits of the ID are discovered with only 20 sessions. And it is done with only one approximation, on the contrary than the original Tango attack, where several approximations for each secret were used.

The performance of the GP approximation is more impressive if we compare it with the manual approximations. They are able to discover, in the best case, an average of 72 bits from the secret. Moreover, manual approximations present the interesting characteristic that, when used individually in the attack, they are unable to increase the knowledge about the secret even when there is more information available. Using more eavesdropped sessions in the attack with individual manual approximations does not improve it. This is not the case of the evolved approximation, which clearly benefits from the presence of more eavesdropped sessions. One possible explanation to this fact was previously introduced, but it is worth to look at it in more detail.

Let us consider an extreme and unrealistic (but didactic) example, an approximation that always recovers one and only one bit per eavesdropped session. This approximation might be a serious threat to the protocol, or not, depending on which bit it discovers. If the bit is always the same one, the approximation would provide one bit of the secret with only one session, but the adversary could not obtain further information, and thus this attack could have a limited effect. Now let us suppose that the approximation were able to discover one bit in each session, but in different locations. Then the adversary, given enough sessions, would have information about the whole secret, and thus it could be used to obtain the secret, which could be a serious security threat.

This fact explains the differences of behaviour found in Figure 4. The evolved approximation increases the number of correctly recovered bits as it had more eavesdropped sessions, while manual approximations do not show any relation between the number of bits that they discover and the number of sessions they have available. This observation suggests that the evolved approximation is able to find bits of the ID in different locations of the secret. Surprisingly, the fitness function used to evolve the approximation did not explicitly use

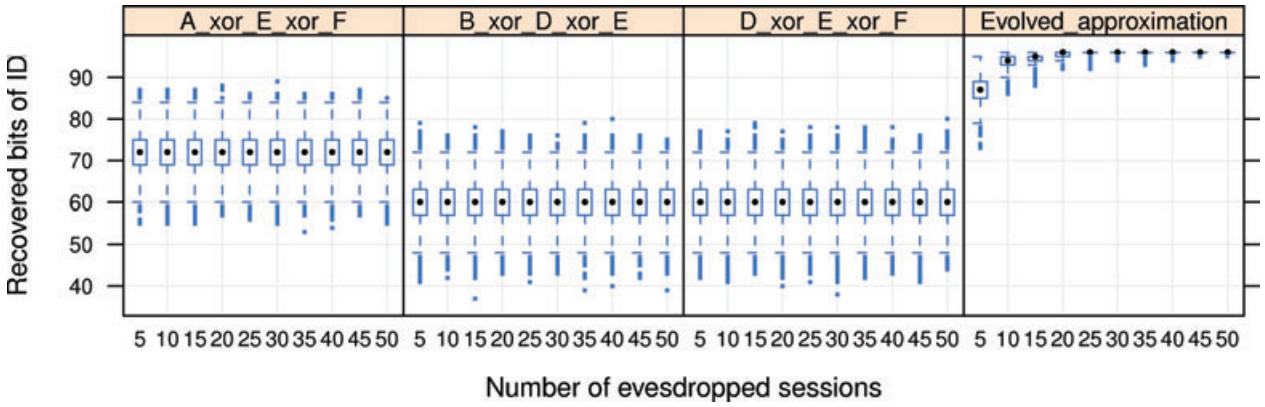

Figure 4: *Number of bits recovered from ID using different good approximations against the number of eavesdropped sessions. The fourth panel represents the approximation calculated by GP ($(D \vee A) \oplus F \oplus E$).*

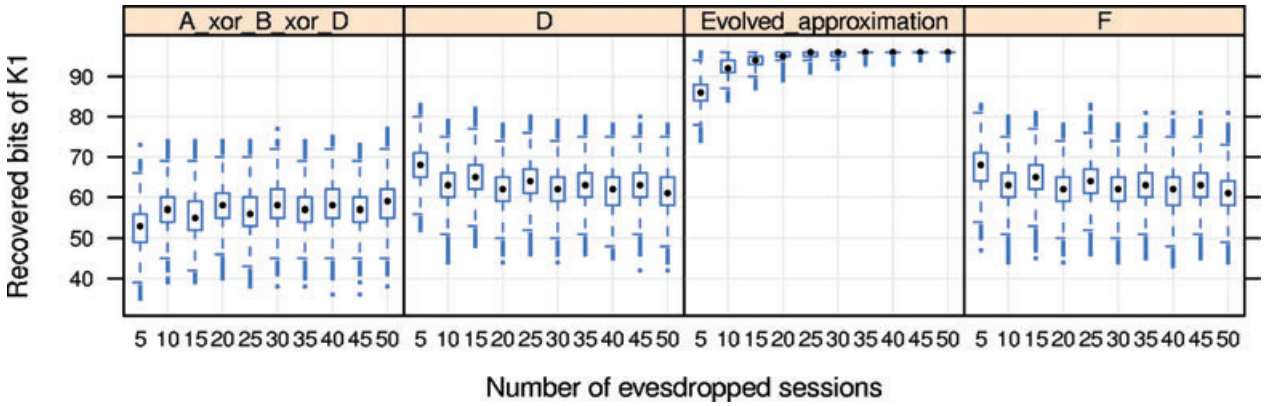

Figure 5: *Number of bits recovered from K_1 using different good approximations against the number of eavesdropped sessions. Third panel represents the approximation calculated by GP ($(((D \wedge F) \vee (B \oplus A)) \wedge ((D \wedge F) \vee (PID2 \wedge PID1))) \vee ((A \wedge F) \vee (D \wedge B))$).*

this criteria to press the population towards this objective. It is actually an unexpected, interesting and very welcomed side effect.

A similar behaviour can be observed in the secrets K_1 and K_2, depicted, respectively, in Figures 5 and 6. The evolved approximation to K_1 is able to extract almost all the 96 bits of the secret with only 20 sessions, while the evolved approximation to K_2 performs worse, in average it recovers around 85 bits. In any case, evolved approximations outperform any of the manual approximations.

Interestingly, the evolved approximation to K_2 that achieved less distance to the secret, only 11.3 (see Table 3), is also the one less able to reveal the secret. Although evolved approximations to ID and K_1 obtained worse distances, respectively 12.5 and 18.3, the Tango cryptanalysis performs better, which seems a contradiction, but is logical under the light of the previous discussion about the role of the position of the discovered bits.

The overall performance of the evolved approximations to the ID, K_1 and even K_2 are excellent, they are able to

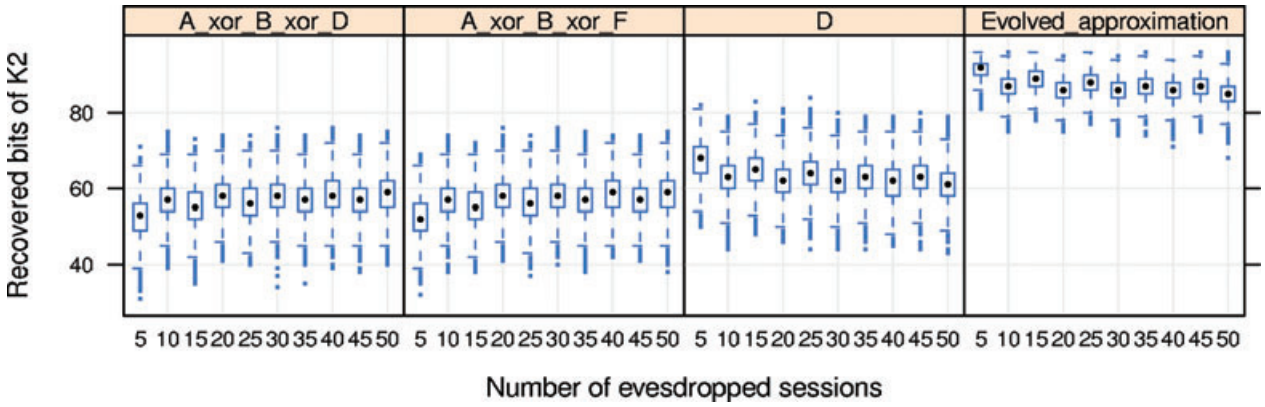

Figure 6: *Number of bits recovered from K_2 using different good approximations against the number of eavesdropped sessions. Fourth panel represents the approximation calculated by GP ($(((A \wedge B) \oplus (B \oplus D)) \wedge (F \oplus A)) \vee ((PID1 \oplus A) \oplus (B \oplus PID2))$).*

Table 4: *List of approximations used in the Tango attack with several approximations, including the evolved approximation and manual approximations reported in the literature*

Secret	*Approximations*
ID	$(D \vee A) \oplus F \oplus E$
	$D \oplus E \oplus F$
	$A \oplus E \oplus F$
	$B \oplus D \oplus E$
K_1	$(((D \wedge F) \vee (B \oplus A)) \wedge ((D \wedge F) \vee (PID2 \wedge PID1))) \vee ((A \wedge F) \vee (D \wedge B))$
	D
	F
	$A \oplus B \oplus D$
K_2	$(((A \wedge B) \oplus (B \oplus D)) \wedge (F \oplus A)) \vee ((PID1 \oplus A) \oplus (B \oplus PID2))$
	D
	$A \oplus B \oplus F$
	$A \oplus B \oplus D$

outperform the manually generated approximations. How GP has improved the approximation is particularly clear in the case of K_1, with a very limited number of sessions, and using only one approximation in the attack, it is possible to recover almost all the bits of K_1.

Summarizing the results, looking at the experimental data, we can conclude that the approximations obtained using GP clearly outperform those computed manually. Nonetheless, these results perhaps could be improved. The previous experience using a Tango attack showed that including several approximations in the attack helps to discover more bits of the secret with less sessions. So, we can expect that mixing manual and evolved approximations would enhance this cryptanalysis tool. We verify experimentally this hypothesis.

5.3. Genetic Tango attack with several approximations

Once that we have verified experimentally the extraction properties of the evolved approximations, it is worth to verify its behaviour when it is used in a Tango attack with other approximations. The original Tango attack uses several approximations in order to join its extraction capabilities. It is supposed that each approximation is likely able to extract only a part of the secret, using several ones, it is more likely to be able to extract more parts of the secret. In this section, we test that conjecture.

In order to verify this point, we have run two different Tango attacks against David–Prasad. The first attack, that we name *partial attack*, uses the approximations shown in Table 1. The second attack includes the same approximations of the partial attack, but it also includes the evolved approximations shown in Table 3. A list of the approximations used in the *full Tango attack* can be found in Table 4. The experimental procedure is analogous to the previous one. We have generated sessions for 10000 random tags, and then we attacked the protocol with the two variants of the Tango attack previously described. The number of sessions used in the attack ranges from 5 to 50.

The result of the attack is summarized in Figure 7. This figure depicts the number of recovered bits of the three secrets with the full attack (left column) and the partial attack (right column). The effect of using the evolved approximations is clear, in all the cases the number of recovered bits is increased in comparison with not using the evolved approximation. In addition, the full attack can learn, and more sessions lead to reveal more details about the secrets. The number of bits recovered with the partial attack, on the contrary, remains almost constant, and there is no evident benefit from eavesdropping more sessions.

Comparing the Tango attack with one or several approximations reveals some interesting information. Figure 7, in comparison to Figures 4–6, shows that the evolved approximations alone outperform the attack performed with several approximations, even those that include the evolved approximation. To illustrate this fact, we can observe that, for instance, the evolved approximation in average is able to obtain close to 96 bits from the ID with only 10 sessions (see Figure 4), while if the evolved approximation is used with the other good approximations, this value raises to around 83 bits. Therefore, the approximations degenerate the performance of the evolved approximation.

This result does not implies that a Tango attack with several approximations is worse than a Tango attack with one evolved approximation. There are several factors involved: the attacked protocol, the approximations and so on. However, in this case, with the evolved approximations shown in Table 3, and in comparison to the approximations obtained by human experts, the evolved approximations perform better alone in a Tango attack.

6. Conclusions and future work

We have proposed an improvement in the Tango attack named Genetic Tango attack. It is based on the use of GP to find good approximations that are used in a Tango attack. Compared to a manual search, GP offers a method to automatically search the approximation with a very limited human intervention, just tuning some parameters. The algorithm allows a more intense exploration and exploitation of the search space. As a result, GP finds better approximations and subsequently the Tango attack is improved.

In order to verify the performance of our proposal, we attacked the David–Prasad authentication protocol using the Genetic Tango attack. Using GP, we were able to find better approximations than the ones reported by the literature, so, in this protocol GP is able to outperform the results obtained by humans. When the Tango attack is performed using the evolved approximations, the result is that the adversary is able to extract more bits of the secret with less sessions than using manual approximations. The result of the attack is devastating, with only some eavesdropped sessions, the adversary can discover all the bits of the ID and K_1. In contrast, K_2 is harder to reveal, but in any case the Genetic Tango attack is able to break David–Prasad with only few sessions.

The approximation to the ID was found in a relatively small search space. Nonetheless, in other cases the search space was considerably larger, and the approximation found was rather complex, and therefore it seems unlikely that a random search could have been able to find it. In any case, the metaheuristic introduced in the Tango attack has shown to be very effective finding good approximations for the David–Prasad authentication protocol and clearly outperforms the manual search originally proposed in the Tango attack.

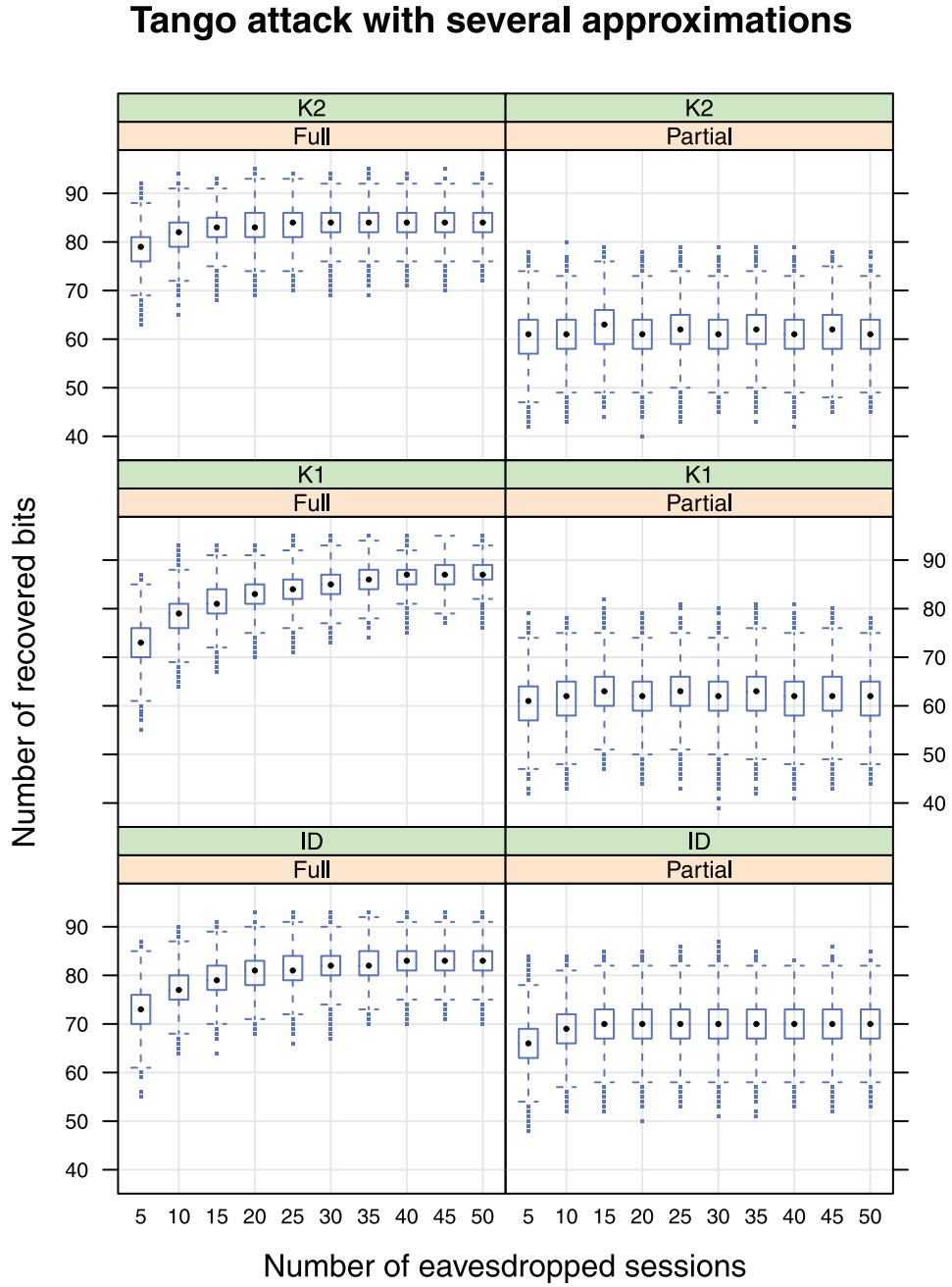

Figure 7: *Tango attack with several approximations. Full attack (left column) is done with the best evolved approximations shown in Table 4, while the partial attack (right column) was done without the evolved approximations (Table 1).*

We have shown that the Genetic Tango attack successfully breaks the David–Prasad authentication protocol, and in doing so is capable of outperforming already published attacks. To study the generality of the Genetic Tango attack, we plan to attack new and stronger protocols using the proposed cryptanalysis tool. In addition, it would be interesting to determine the set of protocols that are vulnerable to a Genetic Tango attack from both a theoretical and empirical perspective. Additionally, we plan to use our approach for testing the security of our own new designs, in this way being sure of the robustness of our future proposals. All in all, we believe that we have presented here a very powerful tool for the study of the security of many new ultra-lightweight protocols that have blossomed in the literature in recent years.

Acknowledgement

This work was partially supported by the MICYT project ABANT (TIN2010-19872).

References

ALTENBERG, L. (1994) *The Evolution of Evolvability in Genetic Programming*. Cambridge, MA: MIT Press, 47–74.

BÁRÁZ, M., B. BOROS, P. LIGETI K. LÓJA and D. NAGY (2007a) Breaking LMAP, in *Proceedings of RFIDSec'07*.

BÁRÁZ, M., B. BOROS, P. LIGETI, K. LÓJA and D. NAGY (2007b) Passive attack agains the M2AP mutual authentication protocol for RFID tags, in *Proceedings of First International EURASIP Workshop on RFID Technology*.

BROWN, D. (2007) *RFID Implementation*, 1st edn, New York, NY: McGraw-Hill, Inc.

Cao, T., E. Bertino and H. Lei (2009) Security analysis of the sasi protocol. *IEEE Transactions on Dependable and Secure Computing*, **6**, 73–77.
Castro, J.C.H., J.M. Estévez-Tapiador, P. Peris-López, T. Li and J.J. Quisquater (2009) Cryptanalysis of the SASI ultralightweight RFID authentication protocol with modular rotations, in *International Workshop on Coding and Cryptography*. Lofthus, Norway, 286–296.
Chien, H.Y. (2006) Secure access control schemes for RFID systems with anonymity, in *Proceedings of the 7th International Conference on Mobile Data Management*, Washington, DC: IEEE Computer Society, 96–99.
Chien, H.Y. (2007) SASI: a new ultralightweight RFID authentication protocol providing strong authentication and strong integrity, *IEEE Transactions on Dependable and Secure Computing*, **4**, 337–340.
Chien, H.Y. and C.W. Huang (2007) Security of ultra-lightweight RFID authentication protocols and its improvements, *SIGOPS Operating Systems Review*, **41**, 83–86.
Cole, P.H. and D.C. Ranasinghe (2008) *Networked RFID Systems and Lightweight Cryptography: Raising Barriers to Product Counterfeiting*, 1st edn, Berlin, Heidelberg: Springer-Verlag.
Daida, J.M., H. Li, R. Tang and A.M. Hilss (2003) What makes a problem GP-hard? Validating a hypothesis of structural causes, in *Proceedings of the 2003 International Conference on Genetic and Evolutionary Computation: Part II*, Berlin, Heidelberg: Springer-Verlag, 1665–1677.
D'Arco, P. and A. De Santis (2008) Weaknesses in a recent ultra-lightweight rfid authentication protocol, in *Proceedings of the Cryptology in Africa 1st international conference on Progress in cryptology*, Berlin, Heidelberg: Springer-Verlag, 27–39.
David, M. and N.R. Prasad (2010) Providing string security and high privacy in low-cost RFID networks, in *Proceedings of Security and Privacy in Mobile Information and Communication Systems*, Berlin, Heidelberg: Springer-Verlag, 172–179.
Eiben, A.E. and J.E. Smith (2009). *Introduction to Evolutionary Computing*. Natural Computing, Berlin, Heidelberg: Springer-Verlag.
Hamming, R.W. (1986) *Coding and information theory*, 2nd edn, Upper Saddle River, NJ: Prentice-Hall, Inc.
Hernández-Castro, J.C., P. Peris-López, R.C.W. Phan and J.M.E. Tapiador (2010) Cryptanalysis of the David-Prasad RFID ultralightweight authentication protocol, in *Proceedings of the 6th international conference on Radio frequency identification: security and privacy issues*, Berlin, Heidelberg: Springer-Verlag, 22–34.
Koza, J. (1992) *Genetic Programming: On the programming of Computers by Means of Natural Selection*. Cambrige, MA: MIT Press.
Koza, J. (2010) Human-competitive results produced by genetic programming. *Genetic Programming and Evolvable Machines*, **11**, 251–284.
Landt, J. (2005). The history of RFID, *Potentials, IEEE*, **24**, 8–11.
Li, T. and G. Wang (2007) Security analysis of two ultra-lightweight RFID authentication protocols, in *Proceedings of IFIP SEC 2007*, Sandton, Gauteng, South Africa 14–16.
Luke, S. and L. Panait (2002) Lexicographic parsimony pressure, in *Proceedings of GECCO-2002*, New York, NY, San Francisco, CA: Morgan Kaufmann Publishers, 829–836.
Luke, S. and L. Panait (2006) A comparison of bloat control methods for genetic programming, *Evolution Computing*, **14**, 309–344.
Oh, R. and J. Park (2008) A development of active monitoring system for intelligent rfid logistics processing environment, in *Proceedings of the 2008 International Conference on Advanced Language Processing and Web Information Technology*, Washington, DC: IEEE Computer Society, 358–361.
Peris-López, P., J.C. Hernández, J.M. Estévez-tapiador and A. Ribagorda (2006a) M 2 AP: a minimalist mutual-authentication protocol for low-cost RFID tags, in *Proceedings of International Conference on Ubiquitous Intelligence and Computing UIC06*, LNCS 4159, Berlin, Heidelberg: Springer-Verlag, 912–923.
Peris-López, P., J.C. Hernández, J.M.E. Tapiador and A. Ribagorda (2006b) LMAP: a real lightweight mutual authentication protocol for low-cost RFID tags, in *Proceeding of second Workshop on RFID Security, Ecrypt*.
Peris-López, P., J.C. Hernández-Castro, R.C.W. Phan, J.M.E. Tapiador and T. Li (2010) Quasi-linear cryptanalysis of a secure RFID ultralightweight authentication protocol, in *6th China International Conference on Information Security and Cryptology*. Shanghai, China, Berlin, Heidelberg: Springer-Verlag, 427–442.
Peris-López, P., J.C. Hernández-Castro, J.M. Tapiador and A. Ribagorda (2009) Advances in ultralightweight cryptography for low-cost RFID tags: Gossamer protocol, in *Information Security Applications*, Berlin, Heidelberg: Springer-Verlag, 56–68.
Phan, R. (2009) Cryptanalysis of a new ultralightweight rfid authentication protocolsasi, *IEEE Transactions on Dependable and Secure Computing*, **6**, 316–320.
R-Moreno, M.D., B. Castaño, M. Carbajo, Á. Moreno, D.F. Barrero and P. Muñoz (2011) Multi-agent intelligent planning architecture for people location and orientation using RFID, *Cybernetics and Systems*, **42**, 16–32.
Tsai, H.C. (2011) Using weighted genetic programming to program squat wall strengths and tune associated formulas. *Engineering Applications of Artificial Intelligence*, **24**, 526–533.
Yeh, K.H. and N. Lo (2010) Improvement of two lightweight RFID authentication protocols. *Information Assurance and Security Letters – IASL*, **1**, 6–11.
Yeh, K.H., N. Lo and E. Winata (2010) An efficient ultralightweight authentication protocol for RFID systems, in *Workshop on RFID Security – RFIDSec Asia'10*, Singapore, Republic of Singapore: IOS Press, 49–60.

The authors

David F. Barrero

David F. Barrero is Lecturer in the Computer Engineering Department of the Universidad de Alcalá (UAH). He holds a Telecommunications Engineering degree and he received a PhD in Computer Science with the distinction of European PhD. He has made research stays in the Centre National d'Etudes Spatiales (CNES) in Toulouse, France, and the University of Portsmouth, in the United Kingdom. His research interests include experimental methods in Evolutionary Computation, run-time analysis and clustering techniques.

Julio César Hernández-Castro

Julio Cesar Hernandez-Castro is now Senior Lecturer in the School of Computing of Portsmouth University in the United Kingdom. He has previously worked as an Associate Professor at Carlos III University in Madrid, Spain. His interests include Cryptography and Cryptanalysis, where he has always tried to apply a non-conventional approach based in the use of Nature-based Computing. He is also interested in RFID Security, particularly Ultralightweight protocols, and in Computer Forensics, Steganography, Steganalysis and Data Leakage Prevention. He is a keen Chess player (peak ELO rating of 1912 so far), and likes Astronomy, Recreational Mathematics and Magic tricks.

Pedro Peris-Lopez

Pedro Peris-Lopez has an MSc in Telecommunications Engineering and PhD in Computer Science. His research interests are in the field of protocols design, primitives design, lightweight cryptography, cryptanalysis, etc. Nowadays, his research is focused on Radio Frequency Identification Systems (RFID) and Implantable Medical Devices (IMD). In

these fields, he has published a great number of papers in specialized journals and conference proceedings. He has been cited over 850 times and his h-index is 13.

David Camacho

David Camacho is currently working as Associate Professor in the Computer Science Department at Universidad Autónoma de Madrid (Spain). He has published over 100 journals, books and conference papers. His research interests include Multi-Agent Systems, Videogames and Virtual Worlds, Data mining (Evolutionary Computation, Classification and Clustering methods) and Semantic Web Technologies. He is currently involved in several research projects related to Videogames, Virtual Worlds, Data Analysis and Advanced Clustering techniques. He is currently the Head of the Applied Intelligence & Data Analysis (AIDA) Group: aida.ii.uam.es.

María D. R-Moreno

Maria Dolores R-Moreno received a PhD in Computer Science from Universidad de Alcalá (UAH) in Madrid with the distinction of European PhD. She has spent one year at NASA Ames Research Center as a postdoc, and 9 weeks as a Research Visitor at ESA's European Space Research and Technology Centre (ESTEC). She has served in the program committee of several international AI conferences and reviewer of international journals. She has published over 100 journal, books and conference papers. She is currently Associate Professor in the Engineering Department in the UAH. She is actively collaborating in ESA projects and participating with research groups at NASA Ames and JPL. Her research focuses on Robotics, Automated AI Planning & Scheduling, Monitoring and Execution applied to real applications (i.e. aerospace, e-learning or the web), Evolutionary Computation and Clustering and Classification techniques.