Chapter 3

Handling Security Threats to the RFID System of EPC Networks

Joaquin Garcia-Alfaro, Michel Barbeau, and Evangelos Kranakis

Contents

3.1 Introduction

Passive radio frequency identification (RFID) is a wireless communication technology that allows the automatic identification of objects, animals, and persons through radio waves. Passive RFID tags are electronic labels without self-power supply. They are energized by the electromagnetic field of radio frequency (RF) front-end devices (hereinafter referred as RFID readers). The radio spectrum used in RFID systems varies from low-frequency (LF) and high-frequency (HF) bands (typically 125 kHz and 13.56 MHz) to ultra-high-frequency (UHF) bands (typically 868 MHz in Europe, 915 MHz in North America, and 950 MHz in Japan). Distances from which the RFID tags can be interrogated vary with the frequency band. It may vary from a few centimeters, while using LF and HF, to a few meters, while using UHF.

Although no single technology is ideal for all applications [1], most of the modern RFID systems seem to be moving toward increasing the integration of long-distance passive tags into self-organizing wireless applications. This is the case with the modern electronic product code (EPC) Gen2 tags. They are becoming truly pervasive in wireless network applications, such as Mobile Wireless *Ad Hoc* Networks (MANETs), Wireless Sensor Networks (WSNs), and Vehicular *Ad Hoc* Networks (VANETs) [2]. Tags are potentially the targets of attack against their security and this raises major concerns. The objective of this chapter is to analyze some of these concerns and survey solutions that handle them.

3.1.1 Background

The EPC technology originates from the MIT's Auto-ID Center (now called the Auto-ID Labs). It had been further developed by different working groups at EPCglobal Inc. [3]. It is a layered service-oriented architecture to link objects, information, and organizations via Internet technologies. At the lowest layer, an identification system based on passive RFID tags and readers provides the means to access and identify objects in motion. This system possesses two primary interfaces: the Class 1 Generation 2 UHF Air Interface Protocol Standard (Gen2 for short) and Low Level Reader Protocol (LLRP). The former defines the physical and logical requirements for RFID readers (or interrogators) and passive tags (or labels). The latter specifies the air interface and interactions between its instances.

The next layer consists of a middleware composed of several services (such as filtering, fusion, aggregation, and correlation of events) that perform real-time processing of tag event data and collect the identifier of objects interrogated by RFID readers at different time points and locations. Data gathered by sensors, such as temperature and humidity, can also be aggregated at the middleware layer within tag events. The middleware forwards the complete set of events to a local repository where they are persistently stored (e.g., into a relational or XML database). The Reader Protocol (RP) and Reader Management (RM) interfaces define the interactions between a device capable of reading/writing RFID tags and the middleware. The middleware relies on a second interface called Application Level Event (ALE) for interaction with other applications (e.g., repository managers). At the top of the architecture, the EPC Information Services (EPCISs) offer the means to access the data stored in EPC network repositories. These EPCISs are implemented using standard Web technologies such as the Simple Object Access Protocol (SOAP) and Web Services Description Language (WSDL). Two additional services are defined for accessing the EPCIS of a given EPC network by external applications: a lookup service binding object identifiers and EPCISs, called the Object Name Service (ONS); and a EPC discovery service (EPCDS) to perform searches with high-level semantics (i.e., similar to Web engines for Web page browsing).

Security attacks can target the different services of the EPC network architecture. They may succeed if weaknesses within the underlying technologies are not handled properly. The exchange of information between EPC tags and readers, for example, is carried out via wireless channels that do not posses basic security attributes such as authenticity, integrity, and availability. This situation allows attackers to misuse the RFID service of an EPC network and perform unauthorized activities such as eavesdropping, rogue scanning, cloning, location tracking, and tampering of data. The attacker motivation for performing these activities is potentially high. The attacker can obtain financial gains (e.g., offering services for corporate espionage purposes). Mechanisms at the RFID level of the EPC architecture must be applied to mitigate these security risks.

The implementation of new security features in EPC tags faces several challenges, the main one being cost. The total cost of an EPC tag was estimated in [4] to be less than 10 cents per unit. The goal is to maintain a low cost. Other challenges include compatibility regulations, power consumption, and performance requirements [5]. In this chapter, we analyze threats to the security of the exchange of information between RFID readers and tags. Some of them need to be handled by appropriate countermeasures. Our threat analysis is based on a methodology proposed by the European Telecommunications Standards Institute (ETSI). It proposes the ranking of threats depending on their likelihood of occurrence, their possible impact on targeted systems, and the risk they represent [6] for corporate systems. The results of our analysis are intended for leading further research and developments of security of EPC-based technologies. We also study countermeasures for threats ranked at the critical or major level. We discuss the benefits and drawbacks associated with the surveyed solutions.

Section 3.2 outlines the methodology used for our analysis of threats. Section 3.3 presents the identified threats and their risk assessments. Section 3.4 surveys traditional security defenses for RFID solutions. Section 3.5 discusses some directions and trends for further research.

3.2 Threat Analysis Methodology

We define a threat as the objective of an attacker to violate security properties of a target system, such as authenticity, integrity, and availability. We define the attacker as an agent that is exploiting a vulnerability of the targeted system to carry out the threat. The exploitation of the vulnerability is defined as the attack. The security officer of the target system must put in place countermeasures to reduce the risk of the undesirable activities associated with all the threats. Given the difficulty of implementing countermeasures for every possible threat against a system, it is crucial for security officers to identify threats with potentially high impact and insure the presence of countermeasures. This is indeed the objective of the threat analysis.

The methodology we use is based on a framework proposed by the ETSI [6]. ETSI identifies three levels of threats: critical, major, and minor. Each level depends on estimated values for the likelihood of occurrence of the threat and its potential impact on a given system. The likelihood of a threat (*cf.* Figure 3.1a) is determined by the motivation for an attacker to carry out an attack

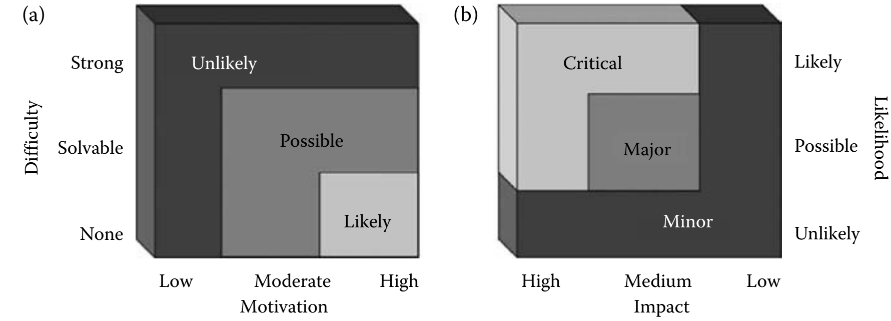

Figure 3.1 Likelihood and risk functions: (a) likelihood of a threat, (b) risk evaluation function.

associated to the threat versus the technical difficulties that must be resolved by the attacker to effectively implement the attack. The three levels of likelihood are: (1) *likely*, if the targeted system is almost assured of being victimized, given a high attacker motivation (e.g., financial gains as a result of selling private information or disrupting network services) and lack of technical difficulties (e.g., a precedent for the attack already exists); (2) *possible*, if the motivation for the attacker is moderate (e.g., limited financial gains) and technical difficulties are potentially solvable (e.g., the required theoretical and practical knowledge for implementing the attack is available); and (3) *unlikely*, in case there is little motivation for perpetrating the attack (e.g., few or no financial gains resulting from the attack) or if significant technical difficulties and obstacles must be overcome (e.g., theoretical or practical elements for perpetrating the attack are still missing).

The impact of a threat evaluates the potential consequences on the system when the threat is successfully carried out. The following three categories are identified: (1) *low*, if the consequences of the attack can be quickly repaired without suffering from financial losses; (2) *medium*, if the consequences are limited in time but might result in few financial losses; and (3) *high*, if the attack results in substantial financial loss and/or law violations. The risk of a threat is ranked in [6] as *minor*, if it is unlikely to happen and it has low or medium potential impact, or if it is possible but with low potential impact. A threat is ranked as *major* if it is likely but has low potential impact, if it is possible and has medium potential impact, or if it is unlikely but has high potential impact. A threat is ranked as *critical* if it is likely and has high or medium potential impact, or if it is possible and has high potential impact. Through our experience with the ETSI methodology, we have observed that several threats are overclassified as major, when they would better be ranked as minor. We have slightly adapted the risk function in order to focus on truly critical or major threats. Figure 3.1b presents the adapted risk function. A threat is ranked as *major* when its likelihood is *possible* and its potential impact is *medium*. A threat is ranked as *minor* when it is *unlikely* to happen or when its potential impact is *low*. Minor risk threats typically require no countermeasures. Major and critical threats need to be handled with appropriate countermeasures. Moreover, critical threats should be addressed with the highest priority.

3.3 Evaluation of Threats

The communication channel between the components of the RFID system of an EPC network, that is, tags and readers, is a potentially insecure wireless channel. It is fair to assume that most of the threats on EPC configurations are going to target this level. We analyze threats targeting basic security features such as authenticity, integrity, and availability during the exchange of data between a RFID tag and a RFID reader. We assume that attackers may only act from outside when trying to exploit the wireless channel between tags and readers, for example, the lack of authentication between these elements. We therefore assume that attackers do have physical access neither to the components of the system nor to the organization itself. The reason we ignore direct physical access is because we assume the presence of other security mechanisms in the organization (e.g., physical access control and surveillance of workers). Attackers, however, may have access to information about the system and its components or services. We summarize in Table 3.1 the results of our evaluation.

3.3.1 Authenticity Threats

The EPC Gen2 standard is designed to balance cost and functionality [4]. However, security features on board Gen2 tags are minimal. They protect message integrity via 16-bit Cyclic Redundancy

Table 3.1 Evaluation of Threats

Threats	*Motivation*	*Difficulty*	*Likelihood*	*Impact*	*Risk*
Eavesdropping, rogue scanning	High	Solvable	Possible	High	Critical
Cloning of tags, location tracking	Moderate	Solvable	Possible	Medium	Major
Tampering of data	Moderate	Solvable	Possible	High	Critical
Destruction of data, denial of service	Moderate	Solvable	Possible	Medium	Major
Malware	Moderate	Strong	Unlikely	Medium	Minor

Codes (CRC) and generate 16-bit pseudorandom strings. Their memory, very limited, is separated into four independent blocks: reserved memory, EPC data, Tag Identification (TID), and user memory. The absence of strong authentication on the tags opens the door to malicious readers that can impersonate legal readers and perform eavesdropping attacks. Figure 3.2 shows a simplified description of the steps of the Gen2 protocol for product inventory. In Step 1, the reader queries the tag and selects one of the following options: select, inventory, or access [3]. Figure 3.2 represents the execution of an inventory query. It assumes that a select operation has been completed in order to single out a specific tag from the population of tags. When the tag receives the inventory query, it returns a 16-bit random string denoted as RN16 in Step 2. This random string is temporarily stored in the tag memory. The reader replies to the tag in Step 3 with a copy of the random string, as an acknowledgment. If the echoed string matches the copy of RN16 stored in the tag memory, the tag enters the acknowledged state and returns the EPC.

Let us observe that any compatible Gen2 reader can access the EPC. The traffic between tags and readers flows through nonauthenticated wireless channels. Illegitimate collection of traffic might be slightly protected by reducing the transmission power or by sheltering the area. It is, although, theoretically possible to conduct eavesdropping attacks. We define forward eavesdropping as the passive collection of queries and commands sent from readers to tags; and backward eavesdropping as the passive collection of responses sent from tags to readers. Although the range for backward eavesdropping could be only of a few meters [3], and probably irrelevant for a real eavesdropping attack, the distance at which an attacker can eavesdrop the signal of an EPC reader can be much longer. In

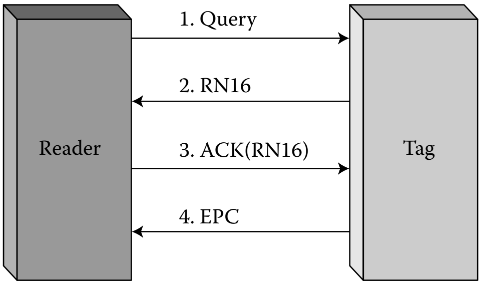

Figure 3.2 Inventory protocol of a Gen2 tag.

ideal conditions, for example, readers configured to transmit at maximum output power, the signal could be received from tens of kilometers away. Analysis attacks inferring sensitive information from forward eavesdropping, for example, analysis of the pseudorandom sequences generated by the tags (denoted as RN16 in Figure 3.2), are hence possible. Replay attacks enabled by this inferred data are also possible. The absence of a strong authentication process also enables scanning attacks. Although, the distance at which an attacker can perform scanning is considerably shorter than the distance for forward eavesdropping. The use of special hardware (e.g., highly sensitive receivers and high gain antennas) could enable rogue scanning attacks.

We can conclude that outsiders equipped with Gen2 compatible readers and special hardware can theoretically eavesdrop the communication between readers and tags; or scan objects in motion if they successfully manage to place their readers at appropriate distances. According to [3], the information stored on an EPC tag is limited to an identification number. No additional data beyond the number itself is conveyed in the EPC. Additional information associated with the code must be retrieved from an EPCIS. However, an attacker accessing these data may determine types and quantities of items in a supply chain and sell the information to competitors or thieves. An attacker can obtain information from the EPC, that is, the manufacturer and product number. This information may be used for corporate espionage purposes by competitors, or for other attacks against other services of the EPC infrastructure. Clearly, the motivation of an attacker to carry out this threat must be rated as high, since attackers can sell their services to competitors, thieves, or any other individual looking for the objects tagged in the organization. The difficulties for performing both eavesdropping and rogue scanning, as shown by the example depicted by Figure 3.2, are solvable. This level of motivation and degree of difficulty lead to a likelihood that is possible. Regarding the potential impact of these threats (e.g., disclosure of information considered by the organization as confidential or trade secrets), it is high, since it may have serious consequences for an organization if an attacker offers the malicious service to competitors or to thieves. These threats are assessed as critical and need to be handled by appropriate countermeasures.

Using the codes eavesdropped or scanned by unauthorized readers, an attacker may successfully clone the tags by conducting, for example, skimming attacks. Indeed, an attacker can simply dump data and responses from a given tag, and program it into a different device. The objective of the attacker for performing the cloning of tags is the possibility for counterfeiting. The attacker may create fake EPC tags that contain data and responses of real tags and sell these counterfeit tags for profit. The forgery of legal tags can be performed without physical access to the organization. We rank the motivation of attackers to carry out the attacks associated with this threat as moderate since they can obtain some financial gain by offering this service to third parties. Current EPC specifications do not include any mechanism for Gen2 compatible readers to verify if they communicate with genuine or fraudulent tags. We thus rate the difficulties associated to this threat as solvable. This level of motivation and degree of difficulty lead to a likelihood that is possible. Regarding the potential impact of this threat, it is medium and thus the threat is assessed as major.

The lack of a strong authentication process in Gen2 tags also has consequences to the privacy of tagged object bearers. Indeed, interrogations of Gen2 tags give attackers unique opportunities for the collection of personal information (and without the consent of the bearer). This can have serious consequences, such as location tracking or surveillance of the object bearers. An attacker can distinguish any given tag by just taking into account the EPC number. Following a reasoning similar to the one used for the cloning threat leads to ranking the risk of the location tracking threat as major. This threat, as well as the cloning threat, must be handled by appropriate countermeasures.

3.3.2 Integrity and Availability Threats

Gen2 tags are required to be writable [3]. They must also implement an access control routine, based on the use of 32-bit passwords, to protect the tags from unauthorized activation of the writing process. Other operations, also protected by 32-bit passwords, can be used in order to permanently lock or disable this operation. Although the writable feature of Gen2 tags is very interesting, it is also one of the least exploited features in current EPC scenarios (due probably to the lack of a strong authentication process, as reported in the previous section). Writable tags are hence locked in most of today's EPC applications. This option will, however, be extremely important in future EPC applications, especially on those self-organizing-based scenarios, where the addition of complementary information into the memory of the tags will require the unlocking of the writing process (e.g., to store routing parameters, locations, or time stamps). It is therefore important to analyze the risk of a tampering attack to the data stored by Gen2 tags, if they can be accessed in write mode from a wireless channel that does not guarantee strong authentication. Figure 3.3 presents a simplified description of protocol steps for requesting and accessing the writing process that modifies the memory of a Gen2 tag. We assume that a select operation has been completed, in order to single out a specific tag from the population of tags. It is also assumed that an inventory query has been completed and that the reader has a valid RN16 identifier (*cf.* Figure 3.2, Steps 2 and 3) to communicate and request further operations from the tag. Using this random sequence (*cf.* Figure 3.2, Step 5), the reader requests a new descriptor (denoted as *Handle* in the following steps). This descriptor is a new random sequence of 16 bits that is used by the reader and tag. Indeed, any command requested by the reader must include this random sequence as a parameter in the command. All the acknowledgments sent by the tag to the reader must also include this random sequence. Once the reader obtains the *Handle* descriptor in Step 6, it acknowledges by sending it back to the tag as a parameter of its query (*cf.* Step 7). To request the execution of the writing process, the reader needs first to be granted access by supplying the 32-bit password that protects the

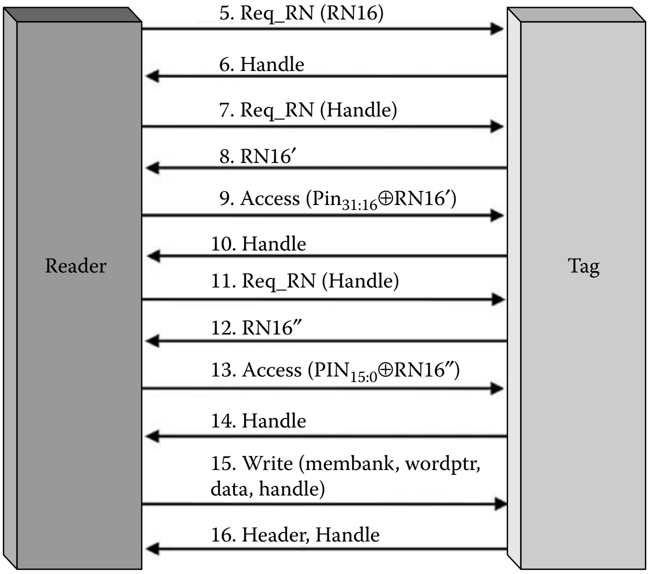

Figure 3.3 Writing protocol of a Gen2 tag.

writing routine. This password is actually composed of two 16-bit sequences, denoted in Figure 3.3 as $\text{PIN}_{31:16}$ and $\text{PIN}_{15:0}$. To protect the communication of the password, the reader obtains in Steps 8 and 12, two random sequences of 16 bits, denoted in as *RN16′* and *RN16″*. These two random sequences *RN16′* and *RN16″* are used by the reader to blind the communication of the password toward the tag. In Step 9, the reader blinds the first 16 bits of the password by applying an XOR operation (denoted by the symbol $\oplus$ in Figure 3.3) with the sequence *RN16′*. It sends the result to the tag, which acknowledges the reception in Step 10. Similarly, the reader blinds the remaining 16 bits of the password by applying an XOR operation with the sequence *RN16″*, and sends the result to the tag in Step 13. The tag acknowledges the reception in Step 14 by sending a new *Handle* to the reader. By using the latter, the reader requests the writing operation in Step 15, which is executed and acknowledged by the tag in Step 16.

An attacker can find the 32-bit password that protects the writing routine. It suffices to intercept sequences RN16′ and RN16″, in Steps 8 and 12, and to apply the XOR operation to the contents of Steps 9 and 13. Other techniques to retrieve similar passwords have also been reported in the literature. For example, in [7] the authors present a mechanism to retrieve passwords by simply analyzing the radio signals sent from readers to tags. Although the proof-of-concept implementation of this technique is only available for Gen1 tags [3], the authors state that Gen2 tags are equally vulnerable. The technical difficulties for setting up attacks to retrieve the password are therefore ranked as solvable. The likelihood of the tampering threat is classified as possible. Regarding the impact, it is ranked, because of some extreme scenarios, as high. For example, in the context of a pharmaceutical supply chain, corrupting data in the memory of EPC tags can be very dangerous: the supply of medicines with wrong information, or delivered to the wrong patients, can lead to situations where a sick person could take the wrong drugs. In these circumstances, the combination of likelihood and impact of the tampering threat lead to critical risk. The threat needs therefore to be addressed by appropriate countermeasures.

Let us note that these attacks enabled by retrieving the passwords, that protect both writing and self-destruction routines of Gen2 tags [3], can be used as models to analyze the risk of threats like destruction of data or denial of service [8]. Tag information can also be destroyed by devices that send strong electromagnetic pulses. Devices, such as the RFID-zapper [9], have been presented in the literature. We can also include here denial of service attacks consisting of jamming channels or flooding channels between tags and readers by sending a large number of requests and responses. For performing a jamming attack, the attacker uses powerful transmitters to generate noise in the range of frequencies used by readers and tags. In any case, the technical difficulties are ranked as solvable. The motivation of attackers to carry out these threats is rated as moderate, since they can obtain some financial gain by offering their malicious services. The likelihood of these two threats is hence classified as possible. However, since the impact of these threats represents to the victim temporal disruption of its operations rather than great financial losses, we rate the impact as medium, and so the risk of these two threats as major.

The final threat analyzed in this section, related to attacks to the integrity and availability of the back-end servers connected to RFID readers, was initially reported in [10]. Rieback et al. uncover the possibility of using malware to attack back-end databases. Their approach classifies such malware into three categories: (1) exploits, (2) worms, and (3) viruses. The exploits are attacks carried out within the information stored into RFID tags. They target the security of middleware services connecting readers to back-end databases. Worms and viruses are attacks that spread themselves over new RFID tags by using network connections (in the case of worms) or connectionless self-replication strategies (in the case of viruses). The malware reported in [10] exploits the trust relationship between back-end databases and the information sent by readers—obtained in turn from malicious tags. Rieback et al. consider that even if there is a very tiny window for storing information into an RFID tag, traditional attacks against information systems (e.g., buffer overflows and SQL injection attacks) might be condensed into a small string of bits harmful enough to break the security of a system. The authors present a proof-of-concept that uses tags carrying an SQL injection attack that compromises the security of the back-end layer of an *ad hoc* RFID setup. The work presented in [10] is interesting and relevant. However, we think that the likelihood for those threats must be rated as unlikely, since no real-world vulnerabilities on the filtering and collection of middleware services specified by EPCglobal can be exploited at the moment. We conclude that even if the impact is potentially serious, due to its unlikely degree of likelihood, the threat is assessed as minor.

3.4 Survey of RFID Security Defences

Research on security countermeasures for RFID technologies can be divided into two categories: (1) hardware-based security primitives for RFID tags, and (2) software protocols using the hardware-based primitives. We review in this section a nonexhaustive list of contributions in both categories.

3.4.1 Hardware-Based Primitives

According to research presented in [4], the cost of passive EPC tags should not exceed 5 cents to successfully enable their deployment on worldwide scale. Of these 5 cents, only 1 or 2 should be used for the manufacturing of the integrated circuit (IC). Another challenge is that the available layout area for the implementation of the IC is in the order of 0.25 mm^2 which, considering current complementary metal–oxide–semiconductor (CMOS) technology, corresponds to a theoretical number of logic gates from 2000 to 4000. Not all the barriers identified in [4] have been removed. Today, the EPC technology is more expensive than what it was originally anticipated—around 10 cents per unit in large quantities. The inclusion of additional features, especially for security purposes, may increase the total end-cost of tags up to 15 cents per unit or more. Although Moore's Law says that the cost of ICs will continue to decrease, cost of analogue devices (i.e., RF front-end of tags) is relatively stable and will remain a constraint [11]. The inclusion of new elements must therefore be clearly justified.

Since EPC tags are powered from the weak energy captured from a reader's electromagnetic waves, their current consumption also needs to be taken into account. This consumption varies according to the operation that is being performed [12] (e.g., responding to a query or writing data into the memory) and other parameters such as the transmission rate, response delay, and memory technology. Most of the operations performed within the modern EPC tags consume about 5–10 μamps—although some special operations, such as write accesses, may consume more. The current consumption of new security primitives must be within this range to allow low-cost tag production. They must also work at the data rate of EPC applications. For example, some supply chain applications demand an average reading speed of about 200 tags/s. This leads to a data transmission rate from tag to reader, of about 640 kbps; and from reader to tag of about 120 kbps. Delays associated to new security mechanisms (e.g., time to perform encryption or random number generation) may also affect the global performance. Delays must hence be taken into account and minimized. We refer the reader to [11] for a more detailed description of aspects that must be considered during the design of new primitives.

Several security proposals aim at including cryptographic primitives on low-cost EPC tags. However, not all the proposals meet the aforementioned constraints or guarantee secure designs. Existing implementations of one-way hash functions, such as MD4, MD5, and SHA-128/SHA-256, exceed cost constraints due to the required number of gates—from 7000 to over 10,000 logic gates according to [13]. The use of cellular automata (CA) theory for the implementation of one-way functions [14] and encryption engines [15]—typically built upon feedback shift registers with a much lower number of gates—has been investigated for the implementation of cryptographic primitives on low-cost RFID tags. However, it has been proved that these implementations are insecure [16,17]. Similarly, the use of linear feedback shift registers and nonlinear feedback shift registers (LFSR & NLFSR) as underlying mechanisms for the implementation of low-cost one-way hash functions and pseudorandom number generators (PRNG)—without appropriate measures that increase the cost—also lead to insecure implementations [18,19]. Light-weight hardware implementations of standard block ciphers to implement one-way hash functions have been discussed. The use of elliptic curve cryptosystems (ECC) [19] for the implementation of primitives for RFID tags has been discussed in [20]. Its use of small key sizes is seen as very promising for providing an adequate level of computational security at a relatively low cost [11]. An ECC implementation for low-cost RFID tags can be found in [21]. In [22], on the other hand, Feldhofer et al. present a 128-bit implementation of the advanced encryption standard (AES) [23] on an IC of about 3500 gates with a current consumption of less than 9 μamps at a frequency of 100 kHz. The encryption of each block of 128 bits requires about 1000 clock cycles. Although, it considerably simplifies previous implementations of the AES, for example, proposals presented in [24,25] that require between 10,000 and over 100,000 gates, respectively, the design is still considered too complex for basic EPC setups [26].

More suitable encryption engine implementations can be found in [27,28]. The first reference presents the implementation of the tiny encryption algorithm (TEA) [29]. It is implemented on an IC of about 3000 gates with a current consumption of about 7 μamps. It fits the timing requirements of basic EPC setups where hundreds of tags must simultaneously be accessed by the same reader. For meeting the constraints, the implementation relies on very simple operators such as XOR, ADD, and SHIFT. The authors of TEA [29] claim that, despite its simplicity and ease of implementation, the complexity of the algorithm is equivalent to data encryption standard (DES) [19]. Variants of the basic algorithm, such as eXtended TEA (XTEA), are however necessary for implementing one-way hash functions. Mace et al. discuss in [28] some of the vulnerabilities of TEA, such as linear and differential cryptanalysis attacks, and present scalable encryption algorithm (SEA). Given the relatively recent invention of these algorithms, their strength is not clear [11].

There are other hardware-based security enhancements for RFID technologies not relying on the implementation of cryptographic primitives. Many signal- and power-based defenses, such as shielding of tags, use of noise and third-party blocker devices have been surveyed in [26]. The use of distance measurements to detect rogue readers has been discussed. In [30], for example, Fishkin et al. propose the inclusion of low-cost circuitry on tags to use the signal-to-noise ratio of readers as a metric for trust. In [31], a similar assumption is used in order to claim that a reader can be authorized to read a tag contents according to its physical distance. The use of trust [32] and trusted computing [33] with similar purposes has also been discussed. For example, Molnar et al. describe in [33] a mechanism consisting of trusted platform modules (TPMs) to enforce privacy policies within the RFID tags. A trusted entity called trusted center (TC) decides whether readers are allowed or not to access tags. Finally, the use of radio fingerprinting [34,35] to detect characteristic properties of transmitted signals and design authentication procedures has been investigated. The authors in [11] consider, however, that this technique is difficult to develop on RFID applications and that the benefits of using it, with respect to performance, cost, and required implementation surface on tags, are unclear. Avoine and Oechslin also debate in [36] the prevention of the traceability via radio fingerprinting.

They conclude that obtaining radio fingerprint of tags is expensive and difficult. The myriad of tags in circulation in future RFID scenarios makes impracticable the individual distinction of them.

Physically unclonable functions (PUFs) and physical obfuscated keys (POKs) are promising for the implementation of new security primitives in low-cost EPC tags. They can be used to handle the authentication threat, as well as the cloning and location tracking threats. Half way between cryptography-based enhancements and physical protection defenses, the ideas behind PUFs and POKs originated in [37] with the conception of optical mechanisms for the construction of physical one-way functions (POWFs). Their use to securely store unique secret keys, in the form of fabrication variations, was proposed as a silicon prototype in [38,39]. These ideas were later improved in [40]. A coating PUF proposed in [41] is implemented with less than 1000 gates. These designs exploit the random variations in delays of wires and logic gates of an IC. For example, the silicon PUF presented in [39] receives input data, as a challenge, and launches a race condition within the IC: two signals propagate along different paths and are compared to determine which one comes first. To decide which signal comes first, a controller, implemented as a latch, produces a binary value. Holcomb et al. [42] propose using the SRAM based on CMOS circuitry to generate physical fingerprints. The key idea is the use of SRAM start-up values as origin of randomness. The use of 256 bytes of Static Random Access Memory (SRAM) can yield 100 bits of true randomness each time that the memory is powered up. While sound in theory, this technique has as important drawback the limitation of memory space of current low-cost tags. The implementation of PUF-based circuits seems to have clear advantages at a cost of less than 1000 logic gates [41]. This technology provides a cost effective and reliable solution that meets the constraints and requirements. Drawbacks, such as the effects of environmental conditions and of power supply voltage [43], must be taken into account. The difficulty of successfully modeling the circuits and their reliability have also raised some concerns. Bolotnyy and Robins [44] address some of these issues. Some attacks on PUF- and POK-based protocols are outlined in [40]. The execution and reinterpretation of existing protocols via new PUF and POK designs—essentially the challenge–response protocols—are outlined in the sequel.

3.4.2 Software Protocols

We review algorithmic solutions and software protocols for handling the threats uncovered in Section 3.3. The solutions rely on the implementation and use of hardware-based primitives discussed in Section 4.1.

Message Authentication Code (MAC)-based security protocols for wireless applications is a typical solution discussed in the literature (e.g., [44–46]). In [45], Takaragi et al. present a very simple MAC-based approach. It uses a static unrewritable 128-bit identifier stored, at manufacturing time, in every tag. This static identifier is not modifiable once the shipment is made. To build up this identifier, the manufacturer uses a unique secret key for each tag and a keyed hash function that accepts as input the secret key and a specific message. All this information (i.e., secret key, hash function, and specific message) is communicated by the manufacturer to the client. By sharing this information among readers and tags, integrity and authenticity of exchanged messages is verified. It therefore reduces the risk of threats to authenticity and integrity by increasing the technical difficulties of performing attacks. However, due to the use of static identifiers embedded in the tags at manufacturing time, the location tracking issue is not solved. Moreover, brute force attacks can break the secrets shared between readers and tags.

The use of public key cryptography and digital signatures is discussed in [47]. The authors address the protection of banknotes embedding the RFID tags. Their approach includes the possibility of deploying cryptographic protocols in RFID applications, but avoids the need to embed cryptographic primitives within the tags. The scheme consists of a public-key cryptosystem used by a central bank aiming to avoid banknote forgery and a law enforcement agency that aims at tracking banknotes. Both authorities, that is, central bank and law enforcement agency, hold an independent pair of public and private keys associated to each banknote. The central bank authority assigns a unique serial number to each banknote. The central bank authority, using its private key, signs the unique serial number. The unique serial number of the banknote and its corresponding digital signature are printed on the banknote as optical data. In addition, the law enforcement agency encrypts with its public key the digital signature, unique serial number, and a random number. The resulting ciphertext is stored into a memory cell of the RFID tag. This memory cell is keyed-protected. The tag only grants write access to this memory cell if it receives an access key derived from the optical data. The random number used to create the ciphertext is also stored into a separated memory cell of the tag. This second memory cell is also keyed-protected. The tag only grants read or write access to this memory cell if it receives an access key derived from the optical data.

Now, a merchant that receives a banknote must verify first the digital signature, printed in the banknote as optical data, using the public key of the central bank. Second, the merchant must also verify the validity of the ciphertext stored in the banknote's tag. To do so, the merchant encrypts the digital signature, serial number, and random number stored in the tag's memory, using the public key of the law enforcement agency and the optical data. If one of these two verification processes fails, the authorities must be warned. To avoid using the same ciphertext on every interaction, Juels and Pappu propose the use of a reencryption process that can be performed by the merchant without the necessity of accessing the private keys of the law enforcement authority. Indeed, based on the algebraic properties of the El Gamal cryptosystem [19], the initial ciphertext can be transformed into a new unlinkable ciphertext only using the public key of the law enforcement authority [26]. This reencryption process is performed outside the tags. Integrity issues of this approach are discussed and fixed in [48]. However, the whole process and requirements for implementing the approach in [47,48] are too complex and expensive for use in EPC supply chain applications.

Mutual authentication protocols among tags and readers are discussed in [49,50]. The work presented by Kinosita et al. in [49] consists of an anonymous ID scheme, in which a tag contains only a pseudonym that is periodically rewritten. Pseudonyms are used instead of real identifiers (e.g., instead of the EPC codes). Similarly, the approach of Juels entitled minimalist cryptography for low-cost RFID tags [50] suggests a very lightweight protocol for mutual authentication between tags and readers based on one-time authenticators. Both solutions rely on the use of pseudonyms and keys stored within tags and back-end servers. Each tag contains a small collection of pseudonyms, according to the available memory of the tag. A throttling process is used to rotate the pseudonyms. Each time the tag is interrogated by a reader, a different pseudonym is used in the response. Authorized readers have access to the complete list of pseudonyms set for each tag and can correlate the responses they receive. Without the knowledge of this list, unauthorized readers are unable to infer any information about the several occurrences of the same tag. The process also forces tags to slow down their data transmissions when queried too frequently, as a defense to potential brute-force attacks. The memory space on current low-cost tags is the main limitation of this approach. Although enhancements can be used to update the list of pseudonyms, communication costs, and integrity threats will remain as main drawbacks.

The use of hash-lock schemes for addressing authentication issues is another possibility. A design can be found in [51]. Weis et al. propose a way to lock tags without storing access keys in them. Only hashes of keys must be known by the tags. Keys must be also stored on back-end servers and be accessible by authorized readers. Most authentication threats are therefore mitigated by locking tags. Cloning and tracking threats are handled by avoiding the use of real identifiers once tags are locked. In [52], Henrici and Müller extend the hash-lock scheme and address some weaknesses in [51] to increase traceability and location resistance. A similar hash-based protocol is presented in [53] in order to deal with those limitations by using time stamps. Other similar hash-based protocols for handling authentication threats can be found in [54–56]. All these protocols rely on synchronized secrets residing in the tags and back-end servers. They require a one-way hash function implemented within the tags. The requirement of reliable hash primitives implemented at the tag level is the main drawback. Workload on back-end servers is also considerably high and can make difficult the deployment in real-world EPC supply chain applications. The Yet Another Trivial RFID Authentication Protocol (YATRAP) protocol presented in [57] reduces the cost of computation by combining precomputed hash-tables for tag verification processes, use of time stamps, and generation of pseudorandom numbers. The protocol is, however, vulnerable to availability attacks when temporal de-synchronizations between tags and readers occur. Some limitations are addressed in [58]. Chatmon et al. define new protocols for anonymous authentication. These improvements notably increase the degree of workload on servers and are highly complex for use in supply chain applications.

3.5 Future Directions for Research

Algorithmic solutions avoiding the execution of on-tag cryptographic processes seems to lead the future of research in RFID security. In this sense, a secret-sharing scheme is presented by Juels et al. in [59] as a defense against the authenticity threats in EPC supply chain applications. Two different models are discussed: dispersion of secrets across space and dispersion of secrets across time. Both models are based on a secret-sharing strategy, where a secret used to encrypt EPCs is split in multiple shares and distributed among multiple parties. In order to obtain the EPC of a tag, a party must collect a minimum number of shares distributed among all the other parties. Authentication is therefore achieved though the dispersion of secrets. The dispersion helps to improve the authentication process between readers and tags, as tags move through a supply chain. Assuming that a given number of shares is necessary for readers to obtain the EPCs assigned to a pallet, for example, a situation where the number of shares obtained by readers is not sufficient to reach the threshold protects the tags from unauthorized scanning (i.e., unauthorized readers that cannot obtain the sufficient number of shares cannot obtain the EPCs either). The approach can be implemented on EPC Gen2 tags without requiring any change to the current tag specification. A limitation is the amount of tag memory space required for storing the shares. However, the shrinking of shares can allow the application of the scheme to current EPC tags. A more important problem is that the location tracking threat is not addressed. Indeed, the shares used in the approach are static. This problem must be solved before deployment of the scheme.

Challenge–response protocols for low-cost EPC tags using physical unclonable functions (PUFs) and POKs have recently gained importance. An approach presented in [60], based on PUFs proposed in [38,39], consists of a challenge–response scheme that probabilistically ensures unique identification of RFID tags. A back-end system must learn challenge–response pairs for each PUF/tag. It then uses these challenges (hundreds of them) at a time, to identify and authenticate tags. Unique identification of tags is only ensured probabilistically. The exposition of tag identifiers to eavesdroppers and lack of randomness in tag responses, make the approach vulnerable to the location tracking threat. Moreover, the great number of challenges that are necessary in the identification process increases the tag response delay and power consumption. Hence, this approach might not meet the constraints and requirements mentioned in Section 4.1. An alternative approach is presented in [61]. Tuyls and Batina discuss an off-line PUF-based mechanism for verifying the authenticity of tags through the PUF technology presented in [41]. Similar to the results presented in [50,62], where readers and tags define *ad hoc* secrets, the PUF-based approach uses the internal physical structure of tags to generate unique keys. A key extraction algorithm from noisy binary data is presented in [61]. The usage of PUF-based keys simplifies the process of verifying tag authenticity. The combination of unique keys generated onboard together with the use of signatures avoid leaking of a single identifier and increases the technical difficulties for an attacker to carry out the location tracking threat. The main drawback is the need of large storage space and reliable searching processes on back-end servers to link readers with PUF/tag identifiers. The use of public key and digital signatures, based on Elliptic Curve Cryptography (ECC), is another important constraint. Following the trend of combining PUFs together with traditional cryptographic primitives and encryption engines, a modification of the tree-based hash protocols proposed in [63] is presented in [64]. Using the notion of POKs introduced in [38] (i.e., application of a fixed hard-wired challenge to the PUF to obtain a unique secret), the authors guarantee the existence of internal keys in basic tree-based hash protocols, now physically obfuscated. They cannot be cloned by unauthorized parties. The use of an AES engine, such as the one presented in [23], is proposed. On the other hand, Bolotnyy and Robins present in [44] a complete set of adapted MAC protocols, based on PUFs, trying to simplify the challenge–response communication scheme of previous proposals and to eliminate requirement of traditional cryptographic primitives. Each tag generates multiple identifiers based on embedded PUFs. Their approach only addresses static identification. It is vulnerable to the location tracking threat identified in Section 3.3. It does not solve the requirement of huge lists of challenge–response pairs for each PUF/tag that must be stored on back-end servers connected to the readers. Indeed, each given pair is of single use to prevent replay attacks.

3.6 Conclusions

At the beginning of this chapter we presented an analysis of threats to the RFID system of the EPC architecture. We identified different groups of threats that we consider relevant for further research. We ranked the eavesdropping, rogue scanning, and tampering threats as critical; and cloning, tracking, and denial of service threats as major. We concluded that they must be handled by appropriate countermeasures. We then surveyed in the sequel practical and theoretical security defenses that can be useful to reduce the risk of the identified threats. We looked at the different defenses from two different research perspectives. On the one hand, we surveyed research on hardware-based defenses that aim at providing additional security primitives on tags such as one-way hash functions, encryption engines, and physically unclonable functions (PUFs). On the other hand, we surveyed research on software protocols that make use of these new on-tag primitives for designing and implementing reliable algorithms for dealing with security and privacy issues. We have seen that the implementation of well-known cryptographic primitives is possible and allows the design of software protocols to reduce the risk of threats ranked as critical or major. The cost and requirements of these proposals are the main difficulties. Indeed, they are too expensive for their deployment in supply chain scenarios based on the EPC technology. We have also surveyed the combination of cryptographic primitives together with the use of PUFs for the design of cost-effective solutions. These solutions present drawbacks, such as the sensitivity of PUFs to physical noise and the difficulty to model and analyze them. They are, however, promising solutions that successfully meet the implementation constraints and requirements for handling the set of threats reported in our work. For the second group, we conclude that the avoidance of on-tag cryptographic processes on current algorithmic solutions seems to lead the future directions of research in RFID security. In this sense, the use of secret-sharing schemes present clear advantages for the management of keys in the design of authentication protocols and to deal with privacy issues. The main drawback is the use of static shares, limiting the use of this approach for addressing the location tracking threat.

Acknowledgments

The authors graciously acknowledge the financial support received from the following organizations: Natural Sciences and Engineering Research Council of Canada (NSERC), Mathematics of Information Technology and Complex Systems (MITACS), Spanish Ministry of Science and Education (TSI2007-65406-C03-03 "E-AEGIS" and CONSOLIDER CSD2007-00004 "ARES" grants), La Caixa (Canada awards), and the IRF project LOCHNESS.

Terminologies

Advanced Encryption Standard (AES)—A block cipher encryption standard, sponsored by the National Institute of Standards and Technology (NIST), for protecting data.

Countermeasure—A defense mechanism designed to mitigate the risk of a threat.

EPC Network—A service-oriented architecture defined by EPCglobal Inc. that proposes the integration of RFID and Internet technologies to enable automatic identification and sharing of item data in supply chain applications.

Electronic Product Code (EPC)—Group of coding schemes defined by EPCglobal Inc.

Elliptic Curve Cryptography (ECC)—A public key cryptosystem based on the algebraic structure of elliptic curves over finite fields.

European Telecommunications Standards Institute (ETSI)—Independent noncommercial organization that produces telecommunications standards to be used in Europe and beyond.

EPC Number—A tag data format compatible with the family of coding schemes proposed by EPCglobal Inc. It typically contains: a header, pointing out the family code that is being used; a manufacturer code; an object class; and a serial number.

EPCGLOBAL Inc.—Joint venture between GS1 (Global Standards One, formerly known as EAN International) and GS1 US™ (formerly the Uniform Code Council, Inc.) created to commercialize the EPC technology.

Linear and Nonlinear Feedback Shift Registers (LFSR & NLFSR)—A digital circuit composed of an n-bit shift register and a feedback function that generates pseudorandom sequences.

Passive Tag—RFID component attached to an item. It contains information about the item. Since it does not have its own power source, it provides the information by backscattering a reader's signal.

Physically Unclonable Function (PUF)—Hardware-based function embedded in a physical structure, that is easy to evaluate but hard to reproduce.

Physical Obfuscated Keys (POK)—Hardware-based function for implementing secrets on digital devices using a physically unclonable function.

Pseudorandom Number Generator (PRNG)—Algorithmic solution to generate deterministic sequences of pseudorandom numbers.

Reader—RFID component that requests and receives information from tagged items.

Tag Identification (TID)—Memory bank or identifier that uniquely identifies an RF tag.

Threat Analysis—Determination and classification, in terms of importance, of threats targeting the security of a system.

Tiny Encryption Algorithm (TEA)—A minimalist block cipher encryption algorithm for protecting data.

Trusted Platform Modules (TPMs)—Hardware-based cryptographic mechanism installed on the motherboard of a digital device (i.e., a personal computer) to enforce security protection and trustworthiness.

Scalable Encryption Algorithm (SEA)—A block cipher encryption algorithm designed to be used on embedded applications such as microcontrollers.

Questions and Sample Answers

1. What is Passive RFID?
   Passive radio frequency identification (RFID) is a wireless communication technology that allows the automatic identification of objects, animals, and persons through radio waves.

2. How a security attack can succeed against EPC network architecture?
   Security attacks can target the different services of the EPC network architecture. They may succeed if weaknesses within the underlying technologies are not handled properly. The exchange of information between EPC tags and readers, for example, is carried out via wireless channels that do not possess basic security attributes such as authenticity, integrity, and availability. This situation allows attackers to misuse the RFID service of an EPC network and perform unauthorized activities such as eavesdropping, rogue scanning, cloning, location tracking, and tampering of data.

3. What is the major challenge for implementing new security features in EPC tags?
   The implementation of new security features in EPC tags faces several challenges. The main one is the cost.

4. What is a threat?
   A threat is the objective of an attacker to violate security properties of a target system, such as authenticity, integrity, and availability.

5. What is an attacker?
   An attacker is an agent that is exploiting a vulnerability of the targeted system to carry out the threat. The exploitation of the vulnerability is defined as the attack.

6. Mention two major areas for research on security countermeasures for RFID technologies.
   Research on security countermeasures for RFID technologies can be divided into two categories: (1) hardware-based security primitives for RFID tags and (2) software protocols using the hardware-based primitives.

7. What does ETSI stand for?
   European Telecommunications Standards Institute (ETSI)

8. When is a threat ranked as minor?
   A threat is ranked as minor when it is unlikely to happen or when its potential impact is low. Minor risk threats typically require no countermeasures.

9. Draw the diagram of Inventory Protocol of a Gen2 Tag.

10. Define: Forward and Backward eavesdropping.
    We define forward eavesdropping as the passive collection of queries and commands sent from readers to tags; and backward eavesdropping as the passive collection of responses sent from tags to readers.

Author's Biography

Joaquin Garcia-Alfaro is a lecturer professor at the Computer Science and Multimedia Studies of the Open University of Catalonia, Spain. He obtained a Bachelor, a Master's, and a PhD (in Computer Science) from Autonomous University of Barcelona (Spain) and TELECOM Bretagne (France) in 2006. From 2007 to 2009, he was postdoctoral fellow at the Computer Science Department of Carleton University, Canada. Since 2009, he also collaborates as an associate researcher at TELECOM Bretagne, France. His research interests include a wide range of network security problems, with an emphasis on the management of security policies, analysis of vulnerabilities, and enforcement of countermeasures.

Michel Barbeau is a professor of computer science. He obtained a Bachelor, a Master's, and a PhD, in computer science, from Université de Sherbrooke, Canada (1985), for undergraduate studies, and Université de Montréal, Canada (1987 and 1991), for graduate studies. From 1991 to 1999, he was a professor at Université de Sherbrooke. During the academic year 1998–1999, he was a visiting researcher at the University of Aizu, Japan. Since 2000, he works at Carleton University, Canada. Wireless communications has been his main research interest. He focuses his efforts on wireless security, vehicular communications, wireless access network management, *ad hoc* networks, and RFID.

Evangelos Kranakis is a professor of computer science. He obtained a BSc (in Mathematics) from the University of Athens (1973) and a PhD (in Mathematical Logic) from the University of Minnesota, Minnesota (1980). From 1980 to 1982 he was at the Mathematics Department of Purdue University, West Lafayette, Indiana, and from 1982 to 1983 at the Mathematisches Institut of the University of Heidelberg, Germany, between 1983 and 1985 he served at the Computer Science Department of Yale University, New Haven, Connecticut, from August to December of 1985 at the Computer Science Department of the Universiteit van Amsterdam, the Netherlands and from 1986 to 1991 at the Centrum voor Wiskunde en Informatica (CWI) in Amsterdam, the Netherlands. Since 1991, he works at Carleton University, Canada. He was director of the School of Computer Science from 1994 to 2000. He received the Carleton Research Achievement award in 2000. He became Carleton University Chancellor's Professor in 2006. He has published in the analysis of algorithms, bioinformatics, communication and data (*ad hoc* and wireless) networks, computational and combinatorial geometry, distributed computing, and network security.

References

1. R. Want, RFID explained: A primer on radio frequency identification technologies, *Synthesis Lectures on Mobile and Pervasive Computing*, Num. 1, 2006, Morgan & Claypool Publishers.
2. G. Roussos, S. Duri, and W. Thompson. RFID meets the internet, *IEEE Internet Computing. Special Issue on RFID*, 13, 105–114, 2009.
3. EPCglobal Overview & Standards, Available from: http://www.epcglobalinc.org/standards/
4. S. E. Sarma, Toward the 5 cent tag, White Paper, November 2001, Auto-ID Center.

5. J. Sounderpandian, R. V. Boppana, S. Chalasani, and A. M. Madni, Models for cost–benefit analysis of RFID implementations in retail stores, *Systems Journal, IEEE*, 1(2), 105–114, 2007.
6. ETSI, Methods and protocols for security; part 1: Threat analysis. ETSI-ts 102 165-1 v4.1.1, 2003.
7. Y. Oren, Remote power analysis of RFID tags, *Cryptology ePrint Archive*, Report 2007/330, IACR, 2007.
8. D. Han, T. Takagi, H. Kim, and K. Chung, New security problem in RFID systems tag killing, *Lecture Notes in Computer Science*, M. L. Gavrilova, O. Gervasi, V. Kumar, C. J. K. Tan, D. Taniar, A. Lagana, Y. Mun, and H. Choo, eds. vol. 3982. Springer, 2006, pp. 375–384.
9. Minime and Mahajivana, RFID Zapper, *22nd Chaos Communication Congress (22C3)*, December 2005.
10. M. Rieback, B. Crispo, and A. Tanenbaum, Is your cat infected with a computer virus?, in *Pervasive Computing and Communications, IEEE*. Pisa, Italy: IEEE Computer Society Press, March 2006, pp. 13–17.
11. P. Cole and D. Ranasinghe, eds. *Networked RFID Systems and Lightweight Cryptography—Raising Barriers to Product Counterfeiting*, 1st ed. Springer, 2008.
12. T. Lohmann, M. Schneider, and C. Ruland, Analysis of power constraints for cryptographic algorithms in mid-cost RFID tags, in *Smart Card Research and Advanced Applications, 7th IFIP WG 8.8/11.2 International Conference (CARDIS 2006), Lecture Notes in Computer Science*, vol. 3928, Berlin/Heidelberg: Springer, April 2006, pp. 278–288.
13. M. Feldhofer and C. Rechberger, A case against currently used hash functions in RFID protocols, *Workshop on RFID Security – RFIDSec 06*, Ecrypt, Graz, Austria, July 2006, pp. 372–381.
14. S. Wolfram, Cryptography with cellular automata, in *Advances in Cryptology, CRYPTO 85, Lecture Notes in Computer Sciences*, vol. 218, New York, NY, USA: Springer-Verlag New York, Inc., 1986, pp. 429–432.
15. S. Sen, C. Shaw, D. R. Chowdhuri, N. Ganguly, and P. P. Chaudhuri, Cellular automata based cryptosystem (CAC), in *4th International Conference on Information and Communications Security (ICICS'02)*. London, UK: Springer-Verlag, 2002, pp. 303–314.
16. P. Bardell, Analysis of cellular automata used as pseudorandom pattern generators, in *International Test Conference*, 1990, Washington DC, pp. 762–768.
17. S. R. Blackburn, S. Murphy, and K. G. Paterson, Comments on theory and applications of cellular automata in cryptography, *IEEE Transactions on Software Engineering*, 23(9), 637–638, 1997.
18. C. Meyer and W. Tuchman, Pseudo-random codes can be cracked, *Electronic Design*, 23, 1972.
19. A. Menezes, P. Van Oorschot, and S. Vanstone, *Handbook of Applied Cryptography*. CRC Press, USA, 1997.
20. J. Wolkerstorfer, Is elliptic-curve cryptography suitable to secure RFID tags? *Workshop on RFID and Lightweight Crypto*, Ecrypt, Graz, July 2005.
21. L. Batina, J. Guajardo, T. Kerins, N. Mentens, P. Tuyls, and I. Verbauwhede, An elliptic curve processor suitable for RFID-tags, *Cryptology ePrint Archive*, Report 2006/227, IACR, 2006.
22. M. Feldhofer, S. Dominikus, and J. Wolkerstorfer, Strong authentication for RFID systems using the AES algorithm, in *Workshop on Cryptographic Hardware and Embedded Systems, CHES 2004, Lecture Notes in Computer Science*, vol. 3156, IACR. Boston, MA, USA: Springer, August 2004, pp. 357–370.
23. J. Daemen and V. Rijmen, *The Design of Rijndael: AES—the Advanced Encryption Standard*. Springer, 2002.
24. S. Mangard, M. Aigner, and S. Dominikus, A highly regular and scalable AES hardware architecture, *IEEE Transactions on Computers*, 52(4), 483–491, 2003.
25. I. Verbauwhede, P. Schaumont, and H. Kuo, Design and performance testing of a 2.29-GB/s Rijndael processor, *IEEE Journal of Solid-State Circuits*, 38(3), 569–572, 2003.
26. A. Juels, RFID security and privacy: A research survey, *IEEE Journal on Selected Areas in Communication*, 24(2), 381–394, 2006.
27. P. Israsena, Securing ubiquitous and low-cost RFID using tiny encryption algorithm, in *International Symposium on Wireless Pervasive Computing, IEEE. Phuket*, Thailand: IEEE Press, January 2006, pp. 1–4.
28. F. Mace, F.-X. Standaert, and J.-J. Quisquater, Asic implementations of the block cipher sea for constrained applications, in *Conference on RFID Security*, Malaga, Spain, July 2007, pp. 103–114.
29. D. J. Wheeler and R. M. Needham, TEA, a tiny encryption algorithm, in *Fast Software Encryption: Second International Workshop (FSE 1994)*, Leuven, Belgium, December, *Lecture Notes in Computer Science*, vol. 1008. Springer, Berlin/Heidelberg, 1995, pp. 363–366.

30. K. Fishkin, S. Roy, and B. Jiang, Some methods for privacy in RFID communication, in *European Workshop on Security in Ad-hoc and Sensor Networks, ESAS 2004, Lecture Notes in Computer Science*, vol. 3313. Heidelberg, Germany: Springer-Verlag, August 2005, pp. 42–53.
31. G. Hancke, Noisy carrier modulation for HF RFID, in *First International EURASIP Workshop on RFID Technology*, Vienna, Austria, September 2007, pp. 63–66.
32. A. Solanas, J. Domingo-Ferrer, A. Martinez-Balleste, and V. Daza, A distributed architecture for scalable private RFID tag identification, *Computer Networks*, 51(9), 2268–2279, 2007.
33. D. Molnar, A. Soppera, and D. Wagner, A scalable, delegatable pseudonym protocol enabling ownership transfer of RFID tags, in B. Preneel and S. Tavares, eds. *Selected Areas in Cryptography, SAC 2005, Lecture Notes in Computer Science*, vol. 3897. Kingston, Canada: Springer, August 2005, pp. 276–290.
34. J. Hall, M. Barbeau, E. and Kranakis, Enhancing intrusion detection in wireless networks using radio frequency fingerprinting, *Communications, Internet, and Information Technology*, 2004, 201–206.
35. J. Hall, Detection of rogue devices in Wireless Networks, PhD dissertation, Carleton University, 2006.
36. G. Avoine and P. Oechslin, RFID traceability: A multilayer problem, in *Financial Cryptography 2005, Lecture Notes in Computer Science*, vol. 3570, IFCA. Roseau, The Commonwealth Of Dominica: Springer-Verlag, February–March 2005, pp. 125–140.
37. R. Pappu, Physical one-way functions, PhD dissertation, MIT, 2001.
38. B. Gassend, Physical random functions, Master's thesis, MIT, 2003.
39. B. Gassend, D. Clarke, M. van Dijk, and S. Devadas, Silicon physical random functions, in *9th ACM Conference on Computer and Communications Security*. New York, NY, USA: ACM, 2002, pp. 148–160.
40. D. Lim, J. Lee, B. Gassend, G. Suh, M. van Dijk, and S. Devadas, Extracting secret keys from integrated circuits, *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 13(10), 1200–1205, 2005.
41. B. Skoric and P. Tuyls, *Secret Key Generation from Classical Physics*, Philips Research Book Series, September 2005.
42. D. Holcomb, W. Burleson, and K. Fu, Initial SRAM state as a fingerprint and source of true random numbers for RFID tags, in *Third International Conference on RFID Security*, RFIDSec 2007, Malaga, Spain, 2007, pp. 31–42.
43. D. Ranasinghe, D. Engels, and P. Cole, Security and privacy solutions for low cost RFID Systems, in *2004 Intelligent Sensors, Sensor Networks & Information Processing Conference*, Melbourne, Australia, 2004, pp. 337–342.
44. L. Bolotnyy and G. Robins, Physically unclonable function-based security and privacy in RFID systems, in *International Conference on Pervasive Computing and Communications—PerCom 2007*, IEEE. New York, USA: IEEE Computer Society Press, March 2007, pp. 211–220.
45. K. Takaragi, M. Usami, R. Imura, R. Itsuki, and T. Satoh, An ultra small individual recognition security chip, *IEEE Micro*, 21(6), 43–49, 2001.
46. A. Willig, M. Kubisch, C. Hoene, and A. Wolisz, Measurements of a wireless link in an industrial environment using an IEEE 802.11-compliant physical layer, *IEEE Transactions on Industrial Electronics*, 49(6), 1265–1282, 2002.
47. A. Juels and R. Pappu, Squealing euros: Privacy protection in RFID-enabled banknotes, in R. N. Wright, ed., *Financial Cryptography 2003, Lecture Notes in Computer Science*, vol. 2742, IFCA. Le Gosier, Guadeloupe, French West Indies: Springer, January 2003, pp. 103–121.
48. X. Zhang and B. King, Integrity improvements to an RFID privacy protection protocol for anti-counterfeiting, in J. Zhou, J. Lopez, R. Deng, and F. Bao, eds. *Information Security Conference, ISC 2005, Lecture Notes in Computer Science*, vol. 3650. Singapore: Springer, September 2005, pp. 474–481.
49. S. Kinosita, F. Hoshino, T. Komuro, A. Fujimura, and M. Ohkubo, Non identifiable anonymous-ID scheme for RFID privacy protection, in *Japanese*. English description as part of http://www.autoidlabs.com/whitepaper/KEI-AUTOID-WH004.pdf, 2003.
50. A. Juels, Minimalist cryptography for low-cost RFID tags, in C. Blundo and S. Cimato, eds. *International Conference on Security in Communication Networks, SCN 2004, Lecture Notes in Computer Science*, vol. 3352. Amalfi, Italia: Springer, September 2004, pp. 149–164.

51. S. Weis, S. Sarma, R. Rivest, and D. Engels, Security and privacy aspects of low-cost radio frequency identification systems, in D. Hutter, G. Müller, W. Stephan, and M. Ullmann, eds., *International Conference on Security in Pervasive Computing, SPC 2003, Lecture Notes in Computer Science*, vol. 2802. Boppard, Germany: Springer, March 2003, pp. 454–469.
52. D. Henrici and P. Müller, Hash-based enhancement of location privacy for radio-frequency identification devices using varying identifiers, in *International Workshop on Pervasive Computing and Communication Security—PerSec 2004*, IEEE Computer Society, March 2004, pp. 149–153.
53. G. Avoine and P. Oechslin, A scalable and provably secure hash based RFID protocol, in *International Workshop on Pervasive Computing and Communication Security—PerSec 2005*, IEEE Computer Society Press, March 2005, Kauai Island, Hawaii, pp. 110–114.
54. S.-M. Lee, Y. J. Hwang, D. H. Lee, and J. I. L. Lim, Efficient authentication for low-cost RFID systems, in O. Gervasi, M. Gavrilova, V. Kumar, A. Lagana'a, H. P. Lee, Y. Mun, D. Taniar, and C. J. K. Tan, eds. *International Conference on Computational Science and its Applications, ICCSA 2005, Lecture Notes in Computer Science*, vol. 3480. Singapore: Springer, May 2005, pp. 619–627.
55. E. Y. Choi, S. M. Lee, and D. H. Lee, Efficient RFID authentication protocol for ubiquitous computing environment, in T. Enokido, L. Yan, B. Xiao, D. Kim, Y. Dai, and L. Yang, eds., *International Workshop on Security in Ubiquitous Computing Systems, SECUBIQ 2005, Lecture Notes in Computer Science*, vol. 3823. Nagasaki, Japan: Springer, December 2005, pp. 945–954.
56. S. Lee, T. Asano, and K. Kim, RFID mutual authentication scheme based on synchronized secret information, in *Symposium on Cryptography and Information Security*, Hiroshima, Japan, January 2006.
57. G. Tsudik, YA-TRAP: Yet another trivial RFID authentication protocol, in *International Conference on Pervasive Computing and Communications, PerCom 2006, IEEE*. Pisa, Italy: IEEE Computer Society Press, March 2006, pp. 640–643.
58. C. Chatmon, T. van Le, and M. Burmester, Secure anonymous RFID authentication protocols, Florida State University, Department of Computer Science, Tallahassee, FL, USA, Technical Report TR-060112, 2006.
59. A. Juels, R. Pappu, and B. Parno, Unidirectional key distribution across time and space with applications to RFID security, in *USENIX Security Symposium*. San Jose, CA: USENIX, July 2008, pp. 75–90.
60. D. Ranasinghe, D. Engels, and P. Cole, Low-cost RFID systems: Confronting security and privacy, in *Auto-ID Labs Research Workshop*, Zurich, Switzerland, September 2004, pp. 54–77.
61. P. Tuyls and L. Batina, RFID-tags for anti-counterfeiting, in *Topics in Cryptology, CT-RSA 2006, The Cryptographers' Track at the RSA Conference 2006, Lecture Notes in Computer Science*, USA: Springer, February 2006, pp. 115–131.
62. A. Juels and S. Weis, Authenticating pervasive devices with human protocols, in *Advances in Cryptology, CRYPTO 2005, Lecture Notes in Computer Science*, vol. 3126, IACR. Santa Barbara, CA, USA: Springer, August 2005, pp. 293–308.
63. D. Molnar and D. Wagner, Privacy and security in library RFID: Issues, practices, and architectures, in *Conference on Computer and Communications Security—ACM CCS*, USA: ACM Press, October 2004, pp. 210–219.
64. J. Bringer, H. Chabanne, and T. Icart, Improved privacy of the tree-based hash protocols using physically unclonable function, *Cryptology ePrint Archive*, Report 2007/294, 2007.