Securing Low-cost RFID Systems: A Research Survey

Qi Chai
Department of Electrical and Computer Engineering
University of Waterloo
Waterloo, Ontario, N2L 3G1, Canada

***Abstract*—RFID (Radio Frequency IDentification) technology has moved from academic obscurity into mainstream business and practice. Although this technology has many catching and exclusive characteristics, security and privacy issues associated are not easy to address due to tags' modest computational and storage capabilities and the necessity to keep their prices low. This paper provides a general overview of the rather broad area of RFID security and privacy and gives the main citations for the comprehensive understanding and further explorations of this area. To be specific, previous attempts to enable or increase the security and privacy of the low-cost RFID systems are examined, including: (1) design and cryptanalysis of lightweight ciphers; (2) privacy-preserving authentication protocols and their designing requirements; and (3) non-conventional solutions leveraging physical characteristics of RFID tags or physical layer of the tag-reader communication for security and privacy purposes.**

I. Introduction

With the remarkable progress in microelectronics and low-power semiconductor technologies, RFID (Radio Frequency IDentification) technology has moved from academic obscurity into mainstream business and practice, due to its potentiality to settle a variety of real-world problems in low-budget ways. Nonetheless, some of the catching and exclusive characteristics of RFID systems raise new concerns about security and privacy, such as counterfeiting of tags for spurious products, rogue scanning of tags to track their bearers, which are quite challenging to deal with due to tags' modest computational and storage capabilities and the necessity to keep their prices low.

This paper selectively reviews the vast literature on various solutions to address miscellaneous security and privacy issues in low-cost RFID systems, where the existed solutions are grouped into the following three categories: (1) design and analysis of cryptographic primitives: cryptography is the most fundamental tool to resolve the security and privacy problems in general, where mathematically-sound algorithms/functions/schemes are proposed to enable confidentiality, integrity and availability. To make cryptography available for computationally weak devices, e.g., low-cost tags, crypto society has been engaged in remarkable efforts throughout recent years that lead to the born of a new sub-field, namely *lightweight cryptography*; (2) development of the protocols: identification/authentication protocols for RFID systems are substantially different from those for computer networks and sensor networks, since the tag bearer's privacy, among other factors, has to be considered. For example, such a protocol should guarantee at least anonymity and untraceability of a legitimate tag (thus its bearer), during and after its execution. Meanwhile, such a protocol should have a compact implementation and is able to operate in a frequent loss of power environment. Both communication and crypto societies made considerable contributions that conduce to the discovery of new paradigms for the designing and the analyzing; and (3) exploration of physical characteristics/layer for security and privacy: there has been a considerable recent attention devoted to exploiting physical characteristics of RFID systems, e.g. the time spent on communication, dielectric/conductive features of tags, or physical layer of the tag-reader communication, which was conventionally responsible for transmission/reception and error correction only, to achieve authenticity, confidentiality and integrity.

Hence, the remainder of this paper is organized as follows. We start with the recent process in the lightweight cryptographic primitives in Sect. II. Design of protocols for privacy-preserving identification/authentication are inspected in Sect. III. Besides, in Sect. IV, solutions leveraging the electronic characteristics of the physical devices or the randomness in the channels between communicating devices are examined. We conclude this paper in Sect. V. Note that, in the current paper, we particularly focus on the works exploring security and privacy as a matter between tags and readers rather than the massive infrastructure and security services of RFID systems, e.g., intrusion detection, forensics, data aggregation.

II. Design and Cryptanalysis of Lightweight Ciphers

The intensive studies toward design, implementation and analysis of lightweight cryptographic primitives for constrained devices lead to the born of a new sub-field of cryptography – *lightweight cryptography*, which is considered as the intersection of electrical engineering, computer science and mathematics. The major tasks of lightweight cryptography are as follows.

- *Design* of new cryptographic primitives/protocols, e.g., stream/block ciphers, hash functions, authentication methods, by: (1) optimizing implementations of standardized and trusted primitives/protocols, e.g., compact ASIC cores for 128-bit AES [36]; (2) tailoring well-investigated primitives/protocols to be more hardware-efficient and less computationally demanding, e.g., a lightweight variant of DES [62]; and (3) inventing brand new primitives/protocols with small hardware footprint, e.g., PRINTCipher [57].

- *Analysis* of the primitives/protocols proposed to ensure their cryptanalytic strength, which is even a more active topic. This is because, in the ongoing competition to reach the most efficient primitives, aggressive designs are adopted, including, but not limited to: (1) innovative techniques that are less well-understood and may potentially introduce vulnerabilities; (2) reduced security margins that cryptographic primitives are traditionally equipped with.

We summarize the lightweight ciphers recently proposed in Table I, where GE is the acronym of *Gate Equivalent*. As one may see, although the stream cipher usually inherits lower hardware complexity and typically operates at a higher speed, the majority candidates proposed are actually block ciphers. This phenomenon may imply that, to be secure enough, the current linear feedback shift register (LFSR)-based designs of stream ciphers can barely be tailored or optimized any further. In parallel to this, the nonlinear linear feedback shift register (NFSR)-based designs, although have a great potentiality due to the large linear complexity and the long period, remain in the infancy and are yet to be fully understood. Additionally, the structures of the proposed lightweight block ciphers can be primarily divided into *Substitution Permutation Network* (SPN), e.g., PRESENT, PRINTCipher, and *Feistel-type* structure, e.g., GOST, HIGHT, Piccolo. SPN is widely accepted due to its success application to AES, while Feistel-type structures, besides its successful application to DES, generally require a larger number of rounds due to its slow diffusion. However, a nice property of Feistel-type structures, as pointed out in [81], is that it supports the decryption function without increasing the implementation cost.

While detailed description of each candidate is impossible, we take a close look at several lightweight ciphers in what follows. The selected ones either theoretically present novel design philosophies/methodologies or practically integrate durable security and compactness in an elegant way.

A. PRINTCipher

PRINTCipher [57] is a novel lightweight block cipher proposed by Knudsen *et al.* in CHES'10, which is the first design that takes IC printing technology into consideration. The authors observed that: (1) a key is unlikely to be changed during a tag's life time; (2) IC printing does not require every copy of the cipher to be identical and a tag can thus be personalized with a unique key without extra cost.

PRINTCipher-48 (PRINTCipher-96 resp.) is a block cipher with $n = 48$-bit (96-bit resp.) block size and a key length of $l = 80$-bit (160-bit resp.), which adopts SPN structure with $r = 48$ ($r = 96$ resp.) rounds. One round of PRINTCipher-48 is shown in Fig. 1, where S represents a 3-bit S-box and P represents a 3-bit permutation. The key is split into two parts: the first n bits are used as the whitening key for each round, and the remaining $(l - n)$ bits are embedded into the permutation blocks, i.e., two bits are embedded into each P block during printing. A round counter RC_i is used to avoid self-similarity.

Cryptanalytic Results Although the designers claimed its security with respect to the main cryptanalytic methods, the first attack, discovered by Abdelraheem *et al.* in [4], appeared quite soon. This attack, exploiting the fact that the differential characteristics are key-dependent, successfully breaks 22 rounds of PRINTcipher-48 with 2^{48} computational steps. Their attack begins with a nice observation that, in S, all occurring differences are with probability $1/4$, and that for every 1-bit input difference, there exists exactly one 1-bit output difference. From this, it follows that starting with a 1-bit input difference, a 1-bit differential trail through r rounds of the cipher occurs with probability $(1/4)^r$. Additionally, if the 1-bit differential occurs, the S-box does not permute the active bit on a differential trail, which is only influenced by the fixed round permutation and the key-dependent permutation P. Therefore, knowing the best differential, one is able to deduce the key.

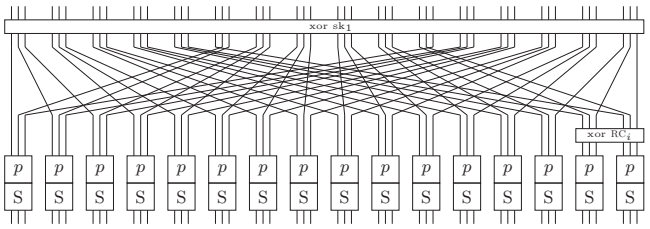

Figure 1. One Round of PRINTCipher-48.

In CRPYTO'11, the same group of researchers presented a so-called *invariant subspace attack* [61] that breaks the full cipher for a significant fraction of its keys, e.g., 2^{52} keys out of 2^{80} for PRINTCipher-48 and 2^{102} keys out of 2^{160} for PRINTCipher-96. The general idea of the invariant subspace is that the round function, including an SP-layer and a key addition layer, maps the affine subspace (out of the entire key space) onto itself. This property is preserved for an arbitrary number of rounds as long as all the round keys are in this subspace, which results in an efficient distinguisher for this fraction of the keys.

B. KATAN/KTANTAN Family of Block Ciphers

KATAN/KTANTAN [29] is a family of hardware oriented lightweight block ciphers proposed by Canniere *et al.*. Both KATAN and KTANTAN have three variants, which have block sizes 32-bit, 48-bit, and 64-bit respectively. All ciphers share the same key length of 80 bits, where the only difference between KATAN and KTANTAN is the key schedule.

For KATAN-32/KTANTAN-32, the plaintext is first loaded into two NFSRs denoted as L_1 and L_2 (of lengths 13-bit and 19-bit). Each round, out of the 254 rounds, L_1 and L_2 are shifted to the left by one position, where the new bits produced are loaded into the least significant bits of L_1 and L_2. By denoting the states of L_1 and L_2 as $(s_0, ..., s_{266})$ and $(l_0, ..., l_{272})$ through the whole encryption process, the nonlinear recursive relations can be represented as:

$$\begin{aligned} s_{13+i} &= l_i + l_{6+i} \cdot l_{8+i} + l_{11+i} + l_{10+i} \cdot l_{15+i} + K_i^s \\ l_{19+i} &= s_i + s_{5+i} + s_{4+i} \cdot s_{7+i} + IR_i \cdot s_{9+i} + K_i^l \end{aligned}$$

where IR_i is a round constant, and K_i^l and K_i^s are the two subkey bits generated by the key scheduling with the key K.

TABLE I. RECENT DESIGN/IMPLEMENTATION OF LIGHTWEIGHT CIPHERS

		Key size [bits]	Block size [bits]	Area [GE]	Throughput [Kb/s]	Logic Process [μm]
Trivum	[65]	80	N/A	749	100	0.35
Grain	[37]	80	N/A	1,294	100	0.13
PRINTCipher-48	[57]	80	48	402	6.25	0.18
PRINTCipher-48	[57]	80	48	503	100	0.18
PRINTCipher-96	[57]	160	96	726	3.13	0.18
PRINTCipher-96	[57]	160	96	967	100	0.18
KTANTAN-32	[29]	80	32	462	12.5	0.13
KTANTAN-48	[29]	80	48	571	9.4	0.13
KTANTAN-64	[29]	80	64	684	8.4	0.13
GOST	[72]	256	64	651	24.24	0.18
LED-64	[42]	64	64	688	5.1	0.18
LED-128	[42]	128	64	700	3.4	0.18
Piccolo-80	[81]	80	64	683	14.8	0.13
Piccolo-128	[81]	128	64	758	12.1	0.13
KATAN-32	[29]	80	32	802	12.5	0.13
KATAN-48	[29]	80	48	916	9.4	0.13
KATAN-64	[29]	80	64	1,027	8.4	0.13
PRESENT-80	[78]	80	64	1,075	11.4	0.18
KLEIN-64	[40]	64	64	1,981	N/A	0.18
KLEIN-80	[40]	80	64	2,097	N/A	0.18
KLEIN-96	[40]	96	64	2,213	N/A	0.18
DESXL	[64]	184	64	2,168	44.4	0.18
mCrypton-128	[62]	128	64	2,500	492.3	0.13
CLEFIA-128	[2]	128	128	2,678	73	0.13
XTEA	[55]	128	64	3,490	57.1	0.13
AES	[36]	128	128	3,400	12.4	0.35
HummingBird-2	[35]	128	16	2,159	N/A	0.13

For the KATAN family, the 80-bit key is expanded to a sequence of 508-bit by an 80-stage LFSR, and two bits from the sequence are applied to each round. As opposed, the key schedule of the KTANTAN family is designed under the consideration that the key is likely to be fixed to the tag. Therefore, it utilizes MUX, AND gate and XOR gate to schedule the key bits in a nonlinear and more efficient way.

Cryptanalytic Results In SAC'11, Bogdanov *et al.* in [22] found a vulnerability in the key scheduling of KTANTAN: several key bits are not used until very late in the cipher, while some others are never used after some surprisingly small number of rounds, which results in a *3-subset meet-in-the-middle attack*. The reported attack is of time complexity $2^{75.170}/2^{75.044}/2^{75.584}$ on the full KTANTAN-32/48/64. Recently, Zhu and Gong presented in [92] a novel extension – guessing the intermediate state before launching the meet-in-the-middle attack, and obtained the best cryptanalytic results so far, e.g., KTANTAN32/48/64 can be broken with the time complexities of $2^{68.06}/2^{70.92}/2^{73.09}$. Unfortunately, the proposed attacks are not significantly better than the exhaustive search and none of them affects KATAN.

In Indocrypt'10, Bard *et al.* [10] presented several experimental results on the reduced rounds of the KATAN family, i.e., 60/40/30 rounds of KATAN-32/48/64, under cube, algebraic and side channel attacks. However, their results are marginal as they are effective toward a small number of rounds. Knellwolf *et al.* proposed to use *conditional differential cryptanalysis* to attack KATAN in [58]. Unlike the conventional differential cryptanalysis that the input pairs are selected uniformly at random, conditional differential cryptanalysis asks for inputs pairs satisfying stringent conditions to control the propagation of the differentials. Since the round function of KATAN/KTANTAN has the slow diffusion, this method works for a particular number of rounds and leads to the recovery of 4 key bits of 78 rounds of KATAN-32 under a single-key scenario and the recovery of 10 key bits of 120 rounds under a related-key scenario. Consequently, this method, though is of theoretic interesting, does not jeopardize the practical security of the KATAN family.

C. Piccolo

In CHES'11, Shibutani *et al.* proposed in [81] a new 64-bit block cipher, called Piccolo. Piccolo supports 80 and 128-bit keys, which are referred as Piccolo-80 and Piccolo-128 respectively. The differences between Piccolo-80 and Piccolo-128 are the number of rounds and the key schedule.

As shown in Fig. 2, Piccolo, consisting of r rounds, takes $X = (X_0, X_1, X_2, X_3) \in \mathbb{F}_2^{64}$, four whitening keys $wk_i \in \mathbb{F}_2^{16}, i = 0, 1, 2, 3$, and $2r$ round keys $rk_i \in \mathbb{F}_2^{16}, 0 \leq i < 2r$, as the inputs, and outputs $Y \in \mathbb{F}_2^{64}$. In each of the r rounds, following is performed

$$\begin{aligned} X_1 &= X_1 + F(X_0) + rk_{2i} \\ X_3 &= X_3 + F(X_2) + rk_{2i+1} \\ (X_0, X_1, X_2, X_3) &= RP(X_0, X_1, X_2, X_3), \end{aligned}$$

where $F : \mathbb{F}_2^{16} \mapsto \mathbb{F}_2^{16}$ is realized by four paralleled 4-bit S-boxes followed by multiplying a diffusion matrix

over $\mathbb{F}_2^4$, and $RP : \mathbb{F}_2^{64} \mapsto \mathbb{F}_2^{64}$ is the round permutation that transforms $(x_0, x_1, x_2, x_3, x_4, x_5, x_6, x_7)$ to $(x_2, x_7, x_4, x_1, x_6, x_3, x_0, x_5)$, where $x_i \in \mathbb{F}_2^8$ for $i = 0, ..., 7$.

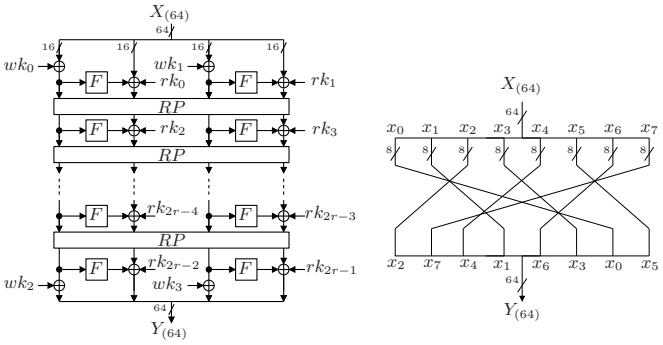

Figure 2. Encryption in Piccolo (left) and Round Permutation RP (right)

The key schedule of Piccolo-80 divides the 80-bit key K into five 16-bit subkeys $k_i \in \mathbb{F}_2^{16}, i = 0, 1, 2, 3, 4$, and produces round keys $rk_j \in F_2^{16}, j = 0, 1, ..., 2r - 1$, deterministically. In addition, the key schedule also generates $wk_i \in F_2^{16}, i = 0, 1, 2, 3$, through $wk_0 = (k_0^L, k_1^R)$, $wk_1 = (k_1^L, k_0^R)$, $wk_2 = (k_4^L, k_3^R)$ and $wk_3 = (k_3^L, k_4^R)$, where k^L and k^R are left and right half eight bits of k, respectively.

Cryptanalytic Results There is no published cryptanalytic result on this new primitive besides its designers' self-evaluation, where Piccolo is claimed to have enough immunity against differential/linear cryptanalysis, related-key attacks and MITM/slide/saturation attacks.

D. RC5

To be away from the dedicated designs targeting hardware lightweightness as aforementioned, RC5 [74] is a conventional word-oriented block cipher developed in 1994, that has a variable block size $2w = 32, 64$ or 128 bits, a variable number of rounds $r = 0$ to 255, and a variable key size $b = 0$ to 255 bytes. Therefore, a particular parameterized RC5 is usually denoted as RC5-w/r/b, e.g., RC5-32/12/16 refers to the instance of RC5 encryption/decryption with a block size of 64 bits, 12 rounds and a 128-bit key. Despite its popularity, the RC5 encryption is mathematical straightforward that uses two registers, i.e., A of w-bits and B of w-bits, an expanded key table S consisting of $(2r + 2)$ w-bit words derived by the key expansion algorithm, and three simple operations that are efficient on most processors, i.e., "$+/-$" addition/subtraction modulo 2^w, "$\lll$ / $\ggg$" left/right rotation and "$\oplus$" XOR. RC5 encryption can thus be represented as:

$$\begin{aligned} A &= A + S[0], \quad B = B + S[1], \\ A &= ((A \oplus B) \lll B) + S[2 \cdot i], \qquad \text{for } i = 1, ...r, \\ B &= ((B \oplus B) \lll A) + S[2 \cdot i + 1], \ \text{for } i = 1, ...r. \end{aligned}$$

Note that the ciphertext is the final states of the registers A and B.

The feasibility of RC5-32/12/16 to be executed on a general-purpose UHF tag has been confirmed by Chae *et al.* in [28], where RC5-32/12/16 has been implemented on a WISP tag driven by MSP430F1232 micro-controller containing 8000 bytes of flash and 256 bytes of RAM. Their empirical study shows that an RC5 encryption of 64-bit data can be finished within 1.43 million second when the WISP tag is one foot away from the reader, and identifies the actual power measurements of the tag beyond the pure arithmetic estimation of the computational overhead.

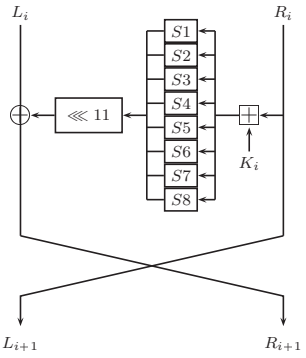

Figure 3. Round Function of GOST. Note that $(S_1, \ ., S_8)$ represents a set of eight S-boxes, which could be selected at the user's will. For example, to minimize the hardware footprint, one S-box is used eight times in parallel in [72].

Cryptanalytic Results RC5 has been analyzed intensively in the past two decades. Currently, the most impressive attack can be found in [15] that requires 2^{44} chosen plaintexts and 2^{36} memory to derive the full set of 25 subkeys for the 12 round RC5 with 64-bit block length. During FSE'99, Borst *et al.* in [21] proposed to use the linear cryptanalysis to mitigate the dependent of the large amount of memory as required by previous attacks. However, this refined method is only effective to attack RC5 with up to 10 rounds.

E. GOST

GOST is the oldest cipher in the field of lightweight cryptography, which was designed in the Soviet Union during 1970's as an alternative to DES. This cipher has received resurrect attention due to the recent effort of Poschmann *et al.* in [72] – they found that GOST is highly efficient in ASIC implementation, e.g., 651 GE.

GOST has a block size of 64 bits and a keysize of 256 bits. The overall structure, as shown in Fig. 3, is a two branch Feistel network with 32 rounds, in which the right half of the block R_i is processed, XORed to the left half L_i, and swapped with the left half. In addition, GOST has a simple key schedule: the 256-bit key is divided into eight 32-bit words, i.e., $K_1, K_2, ..., K_8$. Each round, GOST uses one of them according to the array given below, e.g., K_1 is used in rounds $1, 9, 17$ and 31,

$$K_0, K_1, K_2, K_3, K_4, K_5, K_6, K_7, K_0, K_1, K_2, K_3, K_4, K_5, K_6, K_7,$$
$$K_0, K_1, K_2, K_3, K_4, K_5, K_6, K_7, K_7, K_6, K_5, K_4, K_3, K_2, K_1, K_0.$$

Cryptanalytic Results Although in the past 20 year, GOST has been intensively studied and several related-key attacks and signal-key attacks targeting round-reduced version of GOST have been published, the first single key attack on the full GOST was published recently by Isobe in [51], which leverages the property that, providing $R_{24} = L_{24}$ (which happens with probability 2^{-32}), the last 16 rounds become an identity mapping, and thus the effective number of rounds

of GOST is reduced to 16. Based on this property, Isobe combined the 3-subset meet-in-the-middle attack as mentioned to extract the entire secret key with 2^{32} plaintext/ciphertext pairs, 2^{224} time and 2^{64} memory. This idea has been further extended in [30] such that given the same amount of known plaintext/ciphertext pairs, the memory complexity can be reduced to 2^{36}. Although recent progress in analyzing this cipher is remarkable, neither the time complexity nor the memory complexity are even close to be practical, thanks to its abundant security margin.

F. Other Lightweight Primitives

Besides the ones summarized in Table I, there are a multitude of other lightweight designs and analysis of cryptographic primitives.

Message Authentication Code Shamir in [80] proposed SQUASH, which, though based on the Rabin public-key cryptosystem, performs exceedingly well on benchmarks. By denoting tag's response, tag's secret key, reader's challenge, and truncation function as R, K, C, and T respectively, SQUASH can be simply represented as

$$R = T((\sum_{i=0}^{l-1} f_i(K, C))^2 \bmod N),$$

where the f_i's are the nonlinear mixing functions realized by one NFSR. Note that N is a composite Mersenne number, e.g., $N = 2^{1277} - 1$, which is not only easy to store (since its binary representation is a sequence of 1277 ones), but also makes the modular computation particularly simple. Khaled and Serge later in [71] analyzed and attacked an early version of SQUASH, which uses an LFSR as the mixing function. Nevertheless, the security of SQUASH in general remains open.

Hash Function As early as in 2003, Weis in his thesis [87] discussed the possibility to build a low-cost hash function from NFSRs. However, this idea has not been elaborated so far. Rather recently, lightweight hash functions began to receive attention. Bogdanov *et al.* in [19] firstly described a way of using the PRESENT block cipher in the hashing mode and achieved 64-bit collision resistance with 1600 GE. Aumasson *et al.* in [3] designed a dedicated lightweight hash function Quark (1379 GE for 64-bit collision resistance) using sponge functions as the domain extension algorithm and an internal permutation inspired by the designs of GRAIN and KATAN; Guo *et al.* in [41] proposed a family of lightweight hash functions uses a sponge-like construction as the domain extension algorithm and an AES-like primitive as internal unkeyed permutation and achieved 1120 GE for 64-bit collision resistance. The latest proposal is SPONGENT in [18], which has the smallest footprint among all hash functions published so far at all security levels it attains.

Public-key Schemes Lightweight public-key schemes represent another promising avenue of research, even though their implementations remain too heavyweight at the current stage: WIPR in [68] is a full-fledged public key identification scheme following the idea of randomized Rabin function [79], which is secure, e.g., 1024-bit, yet highly efficient, e.g., 5705 GE. The security of reduced WIPR is investigated in [89], and two variants are proposed to improve its security and to further reduce its hardware footprint; Pendl, Pelnar and Hutter in [73], presented their results of implementing the *elliptic curve cryptography* on a WISP tag, and confirmed that a scalar multiplication using the Montgomery powering ladder demands 1.6 seconds, which cannot meet the practical requirements in majority RFID applications.

III. Lightweight Authentication Protocols

A. Security Requirements

Notwithstanding RFID systems are principally simple at first glance, design of the identification/authentication protocols is quite challenging, e.g., such a protocol should ensure both the anonymity and the untraceability of a legitimate tag during and after the executions[1]. Generally speaking, designing requirements of such a protocol can be characterized by security, privacy and performance, as below.

- Fundamental Requirements: (1) **Authenticity** (security): After execution of the protocol, the reader could identify a legitimate tag with overwhelming certainty, while, an adversary can impersonate any legitimate tag (reader resp.) at the reader (tag resp.) with negligible probability; (2) **Anonymity** (privacy): A protocol between the reader and tag does not leak any fixed or predictable patterns related to a tag's ID or pseudo-ID; (3) **Untraceability** (privacy): The adversary is not able to tell whether a transaction after time $t + \delta$, $\delta > 0$, involves the tag, after eavesdropping on the reader-tag communication before t for a sufficient number of rounds; (4) **Computation/Storage Efficiency** (performance): a suitable protocol must be efficient enough (at least on the tag side) to be implemented and deployed to real-world low-cost RFID systems.
- Additional Requirements: (5) **DoS Resistance** (security): Blocking of arbitrary number of sessions of the reader-tag communication before time t does not affect the success probability of the execution of the protocol after t; (6) **Backward Untraceability** (privacy): If the adversary reveals the internal state, e.g., the secret key, of a tag at time t, the adversary is not able to tell whether a transaction before time t involves the tag; (7) **Forward Untraceability** (privacy): If the adversary reveals the internal state of a tag at time t, the adversary is not able to tell whether a transaction after time $t+\delta$, $\delta > 0$, involves the tag, provided that the adversary does not eavesdrop on every reader-tag communication after time t; (8) **Scalability** (performance): A protocol must be scalable to allow the reader to deal with a large tag population.

Roughly speaking, a "high quality" protocol should have all of the fundamental requirements satisfied, and necessary additional requirements satisfied according to its use. An observation one may make from above are the inherit contradictions in several pairs of these requirements, which are the root causes that the design of such a protocol is non-trivial. To name few,

- Security v.s. Performance: as aforementioned, efficiency is usually obtained at the cost of losing certain security margin.
- Privacy v.s. Scalability: a tag must encrypt its identity with a secret key so that only authorized readers can extract the identity, while an authorized reader, in order to authenticate the tag, needs to know the identity of the tag in order to determine the key associated. Unsurprisingly, the reader has to try every key in its database until a valid one is found. This is the *key search problem* [52], which remains unsolved if the underlying cryptographic primitives are symmetric.
- Privacy v.s. Scalability v.s. Efficiency: the above problem can be perfectly solved by introducing public-key primitives, which is less computation- and storage-efficient at least at present.

In what follows, we summarize the recent progress in this area. Although protocols based on public-key primitives show interesting properties, we constrain ourselves to the symmetric-key-based protocols, as practicality is more concerned in this survey. To deal with the large body of literature focusing on the design and analysis of RFID protocols, we roughly classify related works according to the characteristics they achieve best, i.e., scalability oriented protocols, backward untraceability oriented protocols, untraceability oriented protocols and performance oriented protocols. Note that the classes we present here are not mutual exclusive, e.g., the protocol proposed in [16] can be seen as a scalability oriented protocol as well as a performance oriented protocol.

[1]Zero knowledge proof, which prevents any leakage of the secret information of the prover, is a classical solution to handle such a problem. However, zero knowledge proof seems too heavyweight to be deployed to low-cost tags.

B. Scalability Oriented Protocols

This class of protocols starts from Weis's early work in [87], known as the *randomized hash lock*, which is an access control mechanism to lock the ID against unauthorized readers and works as below:

1) Reader: query to wake up the tag.
2) Tag: respond $(h(ID, n_t), n_t)$, where ID is the tag's identity while n_t is a nonce
3) Reader: accepts the tag when there is an ID in the accepted ID list; If accepted, the reader sends back the secret key to unlock the tag.
4) Tag: the tag sends back its ID in plaintext.

Unfortunately, the scheme allows a tag to be tracked by an eavesdropper, because the ID is used repeatedly. Moreover, the key and ID are transferred in plaintext after a success verification, that is problematic.

Molnar and Wagner [67], by extending Weis's work, proposed a tree-based scheme for library RFID applications. Similar as the Merkle tree, they view N tags as leaves in a binary tree while each edge in the tree is associated with a secret. Each tag stores $\log N$ secrets according to the path from the root to the leaf. During the authentication, the reader starts at the root and uses the secret to check whether the tag uses the "left" secret or the "right" secret. If the reader successfully verifies the tag using one of these two secrets, the authentication proceeds to the next level of the tree. If the reader passes all secrets in the path, the tag accepts the reader. Although this protocol is scalable, it needs $O(\log N)$ rounds of interaction and $O(\log N)$ storage on the tag. Furthermore, more tags an adversary compromises, more secrets in the tree are exposed.

Burmester *et al.* in [12] described an authentication scheme with constant key-lookup, which is one of the most scalable solutions that preserve privacy as claimed. However, a subtle flaw is found in [63], exploiting the fact that an attacker can launch a three-run interleaving attack to trace and identify a tag. An improved version of this protocol is also presented in the same paper. Moreover, Cheon *et al.* in [27] exhibits an interesting idea: use the meet-in-the-middle strategy (usually used to attack symmetric ciphers with poor key schedules, as mentioned in the previous section) to reduce the reader computation to $O(\sqrt{N}\log N)$.

C. Backward Untraceability Oriented Protocols

Ohkubo, Suzki and Kinoshita in [70] proposed a one-way authentication protocol, known as the OSK protocol, to realize the backward traceability. To be specific, their protocol works as below:

1) Reader: query to wake up the tag.
2) Tag: respond $M = g(s)$ and update s to be $h(s)$, where s is the secret shared by the reader and the tag, g and h are hash functions.
3) Reader: compute $g(h^j(s))$ for s of each tag in the database until reaching M, where h^j is the jth composition of h, e.g., $h^2(s) = h((h(s))$.

Despite the vulnerability under the replay attack and poor scalability, this early protocol introduced an innovative concept, i.e., refreshing the state of the tag each time it has been queried. In [7], Avoine and Oechslin reduced the reader's search complexity by using a specific time-memory trade-off. Besides, they noticed that when the two hash functions are modeled as random oracles, the security of this scheme against a strong model of attackers can be proven.

YA-TRAP [85] is designed to achieve untraceability even when the tag is compromised. In this scheme, a tag pre-shares a time interval $[T_t, T_{max}]$, where T_t denotes the last time it was interrogated, and a secret key with the reader. The reader challenges the tag by sending the current time T_r; if T_r is within the interval $(T_t, T_{max}]$, the tag responds with a keyed hash value of T_r which can be verified by the reader, and updates T_t with T_r; otherwise, the tag outputs a random number that an adversary is unable to distinguish. Unfortunately, YA-TRAP is vulnerable to both database-side DoS attack and tag-side DoS attack by incapacitating a tag by sending a wildly inaccurate "current time". To resolve this, O-TRAP [23] introduces the resynchronization mechanism. However, each resynchronization causes the $O(l \times n)$ search burden to the backend database, where n is the number of tags and l is the steps required to ensure synchronization across the hash chain (the adversary could make l a huge number). Hence, its practicality is questioned.

As one may expect, all previous protocols offering backward untraceability requires on-tag hash function, which is prohibitively expensive for RFID tags and possesses an unnecessary security property, namely, the *collision resistance*. In FSE'10, Billet *et al.* in [13] proposed a privacy-preserving mutual authentication protocol based upon a stream cipher and proved that this protocol achieves security, efficiency and a strong privacy close to the backward untraceability. This protocol works as below:

1) Reader: query the tag with a nonce n_r.
2) Tag: respond (n_t, G_t), where n_t is a nonce contributed by the tag, $G_t||G_r||G_s = G(n_t||n_r, K)$ is a sequence produced by the stream cipher G with $IV = n_t||n_r$ and the key K.
3) Reader: search potential Ks such that the produced G_t matches the received one; if so, respond with G_r and update the key to be G_s.
4) Tag: if the received G_r matches the G_r produced locally, update the key to be G_s.

In all, every protocol in this class implicitly demands a stateful tag, i.e., a tag has nonvolatile memory, and enough power provided by a reader to enable the tag's accessing of the memory, which may increase the overall cost of the whole system.

D. Untraceability Oriented Protocols

Song and Mitchell in [82] demonstrated a complicated authentication scheme targeting both forward and backward untraceability, a simplified description of which is given below:

1) Manufacture: before the start of this protocol, assign the reader a pair $(u \in \mathbb{F}_2^l, t = h(u))$ and assign the tag a value $t = h(u)$, where h is a hash function and l is a positive integer.
2) Reader: query the tag with a nonce $n_r \in \mathbb{F}_2^l$.
3) Tag: respond (M_1, M_2), where $M_1 = t + n_t$, $M_2 = f_t(n_r + n_t)$ and $n_t \in \mathbb{F}_2^l$ is the nonce contributed by the tag.
4) Reader: search for a potential t such that $n_t = M_1 + t$ and $M_2 = f_t(n_r + n_t)$ and generate and respond with $M_3 = u + (n_t \ggg l/2)$, where $\ggg$ denotes the right circular shift.
5) Tag: compute $u = M_3 + (n_t \ggg l/2)$ and update t with $h((u \lll l/4) + (t \ggg l/4) + n_t + n_r)$ if $h(u)$ equals t, where $\lll$ denotes the left circular shift.
6) Reader: update u with $((u \lll l/4) + (t \ggg l/4) + n_t + n_r)$ and t with $h(u)$.

This protocol is forward-untraceable from the moment that the adversary misses M_3. Besides, this protocol is secure against tag/reader impersonation, replay, DoS attacks (in its full version, the previous pair of (u, t) is actually stored by the reader for the purpose of resynchronization) and backward untraceability. The scalability is the only remaining problem for this design. Another example protocol that supports forward and backward untraceability is [20].

E. Performance Oriented Protocols

HB Family An interesting avenue of research was initiated by Juels and Weis' HB$^+$ protocol [54], which is a probabilistic algorithm to authenticate a tag to a reader while hiding the tag's identity from an eavesdropper using extremely simple algebraic operations. The security is reduced to the difficulty of the *Learning Parity with Noise* (LPN) problem, which has been proven to be NP-hard. In this protocol, a tag and a reader share a secret vector $(\mathbf{x}, \mathbf{y})$ and during each round,

1) Reader: wake up the tag.
2) Tag: respond with a blinding-factor vector $\mathbf{b}$, which is randomly generated.
3) Reader: randomly select $\mathbf{a}$ as the response.
4) Tag: generate a noise bit u which takes "1" with probability η, i.e., $\text{Prob}[u = 1] = \eta$, compute and respond with $z = (\mathbf{a} \cdot \mathbf{x}^T) + (\mathbf{b} \cdot \mathbf{y}^T) + u$.
5) Reader: independently compute $z' = (\mathbf{a} \cdot \mathbf{x}^T) + (\mathbf{b} \cdot \mathbf{y}^T)$, and validate the tag's response if $z = z'$.

After n rounds, the authentication succeeds if and only if there is no more than $\lceil \eta n \rceil$ mismatched responses. Assuming the intractability of LPN problem, the HB$^+$ protocol is provably secure against passive eavesdroppers.

However, an active adversary can easily break this protocol as identified by Gilbert *et al.* in [45], known as the GRS attack. The attacker first modifies one bit of $\mathbf{a}$ to be $\mathbf{a} + \alpha$ for the second pass of every round of HB$^+$ and observes the authentication result, e.g., acceptance or rejection, to learn $\alpha \cdot \mathbf{x}$. By repeating this lightweight process sufficient number of times, the attack could learn $\mathbf{x}$. The same GRS manipulation can be applied to blinding vectors $\mathbf{b}$ to recover $\mathbf{y}$. Although several variants are proposed to thwart the GRS attack, it is pointed in [43] that none could survive in GRS-like attacks.

In EUROCRYPT'08, Gilbert *et al.* in [44] presented another two interesting variants, known as Random-HB$^\#$ and HB$^\#$, which are proved to be resistant to the GRS attack in the sense that the adversary is only allowed to manipulate the challenges from the reader to the tag. The core idea is to use two secret matrices $\mathbf{x}$ and $\mathbf{y}$ to replace the secret row vectors in HB$^+$, e.g., Random-HB$^\#$ uses two random matrices while HB$^\#$ utilizes two Toeplitz matrices to save space. At AsiaCrypt'08, Ouafi, Overbeck, and Vaudenay in [69] presented a general man-in-the-middle attack against all HB-like protocols, where the adversary is given the ability to modify all messages. Surprisingly, it recovers the shared secret in 2^{25} or 2^{20} authentication rounds for HB$^\#$ and 2^{34} or 2^{28} for Random-HB$^\#$, depending on the parameter set. The crucial observation exploited by the OOV attack is that, providing an adversary modifies the messages going in both directions in a smart way, he can compute the hamming weight of the vector $\bar{\mathbf{a}}\mathbf{x} + \bar{\mathbf{b}}\mathbf{y} + \bar{\mathbf{z}}$, where $\bar{\mathbf{a}}, \bar{\mathbf{b}}, \bar{\mathbf{z}}$ are the modifications applied to $\mathbf{a}, \mathbf{b}, \mathbf{z}$ in the victim protocol.

To completely get rid of this kind of attacks based upon the underlying linearity in the protocol, there is a recent trend to replace the linear encoding or vector/matrix multiplication by nonlinear operations, e.g., [66]. Most recent, Li, Gong and Qin in [60] produced another derivative, which uses a lightweight yet secure (under ciphertext-only attack) encryption/decryption based on the circulant matrix multiplication/inversion to replace the underlying linear encoding. The security of this protocol is also proved under a generalized man-in-the-middle model.

In all, HB$^+$ is still the most elegant design in this family, given its lightweightness and proved security under the passive attacks, which seems to be suffice for low-cost RFID systems. Its variants, though providing extra security under active attacking models, are unnecessarily complicated and away from the original design goal. It is worth to mention, as a conventional challenge and response protocol, HB family still has the scalability problem as the reader has to go through every possible key to identity and authenticate the current tag. Besides, a public denunciation of protocols in this family is that the success of the authentication is only guaranteed within a certain probability less than one.

EPC Gen2 Family EPC Gen2 tag is designed to strike the balance between cost and functionality, with little attention paid to security. To address this problem, particular protocols are proposed exclusively for this standard in light of the basic algebraic operations provided by the EPC Gen2 tags, such as XOR, shift, rotate and cyclic redundancy check (CRC), which are highly linear. After several failed attempts, a common agreement has been reached, i.e., to design a secure protocol targeting even only fundamental requirements without using nonlinear modules is unlikely to succeed.

With this consideration, Blass *et al.* in [16] proposed F_f-family of protocols which is the first cross-layer design of cryptographic primitive, e.g., an HMAC-like function F_f, and the protocol that relies on it. Unfortunately, the authors of [14] found the connections between the F_f protocol and the LPN problem, and showed a key-recovery attack with time complexity of about 2^{38} against the instance equipped with a 512-bit secret key. Nevertheless, this design philosophy deserves further investigation as it exhibits a promising way to achieve necessary security by customizing the underlying cryptographic primitives.

IV. Physical Layer Approaches

In a conventional sense, security and privacy are viewed as an independent feature addressed above the physical layer, and all such mechanism as mentioned are designed and implemented with the assumption that the physical layer has already been established and provides an error-free link. Motivated by Wyner's early work [88] and advances in the communication technologies, there has been a considerable recent interest on studying the fundamental ability of the physical layer to provide security for upper layers. Compared to the prevalent cryptographic approaches, the physical layer approach presents embedded security properties by utilizing random processes from physical world. In this section, we survey the previous designs leveraging the physical characteristics for security and privacy purpose.

A. Distance Bounding Protocols

Mafia fraud was discovered during 1980s that allows the adversary to pass any cryptographic authentication protocols by relaying the messages between a verifier and a legitimate prover. The widely accepted countermeasure from the physical layer, known as *distance bounding protocol*, was proposed in [11] that makes use of two facts: (1) nothing travels faster than light; (2) any relaying operation takes time, which may lead to testable delay and may inform the verifier that the responses might be illegible, even they are algorithmically correct.

A distance bounding implicitly accomplishes, as pointed out by [1], *authentication* and *distance checking*. Authentication, in its conventional sense, is a process whereby one party is assured of the identity of another party involved, while distance checking refers to a process whereby one party is assured, through acquisition of corroborative evidence, e.g., the *round trip time* (RTT) of messages, that a given property on its distance to another party involved is satisfied. Nevertheless, a scheme satisfying both requirements are non-trivial. For example, any authentication protocol provides authenticity but may not be suitable for distance checking, e.g., if, for provers, the time spending on computing the authentication codes is non-negligible, the RTT (thus the distance) would be quite inaccurate for provers with different computation capabilities. As a consequence, any authentication protocol that provides distance checkability must

enable, at least for one party, a quick way to respond. The existed designs all follow this principle.

Hancke and Kuhn's Protocol Brands and Chaum [8] designed the first distance bounding protocol, which is published in EUROCRYPT'93. This protocol works as follows:

- Slow Phase: Both the verifier and the prover generate random binary sequence $C = (c_1, c_2, ..., c_n)$ and $R = (r_1, r_2, ..., r_n)$, respectively.
- Fast Phase: The verifier transmits one challenge bit c_i at the ith time slot, $i = 1, ..., n$, to which the prover responds immediately with r_i. The verifier times the delay between sending c_i and receiving r_i.
- Slow Phase: After all n bits have been exchanged, the prover completes the protocol by transmitting a message authentication code or digital signature for the two binary sequences of C and R.

Hancke and Kuhn's observed in [49] that the last phase of Brands and Chaum's protocol may not be indispensable, thus proposed the first distance bounding protocol for RFID applications. Note that in the scenarios of RFID, the prover is always the tag while the verifier is always the reader. In USENIX'07, Drimer and Murdoch in [34] implemented Hancke and Kuhn's protocol on the Chip & PIN payment system as a practical solution for the relay attacks identified in the same paper.

Multistate Enhancement of Hancke and Kuhn's Protocol A potential vulnerability in this protocol, as expected by sharp readers, is that after eavesdropping the slow phase, the attacker could start to query the victim tag in prior to the start of the fast phase. In this interaction with the tag, the attacker randomly selects a $\hat{c}_i$ (and unsurprisingly he has 0.5 chance to earn $\hat{c}_i = c_i$) and stores the tag's corresponding response. On the other hand, to interact with the reader in the fast phase, he responds using the tag's answer if $\hat{c}_i = c_i$; responds using a random bit otherwise. Therefore, he is expected to fool the reader with $(3/4)^n$ chance, after n bits of c_i are committed. Avoine, Floerkemeier and Martin further modified the Hancke and Kuhn's protocol by introducing another states, called *void challenges*, in [5], as well as generalized this approach to be multistate that improves all existing distance bounding protocols, i.e., the attacker has a success probability of $(5/9)^n$.

Kim and Avoine's Protocol In CANS'09, Kim and Avoine introduced another enhanced version in [56] of Hancke and Kuhn's protocol, as shown pictorially in Table II, based on binary mixed challenges, that converges toward the expected and optimal $(1/2)^n$ bound in case of both noisy and error-free channels.

Implementation Issues of Distance Bounding Protocols Since the underlying theory of distance bounding protocols is mature enough, the attention has shifted to the practical implementations of this technology, which is, counter-intuitively, quite challenging, i.e., its security depends not only on the logic designs, but also on the signal designs at lower layer.

To be specific, at the physical layer, if an attacker could start a response within the allowable time window but still change the value at a later stage, once he knows the correct response, the protocol's security would be compromised. For example, if the reader's receiver integrates the signal amplitude over an entire bit period for demodulation, the attacker could send no energy for the initial $(m-1)/m$ of the time interval and then send an m-times stronger-than-normal signal during the final $1/m$ of the time interval reserved for the bit. By this method, known as *deferred bit signaling* in [48], the attacker could delay committing to a bit's value by $(m-1)/m$ of the bit period. To mitigate this, an UWB (ultra-wideband) transceiver is designed and analyzed in [46]. Besides, Rasmussen and Capkun in [76] further investigated the design of a transceiver that is able to receive, process and transmit signals in real time (less than 1ns). Their implementation leverages a greatly interesting fact – the time needed for signal conversion and demodulation can be saved if the protocol is designed in a proper way.

Others Trujillo-Rasua *et al.* in [86] proposed an instance of the graph-based protocol that resists to both mafia and distance frauds without sacrificing memory. Capkun *et al.* exhibited in [25] another direction of the distance bounding technologies by considering a set of provers interact with a set of verifiers, which is motivated by applications such as group device pairing and location-based access control. Their key idea is to let the passive verifiers learn the distance bounds while the active verifier executes distance bounding with the prover. Rasmussen and Capkun in [75] analyzed location privacy problem in the distance bounding protocols by showing that the location and the distance between communicating partners can be leaked to even passive attackers. A mutual distance bounding protocol is proposed in [90], which uses an additional binary sequence to determine, for the two participants, who plays the role of prover/verifier.

B. Channel Impairment for Good

There has been a considerable recent attention on investigating the security implications of the physical layer, known as *wireless physical layer security*. The breakthrough concept behind is to exploit the characteristics of the wireless channel, such as fading or noise, which are traditionally considered as impairments. Following Wyner's original ideas [88] proposed in 1970s, the theorists intend to create a clear and clean framework by evaluating secret capacity for different channel models and constructing algebraic codes to achieve these optimums, while the practitioners try to apply this idea to the following topics, which are conventionally covered by cryptography:

- Key generation/extraction using channel reciprocity, e.g., [6]. Note that an implicit requirement for key generation/extraction is that the participants are equally powerful that immediately excludes low-cost RFID systems.
- Confidentiality- or integrity- or availability-preserving channel construction under eavesdropping, modification, jamming attacks. Due to the wireless nature of RFID systems, results from the physical layer security are applicable to RFID scenarios. Henceforth, brief descriptions of schemes falling into this sub-category are quite necessary.

To construct a unidirectional confidentiality-preserving channel from the tag to the reader, cooperative-jamming methods are intensively considered. To protect the unwanted scanning of tags, Jules in [53] proposed a conceptual scheme that the common tag and the blocker tag (both worn by the customer) transmit two identifiers at the same time, where the latter could simulate the full set of all possible k-bit identifiers of tags, which are arranged in a binary tree of depth k. This tree is then traversed by the reader, who queries tags in a bit-by-bit manner. Once the reader queries the bit that lies in the "privacy zone" of the tree, the blocker tag simultaneously broadcasts both a '0' bit and a '1' bit that "blocks" the reading of wanted tags. However, bitwise synchronization and pre-shared secrets required between the reader and the friendly jammer may be problematic in real-world applications. Melanie *et al.* implemented a battery powered device, the *RFID Guardian*, in [77], which not only produces jamming signals, but also allows the user to define which party can perform what operation on which population of tags. Later in [24], the cooperative-jamming method has been refined for the key distribution, where the blocker tag is owned by the reader rather than the user. In addition, Bringer and Chabanne in [9] discovered the underlying connection between the wiretap channel [88] and the proposed schemes. In CHES'07, Savry *et al.* in [83] designed and implemented a noisy reader by exploiting the fact that a passive tag is able to modulate a noisy carrier generated by a reader during its reply. However, this approach could cause severe disruption of all nearby RFID systems as well.

To construct a unidirectional confidentiality- and integrity-preserving channel, recently, Chai *et al.* in [26] designed a novel physical layer scheme (without introducing any tag-side cost) in light

TABLE II. KIM AND AVOINE'S PROTOCOL

Prover		Verifier
	Slow Phase	
generate N_P	$\xleftarrow{N_V}$	generate N_V
	$\xrightarrow{N_P}$	
$H^{4n} = H(x, N_P, N_V)$		$H^{4n} = H(x, N_P, N_V)$
$R^0 = (H_1, ..., H_n)$		$R^0 = (H_1, ..., H_n)$
$R^1 = (H_{n+1}, ..., H_{2n})$		$R^1 = (H_{n+1}, ..., H_{2n})$
$T = (H_{2n+1}, ..., H_{3n})$		$T = (H_{2n+1}, ..., H_{3n})$
$D = (H_{3n+1}, ..., H_{4n})$		$D = (H_{3n+1}, ..., H_{4n})$
	Fast Phase (for $i = 1$ to n)	
		pick a random bit s_i
		$c_i = s_i$ if $T_i = 1$; $c_i = D_i$ otherwise;
$r_i = R_i^{c_i}$ if $T_i = 1$	$\xleftarrow{c_i}$	start timer
otherwise $r_i = R_0^i$ if $c_i = D_i$		
otherwise r_i = random	$\xrightarrow{r_i}$	stop timer

of the frequency hopping of the reader's carrier and the backscatter nature of passive RFID tags, e.g., their scheme uses the amplitude of the carrier to transmit messages as normal, while utilizing its periodically varied frequency to hide the transmission from the eavesdropper/relayer and exploiting a random sequence modulated to the carrier's phase to defeat malicious modifications.

In all, how to design security schemes relying on the channel randomness, especially how to apply the fruitful theoretic results in wireless physical layer security to RFID settings, is an interesting yet largely open topic that deserves future research.

C. On the Fingerprinting of RFID Tags

The proliferation of wireless technologies has triggered a number of research initiatives to detect illegally operated radio transmitters and identify wireless devices by using physical characteristics of the transmitted signals. These characteristics are usually introduced by imperfections of transceivers caused by manufacturing deviations.

In 2003, Weis already noted in [87] that non-unique IDs can uniquely identify a tag by observing the signal constellation they carry. Danev *et al.* in [31] officially introduced this concept to the cases of identifying ISO14443 tags in a controlled environment, by exploiting the modulation shape and spectral features of the tag. Their experiments show that a set of 50 tags from the same manufacturer and of the same type can be identified with an error rate of around 4%. In WiSec'10, the same group of researchers further investigated this idea and verified a potential impersonation attack in [33], such that the modulation-based identifications, e.g., [31], can be impersonated almost for sure by simply replaying the used features. Similarly, the physical layer identification of UHF RFID tags in compliance with the EPC Gen2 standard has been studied in [91], where the authors have confirmed that the time interval error, i.e., how far each active edge of the clock varies from its ideal position, and the averaged baseband power, i.e., the average power of an acquired RN16 preamble, provide enough entropy such that at most of 2^6 tags can be uniquely identified independently of the population size.

Every blade has two edges – although the obtained results are exciting for the prevention of device cloning, they cause serious privacy issues as well [93]. To be specific, although the tag's digital identity is usually invisible to the rogue scanners thanks to the privacy-preserving protocols as mentioned, unique and fixed physical identities leak the bearer's location information. Zanetti *et al.* in [93] built a fingerprint for clandestine people tracking in a shopping mall, using which the traces of people wearing EPC Gen2 tags can be reconstructed with a high accuracy. Removing or reducing the effect of the random hardware impairments in the analog circuitry components is the only solution besides killing/blocking tags. However, there seems no interest for the manufacturers to produce low-cost tags that are absolutely identical.

There exists other ways to construct radio fingerprints. For instance, the scheme proposed in [47] harvests static identity from existing volatile CMOS memory. Unfortunately, as noticed in [84], RAM is subject to data remanence, which means that after a portion of memory has been used for entropy collection once, it will require a relatively extended period of time without power before it can be reused. In [32], NFC tags are created from a collection of randomly bent, thin conductive wires to serve as certificates to provide authenticity. The underlying idea is that the random distributed wires within the tag manifest special dielectric and/or conductive properties, which could be further sampled/digitized as the tag's fingerprint.

D. Physical Unclonable Function

A Physical Unclonable Functions (PUF) is a function that is embodied in a physical structure, which is easy to evaluate with low-power consumption, but hard to characterize and duplicate. Generally speaking, PUF is not (or should not be) a purely mathematical function, but the use of PUF can be understood mathematically, i.e., the whole process can be seen as a hash function or a pseudo random function $R = f(C, K)$, where C is a physical stimulus, R is the reaction or response from the PUF f and K can be envisioned as the secret key embedded into the device. Note that K, inherited from manufacture deviations, cannot be simply measured or represented. Henceforth, given an instance of a particular PUF, it is hard to physically reproduce it such that the exact functionality f is preserved.

With this interpretation, PUF can be used obviously in three ways: (1) by treating $f(C, K)$ as an ID or a fingerprint of a device under a fixed stimulus C, PUF can be used for identification, e.g., [47]; (2) by treating K as a cryptographic key, PUF can be used to perform challenge-and-response authentications, e.g., [50], ownership transferring, e.g., [59]; and (3) by treating K as a random seed, PUF can be used to produces randomness, i.e., PUF mixes and expends the C and K to a long sequence, e.g., [47].

Considerable applications of PUFs are proposed based on the assumption that PUF, as a primitive, is both efficient and secure.

Unfortunately, this assumption may not be true always since there is no unified framework or rigorous metric to evaluate and analysis the cryptanalytic strength provided by each of the designs. Especially, the lesson people learnt from LFSR, which provides optimum randomness according to [39] and is extremely efficient in hardware, tells that such a function may have a cryptographically simple representation. Therefore, the cryptanalysis of PUFs, as a missing part in the area, is expected to be done in the future. In prior to that, each design of PUF should ship with a reasonable number of input/output samples as a preparation for cryptanalysts.

On the contrary, the actual constructions of PUFs can be very different, e.g., optical PUF, coating PUF, arbiter PUF, ring oscillator PUF, SRAM PUF, butterfly PUF and flip-flop PUF. Among them, the most classic design is a silicon PUF in [38], which exploits the random variations in delays of wires and gates introduced during the circuit fabrication process. Given an input challenge, a race condition is set up in the circuit, and two transitions that propagate along different paths are compared to see which comes first. An arbiter then produces a '1' or a '0', depending on which transition comes first.

V. Conclusion

This paper provides a general overview of the rather broad area of RFID security and privacy and gives the main citations for the comprehensive understanding and further explorations of this area. To be specific, we have investigated the state of the art of the lightweight symmetric ciphers, and we have examined the recent progress on the design and analysis of lightweight authentication protocols, which are expected to provide not only authenticity and computation/storage efficiency but also anonymity, untraceability and other unique properties desired by low-cost RFID applications. Moreover, solutions leveraging electronic characteristics of the tags and the randomness in the channels between the readers and tags have been scrutinized as well, including: distance bounding protocols, channel impairment for good, fingerprinting technologies, physical unclonable functions.

References

[1] G. Avoine, M.A. Bingöl, S. Kardaş, C. Lauradoux, and B. Martin, A framework for analyzing RFID distance bounding protocols, *Journal of Computer Security*, vol. 19, no. 2, pp. 289–317, 2011.

[2] T. Akishita and H. Hiwatari, Very compact hardware implementations of the blockcipher CLEFIA, *Technical Report, available at http://www.sony.net/Products/cryptography/clefia/ download/data/clefia-hw-compact-20110615.pdf*, pp. 1–15, 2010.

[3] J.P. Aumasson, L. Henzen, W. Meier and M. Naya-Plasencia, Quark: a lightweight hash, *Cryptographic Hardware and Embedded Systems, CHES'10*, LNCS 6225, pp. 1–15, 2010.

[4] M. Abdelraheem, G. Leander and E. Zenner, Differential cryptanalysis of round-reduced PRINTCipher: computing roots of permutations, *Fast Software Encryption, FSE'11*, LNCS 6733, pp. 1–17, 2011.

[5] G. Avoine, C. Floerkemeier and B. Martin, RFID distance bounding multistate enhancement, *Progress in Cryptology, Indocrypt'09*, LNCS 5922, pp. 290–307, 2009.

[6] B. Azimi-Sadjadi, A. Kiayias, A. Mercado and B. Yener, Robust key generation from signal envelopes in wireless networks, *In Proceedings of the 14th ACM conference on Computer and Communications Security, CCS'07*, pp. 401–410, 2007.

[7] G. Avoine and P. Oechslin, A scalable and provably secure hash based RFID protocol. *International Workshop on Pervasive Computing and Communication Security, PerCom'05*, pp. 110–114, 2005.

[8] S. Brands and D. Chaum, Distance-bounding protocols, *Advances in Cryptology, EUROCRYPT'93*, LNCS 765, pp. 344–359, 1994.

[9] J. Bringer and H. Chabanne, On the wiretap channel induced by noisy tags, *Security and Privacy in Ad-Hoc and Sensor Networks*, LNCS 4357, pp. 113–120, 2006.

[10] G. Bard, N. Courtois, J. Nakahara, P. Sepehrdad and B. Zhang, Algebraic, AIDA/Cube and side channel analysis of KATAN family of block ciphers, *Progress in Cryptology, Indocrypt'10*, LNCS 6498, pp. 176–196, 2010.

[11] T. Beth and Y. Desmedt, Identification tokens – or: Solving the chess grandmaster problem, *Advances in Cryptology, CRYPT'91*, pp. 169–176, 1991.

[12] M. Burmester, B. De Medeiros and R. Motta, Robust, anonymous RFID authentication with constant key-lookup, *In Proceedings of the 2008 ACM symposium on Information, Computer and Communications Security, CCS'08*, pp. 283–291, 2008.

[13] O. Billet, J. Etrog and H. Gilbert, Lightweight privacy preserving authentication for RFID using a stream cipher, *Fast Software Encryption, FSE'10*, LNCS 6147, pp. 55–74, 2010.

[14] O. Billet and K. Elkhiyaoui, Two attacks against the F_f RFID protocol, *Progress in Cryptology, Indocrypt'09*, LNCS 5922, pp. 308–320, 2009.

[15] A. Biryukov and E. Kushilevitz, Improved cryptanalysis of RC5, *Advances in Cryptology, EUROCRYPT'98*, LNCS 1403, pp. 85–99, 1998.

[16] E.O. Blass, A. Kurmus, R. Molva, G. Noubir and A. Shikfa, The F_f-family of protocols for RFID-privacy and authentication, *Dependable and Secure Computing, IEEE Transactions on*, vol. 8, no. 3, pp. 466–480, 2011.

[17] A. Bogdanov, L. Knudsen, G. Leander, C. Paar, A. Poschmann, M. Robshaw, Y. Seurin and C. Vikkelsoe, PRESENT: An ultra-lightweight block cipher, *Cryptographic Hardware and Embedded Systems, CHES'07*, LNCS 4727, pp. 450-466, 2007.

[18] A. Bogdanov, M. Knežević, G. Leander, D. Toz, K. Varıcı and I. Verbauwhede, SPONGENT: A lightweight hash function, *Cryptographic Hardware and Embedded Systems, CHES'11*, LNCS 6917, pp. 312–325, 2011.

[19] A. Bogdanov, G. Leander, C. Paar, A. Poschmann, M.J.B. Robshaw and Y. Seurin, Hash functions and RFID tags: mind the gap. *Cryptographic Hardware and Embedded Systems, CHES'08*, LNCS 5154, pp. 283–299, 2008.

[20] M. Burmester and J. Munilla, Lightweight RFID authentication with forward and backward security, *Information and System Security, ACM Transactions on*, vol. 14, no, 1, pp. 1–26, 2011.

[21] J. Borst, B. Preneel and J. Vandewalle, Linear cryptanalysis of RC5 and RC6, *Fast Software Encryption, FSE'99*, LNCS 1636, pp. 16–30, 1999.

[22] A. Bogdanov and C. Rechberger, A 3-subset meet-in-the-middle attack: cryptanalysis of the lightweight block cipher KTANTAN, *Selected Areas in Cryptography, SAC'10*, LNCS 6544, pp. 229–240, 2010.

[23] M. Burmester, T. Van Le and B. de Medeiros, Provably secure ubiquitous systems: universally composable RFID authentication protocols, *Conference on Security and Privacy for Emerging Areas in Communication Networks*, pp. 1-9, 2006.

[24] C. Castelluccia and G. Avoine, Noisy tags: a pretty good key exchange protocol for RFID tags, *International Conference on Smart Card Research and Advanced Applications*, LNCS 3928, pp. 289–299, 2006.

[25] S. Capkun, K. El Defrawy and G. Tsudik, Group distance bounding protocols, *In Proceedings of the 4th International Conference on Trust and Trustworthy Computing*, pp. 302–312, 2011.

[26] Q. Chai and G. Gong, BUPLE: securing passive RFID communication through physical layer enhancements, *In Proceedings of Workshop on RFID Security, RFIDSec'11*, LNCS 7055, pp. 127–146, 2012.

[27] J.H. Cheon, J. Hong and G. Tsudik, Reducing RFID reader load with the meet-in-the-middle strategy, *Cryptology ePrint Archive, Report 2009/092*, pp. 1–9, 2009.

[28] H.J. Chae, D.J. Yeager, J.R. Smith and K. Fu, Maximalist cryptography and computation on the WISP UHF RFID tag, *Proceedings of the Conference on RFID Security*, pp. 1–12, 2007.

[29] C. De Canniere, O. Dunkelman and M. Knezevic, KATAN and KTANTAN – a family of small and efficient hardware-oriented block ciphers, *Cryptographic Hardware and Embedded Systems, CHES'09*, LNCS 5747, pp. 272–288, 2009.

[30] I. Dinur, O. Dunkelman and A. Shamir, Improved attacks on full GOST, *Cryptology ePrint Archive, Report 2011/558*, pp. 1–25, 2011.

[31] B. Danev, T.S. Heydt-Benjamin and S. Capkun, Physical-layer identifi-

cation of RFID devices, *In Proceedings of 18th conference on USENIX security Symposium, USENIX'09*, pp. 199–214, 2009.

[32] G. DeJean and D. Kirovski, RF-DNA: radio-frequency certificates of authenticity, *Cryptographic Hardware and Embedded Systems, CHES'07*, LNCS 4727, pp. 346–363, 2007.

[33] B. Danev, H. Luecken, S. Capkun and K. El Defrawy, Attacks on physical-layer identification, *In Proceedings of the third ACM conference on Wireless network Security, WiSec'10*, pp. 89–98, 2010.

[34] S. Drimer and S.J. Murdoch, Keep your enemies close: distance bounding against smartcard relay attacks, *In Proceedings of 16th USENIX Security Symposium, USENIX'07*, pp. 1–16, 2007.

[35] D. Engels, M.J.O. Saarinen and E. Smith, The Hummingbird-2 lightweight authenticated encryption algorithm, *In Proceedings of Workshop on RFID Security, RFIDSec'11*, LNCS 7055, pp. 19–31, 2011.

[36] M. Feldhofer, J. Wolkerstorfer and V. Rijmen, AES implementation on a grain of sand, *Information Security, IEE Proceedings*, vol. 152, no. 1, pp. 13–20, 2005.

[37] T. Good and M. Benaissa, Hardware results for selected stream cipher candidates, *In Proceedings of the State of the Art of Stream Ciphers, SASC'07*, pp. 191–204, 2007.

[38] B. Gassend, D. Clarke, M. Van Dijk and S. Devadas, Silicon physical random functions, *In Proceedings of the 9th ACM conference on Computer and Communications Security, CCS'02*, pp. 148–160, 2002.

[39] S.W. Golomb and G. Gong, *Signal design with good correlation: for wireless communications, cryptography and radar applications*, Cambridge University Press, 2005.

[40] Z. Gong, S. Nikova and Y.W. Law, Klein: a new family of lightweight block ciphers, *In Proceedings of Workshop on RFID Security, RFIDSec'11*, LNCS 7055, pp. 1–18, 2011.

[41] J. Guo, T. Peyrin and A. Poschmann, The PHOTON family of lightweight hash functions, *Advances in Cryptology, CRYPTO'11*, LNCS 6841, pp. 222–239, 2011.

[42] J. Guo, T. Peyrin, A. Poschmann and M. Robshaw, The LED block cipher, *Cryptographic Hardware and Embedded Systems, CHES'11*, LNCS 6917, pp. 326–341, 2011.

[43] H. Gilbert, M.J. Robshaw and Y. Seurin, Good variants of HB^+ are hard to find, *Financial Cryptography and Data Security, FC'08*, LNCS 5143, pp. 156–170, 2008.

[44] H. Gilbert, M.J.B. Robshaw and Y. Seurin, $HB^\#$: Increasing the security and efficiency of HB^+, *Advances in Cryptology, EUROCRYPT'08*, LNCS 4965, pp. 361–378, 2008.

[45] H. Gilbert, M. Robshaw and H. Sibert, An active attack against HB^+ – a provably secure lightweight authentication protocol, *IEE Electronic Letters*, vol. 41, no. 21, pp. 1169–1170, 2005.

[46] G. Hancke, Design of a secure distance-bounding channel for RFID, *Journal of Network and Computer Applications*, vol. 34, no. 3, pp. 1–11, 2011.

[47] D.E. Holcomb, W.P. Burleson and K. Fu, Power-up SRAM state as an identifying fingerprint and source of true random numbers, *Computers, IEEE Transactions on*, vol. 58, no. 9, pp. 1198–1210, 2009.

[48] G. Hancke and M.G. Kuhn, Attacks on time-of-flight distance bounding channels, *In Proceedings of the first ACM conference on Wireless network Security, WiSec'08*, pp. 194–202, 2008.

[49] G. Hancke and M.G. Kuhn, An RFID distance bounding protocol, *First International Conference on Security and Privacy for Emerging Areas in Communications Networks, SecureComm'05*, pp. 67–73, 2005.

[50] G. Hammouri and B. Sunar, PUF-HB: A tamper-resilient HB based authentication protocol, *Applied Cryptography and Network Security, ACNS'08*, LNCS 5037, pp. 346–365, 2008.

[51] T. Isobe, A single-key attack on the full GOST block cipher, *Fast Software Encryption, FSE'11*, LNCS 6733, pp. 290–305, 2011.

[52] A. Juels, RFID security and privacy: a research survey, *Selected Areas in Communications, IEEE Journal on*, vol. 24, No. 2, pp. 381–394, 2006.

[53] A. Juels, R.L. Rivest and M. Szydlo, The blocker tag: selective blocking of RFID tags for consumer privacy, *In Proceedings of the 10th ACM Conference on Computer and Communication Security, CCS'03*, pp. 103–111, 2003.

[54] A. Juels and S.A. Weis, Authenticating pervasive devices with human protocols, *Advances in Cryptology, CRYPTO'05*, LNCS 3621, pp. 293–308, 2005.

[55] J.P. Kaps, Chai-tea, cryptographic hardware implementations of xTEA, *Progress in Cryptology, Indocrypt'08*, LNCS 5365, pp. 363–375, 2008.

[56] C. Kim and G. Avoine, RFID distance bounding protocol with mixed challenges to prevent relay attacks, *Conference on Cryptology and Network Security, CANS'09*, LNCS 5888, pp. 119–133, 2009.

[57] L. Knudsen, G. Leander, A. Poschmann and M. Robshaw, PRINTCipher: a block cipher for IC-printing, *Cryptographic Hardware and Embedded Systems, CHES'10*, LNCS 6225, pp. 16–32, 2010.

[58] S. Knellwolf, W. Meier and M. Naya-Plasencia, Conditional differential cryptanalysis of Trivium and KATAN, *In Proceedings of Selected Areas in Cryptography, SAC'11*, LNCS 7118, pp. 200–212, 2011.

[59] L. Kulseng, Z. Yu, Y. Wei and Y. Guan, Lightweight mutual authentication and ownership transfer for RFID systems, *In Proceedings of the 29th conference on Information Communications, INFOCOM'10*, pp. 1–5, 2010.

[60] Z. Li, G. Gong and Z. Qin, Secure and efficient LCMQ entity authentication protocol, *Technical Report, CACR 2010-21, University of Waterloo*, pp. 1–24, 2010.

[61] G. Leander, M.A. Abdelraheem, H. AlKhzaimi and E. Zenner, A cryptanalysis of PRINTCipher: the invariant subspace attack, *Advances in Cryptology, CRYPTO'11*, LNCS 6841, pp. 206–221, 2011.

[62] C. Lim and T. Korkishko, mCrypton–a lightweight block cipher for security of low-cost RFID tags and sensors, *Information Security Applications*, LNCS 3786, pp. 243–258, 2006.

[63] B. Liang, Y. Li, C. Ma, T. Li and R. Deng, On the untraceability of anonymous RFID authentication protocol with constant key-lookup, *Information Systems Security*, pp. 71–85, 2009.

[64] G. Leander, C. Paar, A. Poschmann and K. Schramm, New lightweight DES variants. *Fast Software Encryption, FSE'07*, LNCS 4593, pp. 196–210, 2007.

[65] M. Nele, G. Jan, B. Preneel and I. Verbauwhede, A low-cost implementation of Trivium, *In Proceedings of the State of the Art of Stream Ciphers, SASC'08*, pp. 197–204, 2008.

[66] M. Madhavan, A. Thangaraj, Y. Sankarasubramanian and K. Viswanathan, NLHB: a nonlinear hopper-blum protocol, *IEEE International Symposium on Information Theory Proceedings, ISIT'10*, pp. 2498–2502, 2010.

[67] D. Molnar and D. Wagner, Privacy and security in library RFID: issues, practices, and architectures. *In Proceedings of the 11th conference on Computer and Communications Security, CCS'04*, pp. 210–219, 2004.

[68] Y. Oren and M. Feldhofer, A low-resource public-key identification scheme for RFID tags and sensor nodes, *In Proceedings of the second ACM conference on Wireless network Security, WiSec'09*, pp. 59–68, 2009.

[69] K. Ouafi, R. Overbeck and S. Vaudenay, On the security of $HB^\#$ against a Man-in-the-Middle attack, *Advances in Cryptology, AsiaCrypt'08*, LNCS 5350, pp. 108–124, 2008.

[70] M. Ohkubo, K. Suzuki, S. Kinoshita and others, Cryptographic approach to privacy-friendly tags, *RFID Privacy Workshop*, vol. 82, pp. 1–9, 2003.

[71] K. Ouafi and S. Vaudenay, Smashing SQUASH-0, *Advances in Cryptology, EUROCRYPT'09*, LNCS 5479, pp. 300–312, 2009.

[72] A. Poschmann, S. Ling and H. Wang, 256 bit standardized crypto for 650 GE–GOST revisited, *Cryptographic Hardware and Embedded Systems, CHES'10*, LNCS 6225, pp. 219–233, 2011.

[73] C. Pendl, M. Pelnar and M. Hutter, Elliptic curve cryptography on the WISP UHF RFID tag, *In Proceedings of RFIDSec'11*. LNCS 7055, pp. 32–47, 2011.

[74] R. Rivest, The RC5 encryption algorithm, *Fast Software Encryption, FSE'95*, LNCS 1008, pp. 86–96, 1995.

[75] K.B. Rasmussen and S. Capkun, Location privacy of distance bounding protocols, *In Proceedings of the 15th ACM conference on Computer and Communications Security, CCS'08*, pp. 149–160, 2008.

[76] K.B. Rasmussen and S. Capkun, Realization of RF distance bounding, *In Proceedings of the USENIX Security Symposium, USENIX'10*, pp. 1–13, 2010.

[77] M. Rieback, B. Crispo and A. Tanenbaum, RFID guardian: A battery-powered mobile device for RFID privacy management, *Australasian*

Conference on Information Security and Privacy, ACISP'05, LNCS 3574, pp. 184–194, 2005.

[78] C. Rolfes, A. Poschmann, G. Leander and C. Paar, Ultra-lightweight implementations for smart devices–security for 1000 gate equivalents. *In Proceedings of the 8th IFIP WG 8.8/11.2 International Conference on Smart Card Research and Advanced Applications*, LNCS 5189, pp. 89–103, 2008.

[79] A. Shamir, Memory efficient variants of public-key schemes for smart card applications, *Advances in Cryptology, EUROCRYPT'94*, LNCS 950, pp. 445–449, 1995.

[80] A. Shamir, SQUASH – a new MAC with provable security properties for highly constrained devices such as RFID tags, *Fast Software Encryption, FSE'08*, LNCS 5086, pp. 144–157, 2008.

[81] K. Shibutani, T. Isobe, H. Hiwatari, A. Mitsuda, T. Akishita and T. Shirai, Piccolo: an ultra-lightweight blockcipher, *Cryptographic Hardware and Embedded Systems, CHES'11*, LNCS 6917, pp. 342–357, 2011.

[82] B. Song and C.J. Mitchell, RFID authentication protocol for low-cost tags, *In Proceedings of the first ACM conference on Wireless network Security, WiSec'08*, pp. 140–147, 2008.

[83] O. Savry, F. Pebay-Peyroula, F. Dehmas, G. Robert and J. Reverdy, RFID noisy reader: how to prevent from eavesdropping on the communication? *Cryptographic Hardware and Embedded Systems, CHES'07*, LNCS 4727, pp. 334–345, 2007.

[84] N. Saxena and J. Voris, We can remember it for you wholesale: implications of data remanence on the use of RAM for true random number generation on RFID tags, *Workshop on RFID Security, RFIDSec'09*, pp. 1–13, 2009.

[85] G. Tsudik, YA-TRAP: yet another trivial RFID authentication protocol, *International Conference on Pervasive Computing and Communications, Percom'06*, pp. 640–643, 2006.

[86] R. Trujillo-Rasua, B. Martin and G. Avoine, The Poulidor distance-bounding protocol, *Workshop on RFID Security, RFIDSec'10*, LNCS 6370, pp. 239–257, 2010.

[87] S.A. Weis, Security and privacy in radio-frequency identification devices, *Master Thesis, Massachusetts Institute of Technology*, pp. 49–51, 2003.

[88] A.D. Wyner, The wire-tap channel, *Bell Systems Technical Journal*, vol. 54, pp. 1355–1387, 1975.

[89] J. Wu and D.R. Stinson, How to improve security and reduce hardware demands of the WIPR RFID protocol, *IEEE International Conference on RFID, RFID'09*, pp. 192–199, 2009.

[90] D.H. Yum, J.S. Kim, S.J. Hong and P.J. Lee, Distance bounding protocol for mutual authentication, *Wireless Communications, IEEE Transactions on*, vol. 10, no. 2, pp. 592–601, 2011.

[91] D. Zanetti, B. Danev and S. Capkun, Physical-layer identification of UHF RFID tags, *In Proceedings of the sixteenth annual International Conference on Mobile Computing and Networking, MobiCom'10*, pp. 353–364, 2010.

[92] B. Zhu and G. Gong, Guess-then-meet-in-the-middle attacks on the KTANTAN family of block ciphers, *Cryptology ePrint Archive, Report 2011/619*, pp. 1–14, 2011.

[93] D. Zanetti, P. Sachs and S. Capkun, On the practicality of UHF RFID fingerprinting: how real is the RFID tracking problem, *Privacy Enhancing Technologies, PET'11*, LNCS 6794, pp. 97–116, 2011.