Low Cost Countermeasure at Authentication Protocol Level against Electromagnetic Side Channel Attacks on RFID Tags

Yassine NAIJA[1,2,3], Vincent BEROULLE[2], Mohsen MACHHOUT[3]
[1]University of Sousse, ENISO Sousse, Tunisia
[2]University of Grenoble Alpes, LCIS, F-26000 Valence, France
[3]University of Monastir, EμE (FSM) Monastir, Tunisia

***Abstract*—Radio Frequency Identification (RFID) technology is widely spread in many security applications. Producing secured low-cost and low-power RFID tags is a challenge. The used of lightweight encryption algorithms can be an economic solution for these RFID security applications. This article proposes low cost countermeasure to secure RFID tags against Electromagnetic Side Channel Attacks (EMA). Firstly, we proposed a parallel architecture of PRESENT block cipher that represents a one way of hiding countermeasures against EMA. 200 000 Electromagnetic traces are used to attack the proposed architecture, whereas 10 000 EM traces are used to attack an existing serial architecture of PRESENT. Then we proposed a countermeasure at mutual authentication protocol by limiting progressively the number of EM traces. This limitation prevents the attacker to perform the EMA. The proposed countermeasure is based on time delay function. It requires 960 GEs and represents a low cost solution compared to existing countermeasures at primitive block cipher (2471 GEs).**

Keywords*—Radio Frequency Identification (RFID); electromagnetic side channel attack; PRESENT; mutual authentication protocol; countermeasures

I. INTRODUCTION

Passive RFID tag consists of an integrated circuit (IC) attached to an antenna. This integrated circuit is entirely remotely powered from the RF reader. Contactless RFID tags are used in different security applications such as access control and contactless payment systems. For example, among the commercial HF tags that implement cryptographic functions for the authentication protocol, there are MIFARE Ultralight C [2] and MIFARE DESFire EV1 [3] integrating 3DES [4] and AES [5] block cipher circuits, respectively. The mutual authentication protocol implemented in these tags is based on the symmetric challenge-response technique. In addition, in an academic context, Feldhofer et al. [21], [22] presented a strong authentication scheme, also using a symmetric challenge-response technique, based on an AES algorithm for RFID systems. The protocols for these symmetric challenge-response techniques based on encryption are defined in the ISO/IEC 9798-2 standard [27].

Strong cryptographic algorithms, such as AES and 3DES are often too expensive in terms of area and power [6] and are used for applications requiring high level of security. In other hand, many works suggest the implementation of lightweight block ciphers, such as SIMON/ SPECK [7], HIGHT [8], XTEA [9], PRESENT [10], KATAN/KTANTAN [11], PRINCE [12], TWINE [13] and CRYPTON [14]. These lightweight block ciphers satisfy the security needs of some low level of security RFID applications such as access control, ticketing, etc. Indeed, for resource limited embedded systems, it is important to use an adapted level of security (often related to the number of bits of the secret key) in order to reduce both hardware overhead and power consumption. For example, Sai Seshabhattar et al. [15] proposed an implementation of PRESENT in EPC Class1 Gen2 protocol for UHF RFID tags. They implemented a low cost mutual authentication protocol based on encryption operations in the tag and decryption operations in the reader. On the other hand, Naija Yassine et al. [16] proposed a HF tag architecture respecting the IEC/ISO 14443 Type A [1]. This architecture is based on the implementation of the PRESENT block cipher in Mifare Ultralight C mutual authentication protocol.

Side Channel Attacks (SCA) represents a serious threat for RFID tags. SCA are non-invasive attacks and are based on the observation during the execution of the cryptographic devices of physical phenomena such as response time [17], power consumption [18] or electromagnetic radiation [19]. In this article, we focus our study to the Electromagnetic Side Channel Attack (EMA). For example, Timo Kasper et al. attacked some Mifare products (Mifare Desfire, Mifare MF3ICD40 and Mifare Classic) using EMA [28], [29]. These products implement mutual authentication protocols vulnerable to EMA.

This article proposes a low cost countermeasure at the authentication protocol level by limiting the number of successive wrong authentication requests. This limitation prevents the attacker to save enough electromagnetic traces to perform the EMA. First, we choose to study the vulnerability of an existing mutual authentication protocol proposed by Sai Seshabhattar et al. [15] against EMA. This protocol integrating PRESENT block cipher is used for low cost full-fledged RFID tags. Then, we proposed a parallel implementation of PRESENT in order to hide the information leakage (electromagnetic radiation) generated by its S-box function. The EMA is performed in our proposed PRESENT architecture and compared to existing work [20] (serial architecture). Finally, a countermeasure based on time delay function is proposed to delay the response of the tag (especially the

encryption operation) for each wrong authentication. This time delay function allows the tag to enter in killed state progressively and prevents the EMA.

This article is organized as follows. Section II describes the Seshabhatta et al. protocol and explains its vulnerability against EMA. Section III describes the PRESENT algorithm and our parallel implementation. Section IV is devoted to the description of the EMA methodology on PRESENT, EM attack setup and EM attack results and comparison. The countermeasure at protocol level based on time delay function is proposed in Section V. Finally, we conclude the paper in Section VI.

II. Authentication Protocol Description and EMA Vulnerability

Mutual authentication protocols (ISO/IEC 9798-2 [27]) in RFID communication ensure the authentication of both readers and tags. This authentication phase prevents the attacker to impersonate the identity of the tag. However, several passive attacks such as EMA can be a threat to recover the secret parameters of the tag. In this section, we describe the Seshabhatta et al. protocol used for the UHF tags and its vulnerability against EMA in the aim to propose security solutions to overcome this attack.

A. Mutual Authentication Protocol Description

Seshabhatta et al. proposed [15] the integration of two security levels to secure the EPC GEN2 communication between a tag and a reader. The level1 is represented in the secure identification phase that allows the security of the tag identity, whereas the level2 is represented in the mutual authentication protocol that allows to ensure the authenticity of the reader and the tag. In the following, we name the Seshabhatta et al. protocol the ProtocolS. ProtocolS as shown in Fig. 1 consist of five steps roughly described as follows:

- Step (1): Reader sends the request command to start the authentication phase.
- Step (2): Tag generates a 8-byte random number PT1. It replies with PT1.
- Step (3): Reader generates a 8-byte random number PT2. It decrypts PT1 and decrypts PT2 with the key related to the tag ID and then concatenates and sends the results. It replies with Challenge = Dk (PT1) || Dk (PT2).
- Step (4): Tag encrypts the Challenge to get CT1= Ek (Challenge (127 down to 64)) || Ek (Challenge (63 down to 0)). It generates a 8-byte random number PT3. It compares CT1 (127 down to 64) to PT1. If they match, the reader is authenticated. Then, the tag replies with Response = Ek (PT3) || Ek (CT1 (63 down to 0)).
- Step (5): Reader decrypts the Response to get PT4 = Dk (Response (127 down to 64)) || Dk (Response (63 down to 0)). If PT4 (63 down to 0) = PT2 then the tag is authenticated.

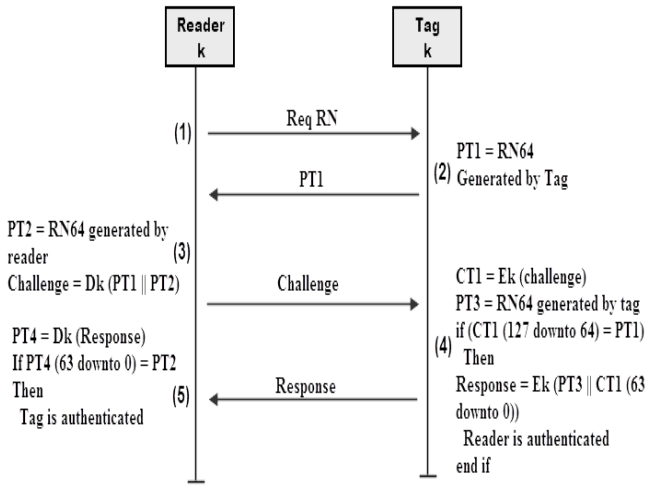

Fig. 1. Seshabhatta et al. protocol (ProtocolS) [15].

B. Protocols Vulnerability against EMA

The ProtocolS is vulnerable against EMA in Step (4). A malicious reader can send wrong challenges to tag. Even though the tag does not respond to these wrong challenges, the attacker obtains the information leakage of the block cipher during the encryption operation. For example, Timo Kasper et al. proposed a technique [28] to save the electromagnetic radiations generated by the Mifare Desfire block cipher. The technique is based on analog demodulator and filters that allows bypassing the influence of the reader field by removing the unwanted carrier frequency. We suppose that we are in Timo Kasper et al. conditions. The ProtocolS integrating unprotected PRESENT block cipher can be attacked by EMA. In Step (3), the attacker can send a 128-bit random challenge. As indicated in (4), the tag encrypts the MSB 64-bit of each received challenge and compares the result with the generated PT1. During the encryption operation of the challenge the attacker can exploit the electromagnetic radiation of the PRESENT block cipher.

In the following, we will propose a parallel architecture of PRESENT in order to test its vulnerability against EMA and compared it (number of EM traces to obtain the key) with an existing unprotected serial PRESENT architecture. A description of PRESENT and its hardware implementation is shown in the next section.

III. PRESENT-80 Block Cipher

A. PRESENT Description

PRESENT is an ultra-lightweight block cipher proposed by A. Bogdanov et al. [10]. It has been designed for secured low power and low area devices such as passive RFID tags. It has a block size of 64-bit and two key lengths of 80 (PRESENT-80) and 128-bit (PRESENT-128) are supported. We chose the implementation of PRESENT-80 bit rather than PRESENT-128 bit because the first one showed a lower area [29]. The algorithm of PRESENT-80 is shown in Fig. 2.

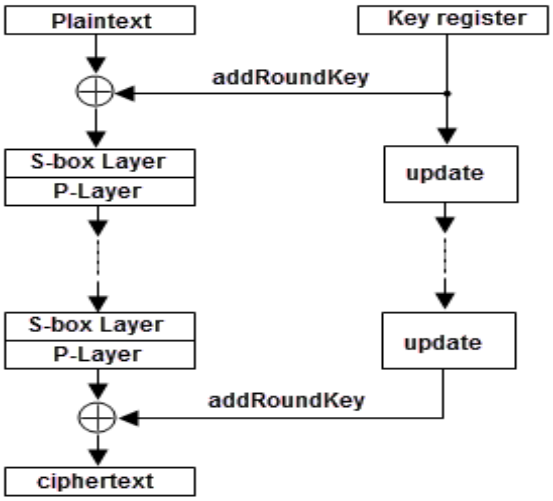

Fig. 2. PRESENT algorithm.

It consists of 31-rounds Substitution-Permutation (SP) network and a final key-whitening, during which:

- Round key is added to plaintext.
- Plaintext goes through S-boxes (substitution boxes).
- Plaintext after S-boxes goes through P-Layer (permutation layer).
- Round key is updated.

The result of the key updater operation for every round is taken as a round key and it is added to the current state b63 b0. This operation is performed as shown below:

$$bj \leftarrow bj \oplus k^{i}j \quad \text{where } 1 \le i \le 32 \quad \text{and} \quad 0 \le j \le 63 \qquad (1)$$

Where, i is the round in processing and j is the bit position.

The second stage is a non-linear S-box Layer that consists of 4-bit to 4-bit S-boxes, which are given in hexadecimal notation in Table 1.

TABLE I. PRESENT S-BOX FUNCTION

x	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
S[x]	C	5	6	B	9	0	A	D	3	E	F	8	4	7	1	2

The permutation layer of PRESENT is the third stage of the round operation. It is a linear bit permutation and it is described in (2), (3), (4) and (5).

$$\text{For } 0 \le i \le 15$$

$$bi \leftarrow b4\times I \qquad (2)$$

$$bi+16 \leftarrow b4\times i+1 \qquad (3)$$

$$bi+32 \leftarrow b4\times i+2 \qquad (4)$$

$$bi+48 \leftarrow b4\times i+3 \qquad (5)$$

The key updater process operates on the user supplied 80-bit key and outputs a 64-bit key for every round. The user-supplied key is stored in a key register K and represented as k79k78…k1k0. For the round i, the left most 64 bits of the current state of register K are the round key. Thus we have:

$$Ki = k63k62 \,.\, k1k0 = k79k78 \,.\, k17k16 \qquad (6)$$

After the round key Ki is extracted, the key register K = k79k78 . . . k1k0 is updated as follows:

1. [k79k78 . k1k0] = [k18k17 . k20k19] (bitwise rotation) (7)
2. [k79k78k77k76] = S [k79k78k77k76] (8)
3. [k19k18k17k16k15] = [k19k18k17k16k15] $\oplus$ round counter (9)

C. PRESENT-80 Implementation

There are many implementations of PRESENT-80 algorithm. For example, Axel Poschmann et al. proposed a serial implementation of the PRESENT algorithm [20]. This unprotected implementation (4-bit data path) requires 1100 gate equivalents (GEs) and 547 clock cycles to process one block of data. Generally, the more parallel level of the data path, the harder it is to attack (Side Channel Attack) the design because parallelism is one way of hiding countermeasures. For this reason, we proposed a parallel architecture of PRESENT in the aim to evaluate its vulnerability against EMA and compared its attack results with serial PRESENT architecture.

Our proposed PRESENT-80 implementation given in Fig. 3 is based on a parallel hardware processing rather than a sequential processing. This parallel architecture is based on 64-bit data bath. It means the 16 S-box blocks operates at the same time which makes saving electromagnetic traces corresponding to one S-box operation is very difficult. The attack setup will be presented in details in the next section. Our PRESENT version has two inputs (data-in, key) and one output (data-out). The data-in and data-out are both on 64-bit and the key is on 80-bit. The architecture consists of two MUXs, one XOR, two 64-bit registers Reg1 and Reg2, 16 4-bit S-boxes, 64-bit shift-register (permutation layer operator), 80-bit key update and 5-bit counter.

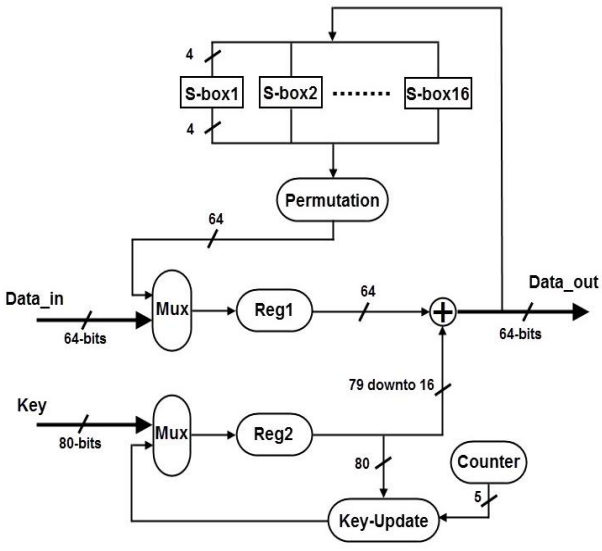

Fig. 3. Hardware architecture of PRESENT-80.

Due to the parallelism of our implementation, one round requires only one clock cycle to substitute the data (S-box), to perform the data permutation and the key updater. Including the initialization phase, 33 clock cycles are required to process one block of data. The synthesis of PRESENT-80 on an ASIC has been done with Leonardo Spectrum from Mentor Graphics using the scl05u library (without optimization). Our architecture requires about 2050 GEs and 33 clock cycles to process one block of data. This implementation is not the best in term of area compared to [20]. However, it is more secure against EMA (see next section).

IV. EM Attack on Present

Until this section, we only present EMA on commercial tags (with a chip based on an ASIC). However, our architecture and its countermeasure will be validated on a FPGA platform. The EMA can also be performed on FPGA that implements the digital tag architecture. The evaluation of EMA performed on FPGA platform is generally considered as realistic. In fact, the exploitation of the extracted information leakage on FPGA is generally also possible once the architecture is implemented on ASIC technology. We chose to implement our parallel architecture of PRESENT in a SAKURA-G starter board to perform EMA. The EM attack methodology on the PRESENT block cipher is presented in order to recover the key. Then, we calculate the attack setup time that depends to the saved electromagnetic traces. Finally, we compare our attack results with Axel Poschmann et al. results [20].

A. EM Attack Methodology on PRESENT

The first DPA (Differential Power Analysis) attack based on the analysis of power consumption has been proposed by P. Kocher in 1999 [18]. The Electromagnetic attack uses the same hypothetical model but using the EM radiations rather than the power consumption. The EM radiations measured with a near field probe are often less noisy than the global circuit power consumption signal. In this work, EMA uses the CPA (Correlation Power Analysis) [23] between the radiations emitted by the encryption circuit and a hypothetical model.

First, we start by locating the best attack point. For PRESENT, this point is the output of the nonlinear S-box function (see Section III-A). This point is chosen because the secret key (ks) information is contained in the power consumption of the circuit when performing the S-box operations. We model the dynamic power of the output of the S-box operation Pdyn_Sbox as follows with the Hamming Weight (HW) function:

$$Pdyn\_Sbox = HW\ (S\text{-}box\ (PT \oplus ks)) \quad (10)$$

The presence of the secret key in the power consumption will be exploited by the EM attack. Once the attack point is identified, the EM attack on PRESENT can be realized. Fig. 4 shows the different steps of the EM attack:

- Plain texts of 64-bit are randomly generated and encrypted by the PRESENT block cipher. During each of those encryptions, the electromagnetic emissions of the chip, as well as plain texts sent to the circuit are recorded.
- The PRESENT secret key on 80-bit is divided into 20 4-bit wide sub-keys. The MSB 16 sub-keys (k79 k78 k16) are recovered in the first round of the encryption operation and the LSB 4 sub-keys (k15 k14 … k0) are recovered in the second round of the encryption operation (see, (6), (7), (8) and (9)).
- The PRESENT architecture previously described shows that the random input data and the key are XORed 4-bit to 4-bit and fed out to the non-linear S-box function. The output of each S-box is on 4-bit. These S-box outputs are the locations of ours attacks. Each attack location allows us to recover one sub-key (4 bits).
- To recover each sub-key, we calculate the Pearson Correlation [23] between the output of the attack model (HW) and the real traces. For each sub-key hypothesis, we obtain for each EM trace oscilloscope sample, a correlation value.
- A comparison is performed between the correlations for all hypothetical sub-keys, and the correlation with the highest amplitude corresponds to the value of the right sub-key. The attack setup will be described in detail in Section IV-C.

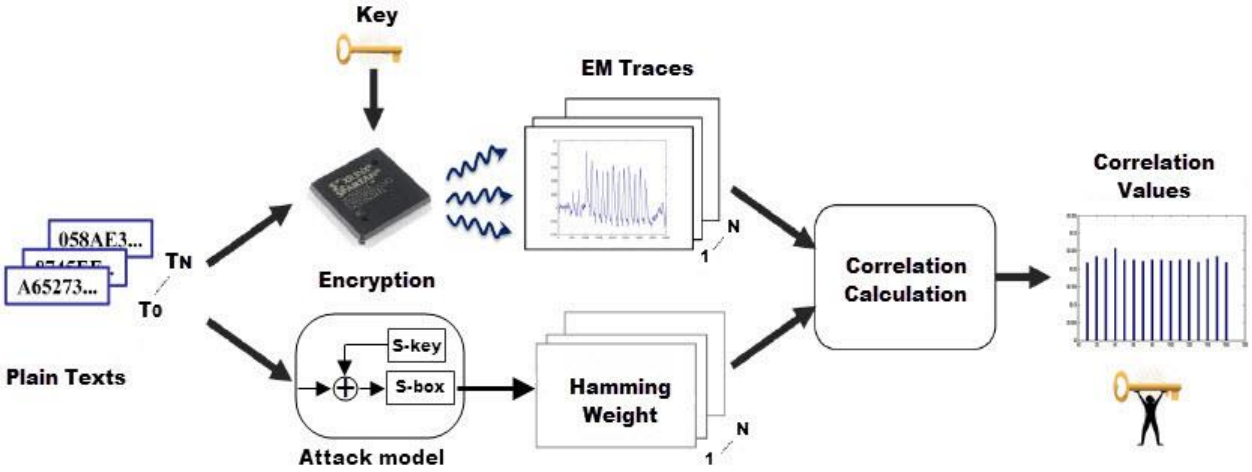

Fig. 4. EM attack methodology.

B. Measurement Set-Up

In order to perform the EM attack, the PRESENT encryption unit has been implemented on a SAKURA-G [24] starter board containing a Spartan 6 FPGA (XC6SLX75). Electromagnetic radiations during the encryption operation are measured using a near field probe RF-U5-2 [25] and a Wave Runner 6 Zi oscilloscope features 400 MHz - 4 GHz of bandwidth and 40 GS/s sampling rate [26]. We used also a XY table to control the placement of the EM probe on the FPGA surface to find the best point to make the attack. Fig. 5 shows the electromagnetic measurement bench to perform the EM attack.

The interconnection of the oscilloscope to the FPGA platform is performed by two cables. The first cable is connected to one pin of the User Header Pin (in/out logic pins) to detect the trigger signal coming from the FPGA to trig the oscilloscope sampling. The trigger signal is coming from the encryption design and appears in every first round to save the EM traces. The second cable is connected to the near field probe to visualize the Electromagnetic radiation of the encryption block.

C. Attack Setup

The controlling design shown in Fig. 6 has been built for carrying out the functioning of the PRESENT unit cipher. This design also contains a Linear Feedback Shift Register (LFSR) that allows us to generate random plain texts to feed the encryption unit. In addition, we use a frequency divider block to transform the FPGA frequency from 48 MHz to 100 kHz. Indeed, this 100 kHz frequency is a widely used as operating frequency in RFID tags. A controller block is implemented to control LFSR and PRESENT blocks.

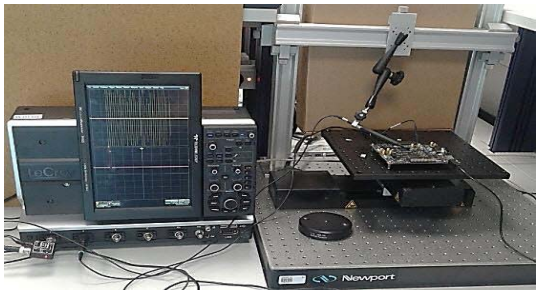

Fig. 5. Electromagnetic measurement bench.

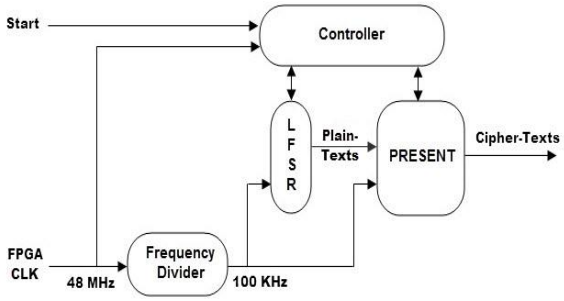

Fig. 6. Design architecture controlling the PRESENT block cipher during EMA.

To synchronize the generation of the plain text with the sampling of its corresponding electromagnetic radiation, we add a delay state in the finite state machine of the controller. This delay is necessary to give more time for the oscilloscope to perform the storage of the EM trace (0.5 ms). When the attack is performed at the first or the second round of PRESENT, for each trigger signal event the oscilloscope saves one EM radiation trace. A matrix T of 100 000 plaintexts is used as PRESENT encryption inputs and thus a matrix M comprising 100 000 Electromagnetic traces is obtained. The same 100 000 plaintexts and their relative 100 000 EM traces are used to recover the 16 MSB 4-bits sub-keys. Each of these traces consists of 4002 oscilloscope samples.

In this work, we use the previous EM attack model: the Hamming Weight (HW) at the S-box (10). This attack model was developed with Matlab. The first step performs the attack at the first round of the encryption unit where we are able to recover 16 MSB 4-bits sub-keys. After recovering these 16 sub-keys, the second step performs the attack at the second round of PRESENT to get the last 4 sub-keys. To predict one sub-key in the step one of the attack, the attack model input (plaintext) is a matrix T of 4-bits vectors of dimension (100000 × 1). This matrix is XORed with the 16 possible 4-bits sub-keys to get a matrix with dimension of (100000 × 16). As we mentioned before, the output of the logic gate XOR is fed out to the attack S-box function and the output of this S-box is a matrix of dimension (100000 × 16). Using the same 100 000 traces, permits doing all the attacks on all the S-box functions. For each attack, the attack model input changes but the EM traces remain the same 100 000 traces. At this stage, if we apply the Hamming Weight model, we must calculate the HW of each S-box 4-bits vector to get the H_W matrix of dimension (100000 × 16). The last step of the attack is to calculate the correlations between the real traces, which is a matrix of dimension (100000 × 4002), and the H_W matrix to get a CORL matrix of dimension (4002 × 16). The correlation with the highest value corresponds to the recovered sub-key.

After recovering the MSB 64-bits of the secret key, in the second step of the attack we recover the last 4 sub-keys. We keep the same attack model but the inputs are the cipher outputs data of the first round of the algorithm instead of the random data generated by our LFSR. Also we save 100 000 electromagnetic traces corresponding to the second round of PRESENT. As it was mentioned in the PRESENT algorithm description, in each encryption round the key must be updated (see Section III-A). So when we recover the 64-bits of the key updated used in the second round, we can recover the initial LSB portion key (k15 k14… k1k0) by the use of the key updater reverse operation (see, (7), (8) and (9)).

D. EM Attack Results and Comparison

This section shows the EM attack results on PRESENT. Fig. 7 is a sample of an electromagnetic radiation trace of the block cipher saved during the encryption operation. In Fig. 7 every round of PRESENT is associated with a voltage peak. Therefore, there are 32 voltage peaks.

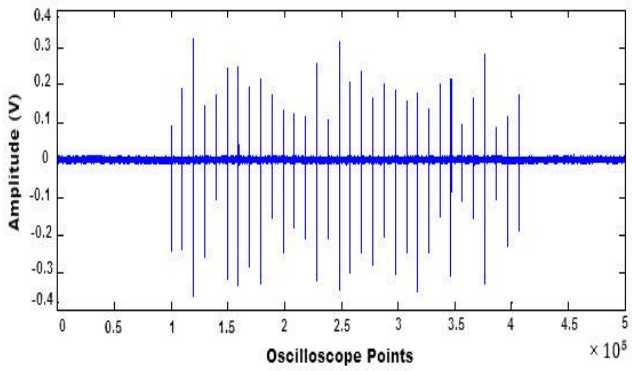

Fig. 7. Electromagnetic trace observe during the encryption operation.

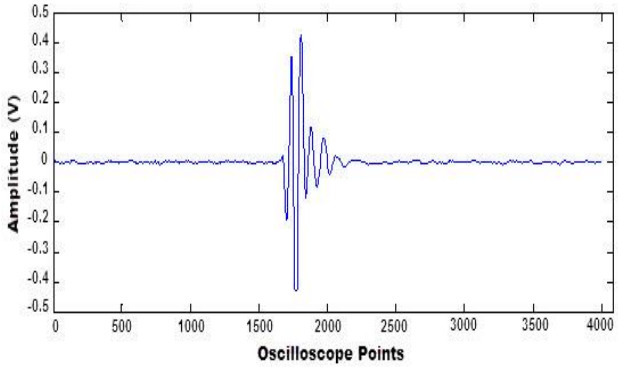

Fig. 8. Electromagnetic trace of the first round.

Fig. 8 shows the electromagnetic trace saved at the first round.

In order to extract information from the leakage resources, an EM attack was performed using the power model based on the HW of the S-box outputs. As we have mentioned in Section IV-C, during the first step of the attack we are able to recover the MSB 64-bit of the key. This first step of the attack is performed in the first round of the PRESENT algorithm. The experimental results show that after the encryption of 100 000 plaintexts (corresponding to the sampling of 100 000 EM traces), we achieve to recover all the 16 sub-keys. Fig. 9 shows the correlations according to the oscilloscope points. These correlation values are extracted from the CORL correlation matrix (4002 x 16) previously described in Section IV-C.

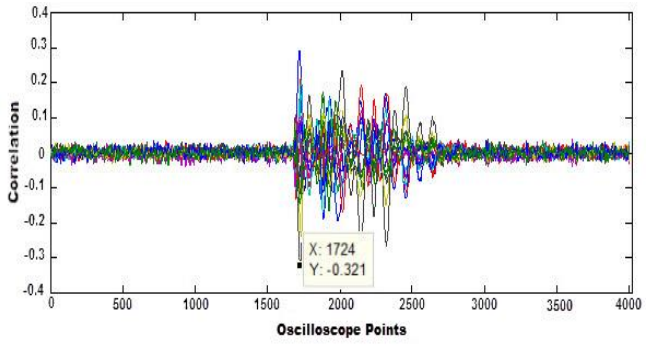

Fig. 9. EM attack correlations using the HW model.

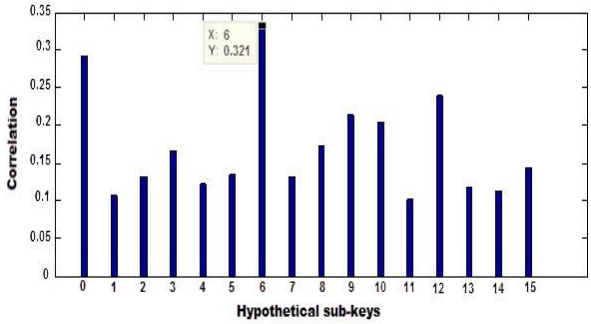

Fig. 10. Example of calculated correlations for one 4-bit PRESENT hypothetical sub-key.

In this example, the EM attack was done with the HW model performed at the 10th sub- key nibble (4 bits). The correlations between the traces and the HW model show a maximum correlation value (y=0.321) that corresponds to the correct hypothetical sub-key which is (x=6) h as shown in Fig. 10.

After recovering the MSB 64-bit (16 4-bit sub-keys) of the total key, the second step of the attack allows predicting the rest of the key (4 LSB 4-bit sub-keys) by the use of the same HW model. As we mentioned before (see, Section IV-C), this second step of the attack is performed in the second round of the algorithm in order to recover the round key after the first update. We use the outputs of the permutation layer of the first round as the inputs of the HW model. We obtain the correlation between the outputs of the HW model and the 100 000 EM traces extracted at the second round of the PRESENT algorithm. Finally, we reverse the update operations of the round key to compute the missed part of the initial key.

After processing 100 000 EM traces at the first round and 100 000 EM traces at the second round of PRESENT, we succeed to recover the key. Table 2 summarizes the attack results on our PRESENT architecture compared to Axel Poschmann et al. architecture [20]. Our parallel architecture is attacked using 200 000 EM traces, whereas the serial PRESENT architecture proposed by Axel Poschmann et al. is attacked using only 10 000 traces. As we mentioned before that the attack is located at the output of the S-box function. Each attack location allows us to recover one sub-key (4 bits). So, the parallelism of the S-boxes hides the amplitude of the signal of interest. The parallel implementation of PRESENT represents one way of countermeasure against EMA.

TABLE II. ATTACK RESULTS

Block cipher	**Our PRESENT**	**[20]**
Number of traces to perform SCA	200 000	10 000
Attack setup time (s)	100 000	5 000
Area (GEs)	2050	1100

We note that the save of one EM trace on the oscilloscope memory needs about 0.5 s. Thus, the attack setup time equals to number of traces multiplies to 0.5 s.

A parallel implementation of PRESENT block cipher can be a solution to protect the architecture against EMA. However, this solution can be attacked with the use of 200 000 traces. For this reason, in the next session we propose a countermeasure at the authentication protocol level based on time delay function. This function limits the number of successive wrong authentication requests. Then, this limitation of false request prevents the attacker to perform the EM attack previously presented.

V. PROTOCOL BASED COUNTERMEASURE

Different countermeasures can be used to protect RFID chips against SCA. Countermeasures are generally implemented either at the primitive security level (i.e. in the block cipher) or at the communication protocol level. For example, Axel Poschmann et al. proposed a hardware countermeasure on the PRESENT block cipher [20]. In other hand, Chiraag S Juvekar et al. proposed a design of a secure authentication tag [30] that updates the secret key every challenge-response protocol. The tag is based on specific technologies (FRAM and Energy backup unit) which represents high cost security solution.

As we mentioned in Section II-A that ProtocolS is vulnerable to EMA in Step (4). We note that the EMA is always possible in ProtocolS by using a valid reader. However, this attack needs the use of eavesdropping attack to know the RFID communication between reader and tag. To perform the EMA, the attacker needs to eavesdrop the challenge (plaintext) or the response of the tag (cipher text) (see Fig. 1). This attack is considered difficult because the hardness of setting up of the eavesdropping attack. In addition, the EMA is longer because the attacker is oblige to wait for the availability of the reader. In other hand, the ProtocolS allows the attacker to emulate the tag with invalid reader. Therefore, the attacker can send wrong challenges (plaintext) to tag and saves the EM traces easily. In this section, we proposed a countermeasure at ProtocolS to prevent an attacker to emulate the tag with wrong challenges and getting the electromagnetic traces rapidly to perform the EMA. The proposed countermeasure is based on the incrementation of a counter Iwrong every successive wrong authentication request. A delay function allows delaying the response of the tag (especially the encryption operation) with a time delay for each wrong authentication. The time delay function will be described in the following (see Fig. 13). More the number of the wrong authentication request increases (Iwrong), more the time delay increases, more the time to save EM traces increases (see Fig. 12). A state diagram describing this countermeasure is shown in Fig. 11.

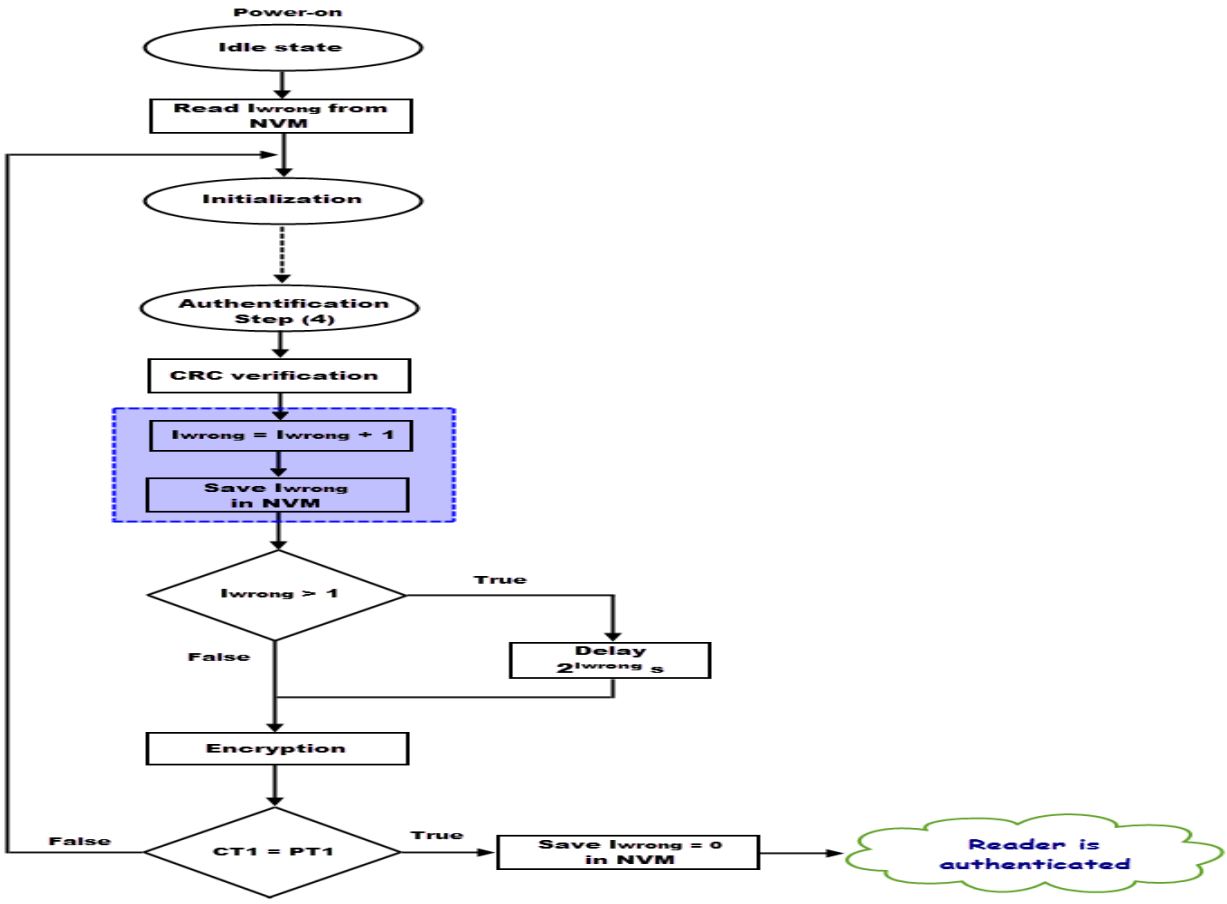

Fig. 11. State diagram describing the countermeasure at ProtocolS.

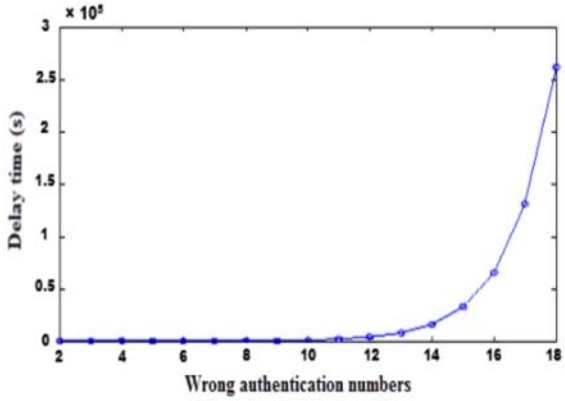

Fig. 12. Delay function description.

At the Step (4) of the ProtocolS, the tag controller verifies the challenge CRC to ensure the integrity of data. If this CRC is correct the Iwrong counter value is first incremented and then saved in a NVM. This backup operation avoids an under-powering attack, which could prevent the incrementation of the counter when the authentication request is false. After this backup operation, in the case of a first authentication, the challenge encryption operation is performed by the PRESENT block cipher without time delay. After each successful authentication (generated CT1= PT1), the tag controller resets the Iwrong value and resets its backup value. The time delay is introduced when the tag detects more than one wrong authentication. It allows delaying the encryption operation when it receives a wrong challenge. The time delay is an exponential function described as:

Time Delay (s) = 2^{Iwrong}

Fig. 12 shows the time delay progression based on the wrong authentication numbers.

Table 3 shows examples of time delay, which is depending to the wrong authentications (Iwrong).

TABLE III. TIME DELAY

Iwrong	2	6	10	14	18
Delay (s)	4	64	1024	16384	262 144

Let's assume that an attacker sends 18 wrong challenges, the total time delay is:

$$Total\ time\ delay\ (s) = \sum_{Iwrong=1}^{18} 2^{Iwrong}$$

The attacker must wait about 524 288 s (~ 145 hours) to obtain 18 EM traces. The results of the table shows that more the number of the wrong authentication increase, more the time to obtain EM traces increases. However, the time delay function allows the tag to enter in killed state progressively. Only the tag manufacturer can reinitialize the tag state to the idle state. The delay function is implemented in our RFID tag prototype described in Fig. 13.

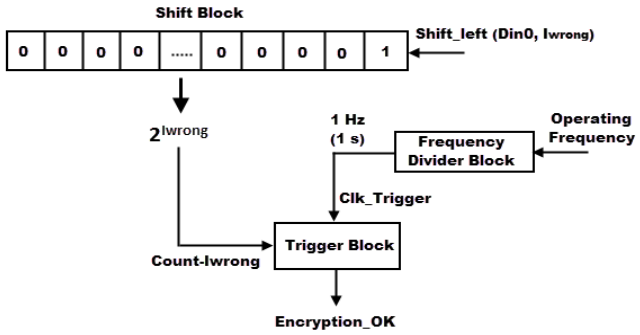

Fig. 13. Delay function description.

The delay function consists of three main blocks: Shift Block, Frequency Divider Block and Trigger Block. The Shift Block uses the shift left operation to calculate 2Iwrong: shift_left (Din0, Iwrong). For example, we designed a Shift Block allows the calculation of a time delay 2Iwrong, with $2 \leq$ Iwrong ≤ 20 (220 second (~ 291 hours) is needed to obtain the 20th EM trace). Din0 is initially defined to (00001) hexadecimal. The Frequency Divider Block generates a frequency of 1 Hz from the operating frequency of the tag. Finally, depending to the Iwrong value, the Trigger Block allows generating a time delay between: 22 s $\leq$ time delay $\leq$ 220 s. The Trigger Block receives the count-Iwrong (from 2 to 20) from the Shift Block. Then it loads its intern counter value by the received count-Iwrong value. The Trigger Block operates at frequency of 1Hz. When the decrementation of the intern counter achieves zero, the Trigger Block generates the signal Encryption_OK that gives the order to perform the encryption operation of the challenge.

The countermeasure at ProtocolS implements the time delay function requires about 960 GEs. It prevents the attacker to save enough EM traces to perform the EMA. It looks economic compared to countermeasures proposed by Axel Poschmann et al. [20] that require 2471 GEs. They proposed countermeasures to PRESENT block cipher based on data masking, key masking and random permutations.

VI. CONCLUSION

This article addresses the issue of EM attacks against mutual authentication protocol in RFID. An improved authentication protocol limiting the number of successive wrong authentication requests is proposed as a countermeasure against EM attacks. This countermeasure prevents an attacker to save enough electromagnetic traces to perform the EMA.

In the first part of this paper, we analyzed the mutual authentication protocol (ProtocolS) and showed how attackers can perform the EM attack on this protocol. Then, we proposed a parallel implementation of PRESENT block cipher in order to hide its information leakage against EMA. Our architecture is attacked after 200 000 traces, whereas the serial architecture of PRESENT proposed by Axel Poschmann et al. [20] is attacked after 10 000 traces. Our PRESENT architecture (2050 GEs) occupies more gates than Axel Poschmann et al. architecture (1100 GEs), but it's more secure against EMA.

In the second part of the paper, we proposed a countermeasure at the protocol level based on time delay function. This countermeasure prevents an attacker to emulate the tag using malicious reader and getting the electromagnetic traces to perform the EMA. Our countermeasure requires only 960 GEs, whereas the countermeasures proposed by Axel Poschmann et al. [20] at PRESENT block cipher requires 2471 GEs. In addition, our countermeasure at protocol level is compatible with unprotected symmetric block ciphers.

REFERENCES

[1] International standard ISO/IEC 14443, "Identification cards, Contactless integrated circuit(s) cards, Proximity cards, Part 2: Radio frequency power and signal interface, First edition", 2001-07-01., Part 3: Initialization and anti-collision, First edition", 2001-02-01., Part 4: Transmission protocol", 2007-06-13.

[2] NXP Semiconductors, MF0ICU2. MIFARE Ultralight C – Contactless ticket IC, Rev. 3.2, 30 June 2014.

[3] Texas Instruments. MIFARE DESFire EV1 AES Authentication with TRF7970A. Application Report SLOA213–December 2014.

[4] U.S. departement of commerce/National Institute of Standards and Technology. Data Encryption Standard (DES). Fips Pub 46-3, Federal Information Processing Standards Publication Reaffirmed 25/10/1999.

[5] National Institute of Standards and Technology (NIST). FIPS-197: Advanced Encryption Standard, November 2001.

[6] Lejla Batina, Amitabh Das, Barıs Ege, Elif Bilge Kavun, Nele Mentens, Christof Paar, Ingrid Verbauwhede, and Tolga Yalcın. Dietary Recommendations for Lightweight Block Ciphers: Power, Energy and Area Analysis of Recently Developed Architectures. Radio Frequency Identification Security and Privacy Issues. 9th International Workshop, RFIDsec 2013 Graz, Austria, July 2013.

[7] Ray Beaulieu, Douglas Shors, Jason Smith, Stefan Treatman-Clark, Bryan Weeks, Louis Wingers. Simon and Speck: Block Ciphers for the Internet of Things. National Security Agency 9800 Savage Road, Fort Meade, MD, 20755, USA. 9 July 2015.

[8] D. Hong, J. Sung, S. Hong, J. Lim, S. Lee, B. Koo, C. Lee, D. Chang, J. Lee, K. Jeong, H. Kim, J. Kim, and S. Chee. HIGHT: A New Block Cipher Suitable for Low-Resource Device. Proc. of Cryptographic Hardware and Embedded Systems, LNCS, vol. 4249, Oct. 2006, pp. 46-59.

[9] R. Needham, D. Wheeler. eXtended Tiny Encryption Algorithm. Technical Report, Cambridge University, England, Oct. 1997.

[10] A. Bogdanov, G. Leander, L. Knudsen, C. Paar, A. Poschmann, M. Robshaw, Y. Seurin, and C. Vikkelsoe. PRESENT - An Ultra-Lightweight Block Cipher. In P. Paillier and I. Verbauwhede, editors. Cryptographic Hardware and Embedded Systems — CHES 2007, number 4727 in Lecture Notes in Computer Science, pages 450–466. Springer-Verlag, 2007.

[11] C. Canniere, O. Dunkelman, and M. Knezevic. Katan and ktantan - a family of small and effcient hardware-oriented block ciphers. Proc. of Cryptographic Hardware and Embedded Systems, LNCS, vol. 459, Sept. 2009, pp. 272-288.

[12] J. Borgho,A. Canteaut T. uneysu, S.S. Thomsen and T. Yalcin. Prince - a low-latency block cipher for pervasive computing applications - extended abstract. Proc. of Advances in Cryptology, LNCS, vol.7658, Dec. 2012, pp. 208-225.

[13] T. Suzaki, K. Minematsu, S. Morioka, and E. Kobayashi. TWINE: Lightweight Block Cipher for Multiple Platforms. Proc. of Selected Areas in Cryptography, LNCS, vol. 7707, Aug. 2013, pp. 339-354.

[14] Chae Hoon Lim and Tymur Korkishko. mCrypton - A Lightweight Block Cipher for Security of Low-Cost RFID Tags and Sensors. Proc. of Information Security Applications, LNCS, vol. 3786, Aug. 2005, pp. 243-258.

[15] Sai Seshabhattar, Shesh Kumar Jagannatha, Daniel W. Engels. Security implementation within Gen2 protocol. 2011 IEEE International Conference on RFID-Technologies and Applications.

[16] Yassine NAIJA, Vincent BEROULLE, David HELY, Mohsen MACHHOUT. Implementation of a Secured Digital Ultralight 14443-Type A RFID tag with an FPGA Platform. 2016 11th International Conference on Design & Technology of Integrated Systems in Nanoscale Era, Istanbul Turkey.

[17] P. C. Kocher. Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems. In N. Koblitz, editor, Advances in Cryptology – CRYPTO '96, 16th Annual International Cryptology Conference, Santa Barbara, California, USA, August 18-22, 1996, Proceedings, number 1109 in Lecture Notes in Computer Science, pages 104–113. Springer, 1996.

[18] P. C. Kocher, J. Jaffe, and B. Jun. Differential Power Analysis. In M. Wiener, editor, Advances in Cryptology - CRYPTO 1999, volume 1666 of Lecture Notes in Computer Science, pages 388–397. Springer Verlag, 1999.

[19] D. Agrawal, B. Archambeault, J. R. Rao, and P. Rohatgi. The EM Side-channel(s). In Burton S. Kaliski, Cetin K. Koc, and Christof. Paar, editors, Cryptographic Hardware and Embedded Systems – CHES 2002, 4th International Workshop, Redwood Shores, CA, USA, August 13-15, 2002, Revised Papers, volume 2523 of Lecture Notes in Computer Science, pages 29–45. Springer, 2003.

[20] Axel Poschmann, Amir Moradi, Khoongming Khoo, Chu-Wee Lim, Huaxiong Wang, San Ling. Side-Channel Resistant Crypto for less than 2,300 GE. International Association for Cryptologic Research, 2011.

[21] M. Feldhofer, S. Dominikus, and J. Wolkerstorfer, "Strong authentication for RFID systems using the AES algorithm," Lecture notes in computer science, pp. 357–370, 2004.

[22] M. Aigner and M. Feldhofer, "Secure symmetric authentication for RFID tags," in Telecommunication and Mobile Computing TCMC2005 Workshop, Graz, Austria. Citeseer, 2005.

[23] E. Brier, C. Clavier, and F.Olivier. Correlation Power Analysis with a Leakage Model. InCHES, volume 3156 of LNCS, pages 16–29. Springer, August 11–13 2004. Cambridge, MA, USA.

[24] MORITA TECH CO, LTD. SAKURA-G, Side-channel Attack User Reference Architecture. Version 1.0. August 1, 2013.

[25] LANGER EMV-Technik. MEASURING SET-UP NEAR FIEMEASURING. Available at www.langer-emv.com.

[26] TELEDYNE LECROY Every were you look. Wave Runner 6 Zi Oscilloscopes 400 MHz –4 GHz. Available at teledynelecroy.com.

[27] International Standard. ISO/IEC 9798-2: Information technology - Security techniques - Entity authentication. Part 2: Mechanisms using symmetric encipherment algorithms, second edition 1999.

[28] Timo Kasper, David Oswald and Christof Paar. EM Side-Channel Attacks on Commercial Contactless Smartcards Using Low-Cost Equipment. In Youm, Heung Youl and Yung, editors. Information Security Applications: 10th International Workshop, WISA 2009, Busan, Korea, August 25-27, 2009.

[29] Timo Kasper, David Oswald, and Christof Paar. Side-Channel Analysis of Cryptographic RFIDs with Analog Demodulation. RFIDSec'11 Proceedings of the 7th international conference on RFID Security and Privacy. Pages 61-77. Amherst, MA - June 26 - 28, 2011.

[30] Chiraag S. Juvekar, Hyung-Min Lee, Joyce Kwong, Anantha P. Chandrakasan. A Keccak-Based Wireless Authentication Tag with per-Query Key Update and Power-Glitch Attack Countermeasures. 2016 IEEE International Solid-State Circuits Conference (ISSCC).