Research Article

Enhanced-Bivium Algorithm for RFID System

Shiyong Zhang, Gongliang Chen, Yongkai Zhou, and Jianhua Li

School of Information Security Engineering, Shanghai Jiaotong University, 800 Dongchuan Road, Shanghai 200240, China

Correspondence should be addressed to Shiyong Zhang; poetzhangzi@sjtu.edu.cn

Received 3 March 2015; Accepted 16 July 2015

Academic Editor: Anna Pandolfi

Copyright © 2015 Shiyong Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

RFID (radio frequency identification) is an emerging technology for automatic identification of physical objects using radio frequency. The security and privacy of the system is increasingly concerned and some light-weight encryption schemes such as Trivium and Bivium have been proposed. In this paper, the internal structure of Bivium is exploited and generalized to an algorithm called Bivium-Model. It is shown that the original Bivium has its inherent imperfections due to the improper parameter selection. A set of better parameters are given out to fix the security vulnerability of original Bivium. Emulation results also show that the Enhanced-Bivium algorithm achieves lower power consumption and lower die size.

1. Introduction

RFID (radio frequency identification) systems are used for the automatic retrieval of data about any object which is equipped with a small circuit called RFID tag. The tag retrieved by a reader device could be used for tracking of goods in industrial systems. In this paper we focus on passive tags that receive their energy from the reader field. For this kind of RFID tags, the field's intensity and the received power consumption are both limited. Therefore, power-aware designing of the tag circuitry is necessary for RFID application [1].

On the other hand, the security and privacy of the system is increasingly concerned because a reader can scan and obtain the data from any tags if they are accessed without any authorization. Some encryption schemes were developed based on classic cipher algorithms such as Advanced Encryption Standard (AES) or Elliptic Curve Cryptography (ECC) [2–4]; Others utilised the hash-based algorithm, pseudo random number generator, and Cyclic Redundancy Check (CRC) function with some XOR and rotation operations [5–8].

However, these schemes are all based on traditional cryptography techniques, which are too costly for the RFID tags. In order to embed the cipher engine into the low-price passive tags, light-weight cipher design is needed.

Trivium is a notable light-weight stream cipher designed by De Cannire and Preneel, submitted to the European

eSTREAM project in April 2005 [9]. It contains 3 rounds in each iteration. In 2007, Raddum proposed a reduced version of Trivium, named as Bivium [10]. He tried to reduce the process of Trivium from 3 to 2 rounds so as to further lower the complexity. However, Bivium has severe security problems. Maximov and Biryukov study two attacks on Trivium [11], which are state recovering and statistical tests. The former is regarded as the best result for attacking Bivium so far and the time complexity to break Bivium is about 2^{37} . Raddum also presents a technique to solve systems of equations associated with Trivium and successfully breaks Bivium in one day [10]. But his attack is very complex when applied to the full cipher and is not faster than exhaustive search. Borghoff et al. presents a numerical attack on Biviums. However, the estimated time complexity of this attack is about $2^{63.7}$ [12].

Based on the above work, in this paper we aim to design a 2-round Bivium-like cipher algorithm without compromising the security and still keeping low resource consumption. Our method is to exploit the internal structure of Bivium and generalize it to an algorithm called “Bivium-Model,” which can be used to analyze the root cause of Bivium's weakness. By adjusting the parameters of “Bivium-Model,” an “Enhanced-Bivium” with better security and power performance will be proposed.

The following part of the paper is organized as follows. The generalized “Bivium-Model” algorithm will be described

in Section 2. Section 3 will analyze the security of the original Bivium and then present an “Enhanced-Bivium.” Section 4 will compare the resource consumption of Trivium, Bivium, and Enhanced-Bivium through emulation results. The conclusion and future work will be given in Section 5.

Notation 1. Bold symbols in capital letter and small letter denote matrices and vectors, respectively. $|\cdot|$ is the determinant of a square matrix. $\mathbf{I}$ stands for the identity matrix. $\text{GF}(2)$ denotes the Galois field of two elements. $a \mid b$ means b is divisible by a .

2. Bivium-Model Algorithm

Trivium [9] is designed to generate up to 2^{64} bits of key stream from an 80-bit secret key (Key) and an 80-bit initial value (IV). The process consists of two phases: first the internal state of the cipher is initialized using Key and IV; then the state is repeatedly updated and used to generate key stream

$$\mathbf{s}(t) = \begin{cases} (s_1, s_2, \dots, s_{3n_1}) \leftarrow (K_1, K_2, \dots, K_{80}, 0, \dots, 0) \\ (s_{3n_1+1}, s_{3n_1+2}, \dots, s_{3n_2}) \leftarrow (IV_1, IV_2, \dots, IV_{80}, 0, \dots, 0, 1, 1, 1) \end{cases} \quad (1)$$

By this way, any of the Bivium-like algorithm can then be represented by $\{3u_1, 3u_2, 3n_1\}$ and $\{3u_3, 3u_4, 3n_2\}$ tuples.

Proposition 1. *Bivium algorithm is a Bivium-Model algorithm, with parameters shown in Table 1.*

3. Security Analysis and “Enhanced-Bivium”

The original Bivium algorithm has some security problems. In this section, we will analyze its security based on the generalized “Bivium-Model” algorithm and adjust the parameters to induce a better performance.

We begin by studying the property of Bivium-Model algorithm. Denote the internal state bits at time t as $\mathbf{s}(t) = (s_1(t), s_2(t), \dots, s_{3n_2}(t))$; then the internal bits from time t to time $t+1$ can be expressed as

$$\mathbf{s}(t+1) = \mathbf{A} \cdot \mathbf{s}(t) + \mathbf{b}, \quad (2)$$

where $\mathbf{A} = (a_{ij})_{3n_2 \times 3n_2}$ is the state-transition matrix of the algorithm with size $3n_2 \times 3n_2$:

$$a_{ij} = \begin{cases} 1, & i = 1, j = 3u_2, 3u_3, 3n_2, \\ 1, & i = 3n_1 + 1, j = 3u_1, 3u_4, \\ 1, & 1 < i \leq 3n_2, j = i - 1, \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

bits. There are 288 bits in the internal state. Figure 1 shows the structure of the algorithm. Each iteration consists of 3 rounds with similar structure.

Bivium proposed by Raddum [10] is a reduced version of Trivium. It breaks the cipher into smaller parts and reconstructs these parts as 2 rounds. The internal state $\mathbf{s} = (s_1, s_2, \dots, s_{177})$ of Bivium drops to 177 bits. Denote the intermediate variable as t_1, t_2 and the output stream as $\mathbf{z} = (z_1, z_2, \dots, z_N)$, with N standing for the number of output bits. The complete process of the generation key-stream phase is given by the pseudo-code shown in Algorithm 1.

By extracting the sequence index of Algorithm 1, the process of Bivium can be generalized to a “Bivium-Model” algorithm, shown as Algorithm 2. u_i ($i = 1, \dots, 4$), n_i ($i = 1, 2$), and $u_1 < u_2 < n_1 < u_3 < u_4 < n_2$ are the parameters of Algorithm 2. We will show in later section that these indices actually correspond to the degree of the characteristic polynomials.

The structure of the Bivium-Model algorithm is given by Figure 2. Key and IV are loaded as follows:

$\mathbf{b} = (b_i)_{3n_2}$ is the nonlinear part of the algorithm which is treated as vectors of bits:

$$b_i = \begin{cases} s_{3n_2-2} \cdot s_{3n_2-1}, & i = 1, \\ s_{3n_1-2} \cdot s_{3n_1-1}, & i = 3n_1 + 1, \\ 0, & \text{otherwise.} \end{cases} \quad (4)$$

Hence we can get the theorem as follows:

Theorem 2. *The characteristic polynomial of Bivium-Model algorithm is in the following form:*

$$f(x) = (x^3 + 1)^2 \cdot g(x^3), \quad (5)$$

where $g(\cdot)$ is a polynomial.

Proof. $\mathbf{A}$ is the state transformation matrix of Bivium-Model algorithm defined in (3). The characteristic polynomial $f(x)$ is

$$\begin{aligned} f(x) &= |x\mathbf{I} - \mathbf{A}| \\ &= x^{3n_2} + x^{3n_2-3u_2} + x^{3n_2+3n_1-3u_4} + x^{3n_2+3n_1-3u_1-3u_3} \\ &\quad + x^{3n_2+3n_1-3u_2-3u_4} + x^{3n_1-3u_1} + x^{3n_2-3u_3} + 1. \end{aligned} \quad (6)$$

Let $h(x) = x^{n_2} + x^{n_2-u_2} + x^{n_2+n_1-u_4} + x^{n_2+n_1-u_1-u_3} + x^{n_2+n_1-u_2-u_4} + x^{n_1-u_1} + x^{n_2-u_3} + 1$; we have $f(x) = h(x^3)$.

Denote $y = x+1$; we only need to prove that $y^2 \mid h(y+1)$. That is to say, we need to prove that the constant coefficient and the monomial coefficient of $h(y+1)$ are both 0.

FIGURE 1: Structure of Trivium.

FIGURE 2: Structure of Bivium-Model.

TABLE 1: Parameters of Bivium.

u_1	u_2	n_1	u_3	u_4	n_2
22	23	31	54	57	59

The constant coefficient C_0 of $h(y+1)$ is

$$C_0 = h(y+1)|_{y=0} = h(1) = 0. \quad (7)$$

```

for  $i = 1$  to  $N$  do
   $t_1 \leftarrow s_{66} + s_{93}$ 
   $t_2 \leftarrow s_{162} + s_{177}$ 
   $z_i \leftarrow t_1 + t_2$ 
   $t_1 \leftarrow t_1 + s_{91} \cdot s_{92} + s_{171}$ 
   $t_2 \leftarrow t_2 + s_{175} \cdot s_{176} + s_{69}$ 
   $(s_1, s_2, \dots, s_{93}) \leftarrow (t_2, s_1, \dots, s_{92})$ 
   $(s_{94}, s_{95}, \dots, s_{177}) \leftarrow (t_1, s_{94}, \dots, s_{176})$ 
end for

```

ALGORITHM 1: Bivium algorithm.

```

for  $i = 1$  to  $N$  do
   $t_1 \leftarrow s_{3u_1} + s_{3n_1}$ 
   $t_2 \leftarrow s_{3u_3} + s_{3n_2}$ 
   $z_i \leftarrow t_1 + t_2$ 
   $t_1 \leftarrow t_1 + s_{3n_1-2} \cdot s_{3n_1-1} + s_{3u_4}$ 
   $t_2 \leftarrow t_2 + s_{3n_2-2} \cdot s_{3n_2-1} + s_{3u_2}$ 
   $(s_1, s_2, \dots, s_{3n_1}) \leftarrow (t_2, s_1, \dots, s_{3n_1-1})$ 
   $(s_{3n_1+1}, s_{3n_1+2}, \dots, s_{3n_2}) \leftarrow (t_1, s_{3n_1+1}, \dots, s_{3n_2-1})$ 
end for

```

ALGORITHM 2: Generalized “Bivium-Model” algorithm.

The monomial coefficient C_1 of $h(y+1)$ is

$$\begin{aligned}
 C_1 &= \sum_i \binom{i}{1} = \sum_i i \\
 &= n_2 + n_2 - u_2 + n_2 + n_1 - u_4 + n_2 + n_1 - u_1 - u_3 \\
 &\quad + n_2 + n_1 - u_2 - u_4 + n_1 - u_1 + n_2 - u_3 = 0.
 \end{aligned} \quad (8)$$

Therefore,

$$\begin{aligned}
 &y^2 \mid h(y+1) \\
 &\Rightarrow (1+x)^3 \mid h(x) \\
 &\Rightarrow h(x) = (1+x)^2 \cdot g(x) \\
 &\Rightarrow f(x) = h(x^3) = (x^3+1)^2 \cdot g(x^3).
 \end{aligned} \quad (9)$$

□

Here, we define the term “ k -order primitive polynomial” to evaluate the property of the characteristic polynomial, $f(x)$.

Definition 3. Given $f(x) = \sum_{i=0}^n a_i x^i$, $n > k$, $a_i \in \text{GF}(2)$, $i = 0, 1, \dots, n$, $f(x)$ is called a k -order primitive polynomial if $f(x) = (x+1)^k \cdot g(x)$, where $g(x)$ is a primitive polynomial.

Proposition 4. Primitive polynomial is 0-order primitive polynomial.

Proposition 5. The characteristic polynomial of the Bivium is not a 2-order primitive polynomial.

TABLE 2: Parameters of Enhanced-Bivium.

u_1	u_2	n_1	u_3	u_4	n_2
21	25	30	31	46	58

Proof. By (6), the transformation matrix of the Bivium can be expressed as

$$\begin{aligned} f(x) &= x^{177} + x^{108} + x^{99} + x^{42} + x^{30} + x^{27} + x^{15} + 1 \\ &= (x^3 + 1)^2 \cdot g(x^3). \end{aligned} \quad (10)$$

Here g is not a primitive polynomial and can be decomposed as

$$\begin{aligned} g(y) &= y^{57} + y^{55} + y^{53} + y^{51} + y^{49} + y^{47} + y^{45} + y^{43} \\ &\quad + y^{41} + y^{39} + y^{37} + y^{35} + y^{34} + y^{33} + y^{32} + y^{30} \\ &\quad + y^{28} + y^{26} + y^{24} + y^{22} + y^{20} + y^{18} + y^{16} + y^{14} \\ &\quad + y^8 + y^7 + y^6 + y^5 + y^4 + y^2 + 1 = (y^5 + y^4 + y^3 \\ &\quad + y^2 + 1) \cdot (y^7 + y^4 + 1) \cdot (y^{45} + y^{44} + y^{43} + y^{41} \\ &\quad + y^{34} + y^{32} + y^{26} + y^{21} + y^{18} + y^{17} + y^{13} + y^9 \\ &\quad + y^7 + y^5 + y^4 + y^3 + 1). \end{aligned} \quad (11)$$

Therefore $f(x)$ is not a 2-order primitive polynomial. $\square$

From Proposition 5, we can see that the characteristic polynomial of the Bivium algorithm is not a 2-order primitive polynomial. Therefore, it cannot guarantee large periods, and this is the root cause for the state recovering attack on Bivium [11].

In order to solve the imperfection of the original Bivium, we try to choose sets of better parameters. Direct calculation of all the 2-order primitive polynomials is too complex. Therefore we determine the parameters under the following conditions:

- (1) The characteristic polynomial of the new parameters is a 2-order primitive polynomial.
- (2) To reduce the resource consumption while keeping the security, we specially fix the value $n_2 = 58$ and the range of $u_1 \in [20, 24]$.
- (3) To improve the security, we suppose $u_3 = n_1 + 1$.

However, except for condition 1, the other 2 conditions do not necessarily result in the best parameter set. We just guarantee that the parameters that meet these conditions are more likely to be better. We use Maple to search the parameters [13]. The code is shown in Algorithm 3.

Algorithm 3 outputs more than 200 results, and we choose one of these parameter sets, shown in Table 2: and the resulting algorithm is referred to as “Enhanced-Bivium.”

Proposition 6. *The characteristic polynomial of “Enhanced-Bivium” is a 2-order primitive polynomial.*

Proof. By (6), the transformation matrix of the “Enhanced-Bivium” can be expressed as

$$\begin{aligned} f(x) &= x^{174} + x^{126} + x^{108} + x^{99} + x^{81} + x^{51} + x^{27} + 1 \\ &= (x^3 + 1)^2 \cdot g(x^3), \end{aligned} \quad (12)$$

where

$$\begin{aligned} g(y) &= y^{56} + y^{54} + y^{52} + y^{50} + y^{48} + y^{46} + y^{44} + y^{42} \\ &\quad + y^{34} + y^{32} + y^{31} + y^{30} + y^{29} + y^{28} + y^{27} \\ &\quad + y^{26} + y^{24} + y^{22} + y^{20} + y^{18} + y^{16} + y^{15} \\ &\quad + y^{14} + y^{13} + y^{12} + y^{11} + y^{10} + y^9 + y^8 + y^6 \\ &\quad + y^4 + y^2 + 1 \end{aligned} \quad (13)$$

and it can be verified that $g(\cdot)$ is a primitive polynomial.

Therefore $f(x)$ is a 2-order primitive polynomial. $\square$

According to Proposition 6, the new set of parameters ensure $f(x)$ to be a 2-order primitive polynomial, thus making the resulting algorithm more resistant to the state recovering attack.

Note that the characteristic polynomial of the 3-round Trivium [9] can be derived in a similar way, expressed by

$$\begin{aligned} f(x) &= x^{288} + x^{219} + x^{210} + x^{201} + x^{141} + x^{132} + x^{123} \\ &\quad + x^{87} + x^{72} + x^{60} + x^{54} + x^{45} + x^{42} + x^{27} \\ &\quad + x^{15} + 1 = (x^3 + 1)^3 \cdot g(x^3), \end{aligned} \quad (14)$$

where $g(\cdot)$ is a primitive polynomial which can be expressed as

$$\begin{aligned} g(y) &= y^{93} + y^{92} + y^{89} + y^{88} + y^{85} + y^{84} + y^{81} + y^{80} \\ &\quad + y^{77} + y^{76} + y^{73} + y^{72} + y^{70} + y^{68} + y^{67} \\ &\quad + y^{44} + y^{43} + y^{41} + y^{39} + y^{38} + y^{35} + y^{34} \\ &\quad + y^{31} + y^{30} + y^{27} + y^{25} + y^{23} + y^{20} + y^{19} \\ &\quad + y^{17} + y^{14} + y^{13} + y^{12} + y^9 + y^8 + y^6 + y^4 \\ &\quad + y + 1. \end{aligned} \quad (15)$$

Therefore, we will have the following proposition.

Proposition 7. *The characteristic polynomial of the Trivium algorithm is a 3-order primitive polynomial.*

Finally, we compare the Trivium, Bivium, and Enhanced-Bivium under the state recovering attack. The result is shown in Table 3.

It is observed that the Trivium algorithm is most robust to state recovering attack. Bivium algorithm can be broken in time around 2^{37} , which is a relatively low level. “Enhanced-Bivium” performs much better than original Bivium due to its 2-order primitive structure.

```

n2 = 58
for u1 = 20 to 24 do
  for u2 = u1 + 1 to 54 do
    for n1 = u2 + 1 to 55 do
      u3 = n1 + 1
      for u4 = u3 + 1 to 57 do
        g(x) = xn2 + xn2-u2 + xn2+n1-u4 + xn2+n1-u1-u3 + xn2+n1-u2-u4 + xn1-u1 + xn2-u3 + 1
        if Divide(g(x), (x + 1)2, "h") mod 2 then
          if Irreduc(h) mod 2 and CheckPrimitive(h) mod 2 then
            printf("%d, %d, %d, %d, %d, %d \ n", u1, u2, n1, u3, u4, n2)
          end if
        end if
      end for
    end for
  end for
end for
end for
end for

```

ALGORITHM 3: Maple code for parameter selection.

TABLE 3: Comparison of three algorithms under state recovering attack.

Algorithm	Trivium	Bivium	Enhanced-Bivium
Breaking complexity	$2^{83.5}$	2^{37}	2^{57}

4. Emulation Results for Resource Consumption

For hardware emulation, we use protocols for symmetric challenge-response techniques based on encryption which are defined in the ISO/IEC 9798-2 standard [14]. Unilateral authentication works as follows: there are two partners *A* and *B*. Both possess the same private key *K*. *B* sends a random number r_B to *A*. *A* then encrypts the random number with the shared key *K* and sends it back to *B*. *B* proofs the result and can verify the identity of *A*.

In order to achieve a significant economic benefit from using RFID systems, tags will need to be priced under US\$ 0.10 [15]. The available power consumption for the digital part of the RFID tag is amounting to 20 μ A. Estimating the current consumption of the digital controller to be 5 μ A, 15 μ A remains for the Bivium-Model module which should not exceed a chip area of 5,000 gates [3]. In a word, the less gates, the better.

We use Verilog to simulate the "Enhanced-Bivium" algorithm as well as other algorithms and use Lattice Diamond which can offer tools optimized for FPGA architectures to simulate the hardware property and action on LFXP3C-5T100C. The implementation of the data-path of Enhanced-Bivium algorithm has a current consumption of 0.41 μ A. The required hardware complexity is estimated to be 2109 gates. The resource consumption of Trivium, Bivium, Enhanced-Bivium, and AES algorithm are shown in Table 4 and the comparison is based on energy consumption and gate equivalent (GE) count.

TABLE 4: Resource consumption of 4 cipher algorithm.

Algorithm	μ A@100 kHz	GE
Trivium	0.68	3488
Bivium	0.42	2145
Enhanced-Bivium	0.41	2109
AES	3	3400

From the result, it can be seen that, compared to the 3-round Trivium, Bivium and Enhanced-Bivium have better performance due to less internal rounds. Furthermore, Enhanced-Bivium consumes even lower resources than the original Bivium; this is because the internal state bits of Enhanced-Bivium drops from 177 to 174 bits and the degree of the characteristic polynomial of the Enhanced-Bivium is lower than the degree of original Bivium.

5. Conclusion and Future Work

In this paper, an "Enhanced-Bivium" encryption scheme is proposed to meet the need of the RFID system in terms of high security and low resource consumption. The internal structure of Bivium is studied and generalized to "Bivium-Model" algorithm. A set of better parameters are given out to fix the security vulnerability of the original Bivium under state-recover attack. Emulation results show that it also achieves lower power consumption and die size.

We will continue to study the property of "Bivium-Model" algorithm. Future work will focus on the search of new parameters with better consideration of the balance of security and effectiveness.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was supported in part by International Researcher Exchange Project of National Science Foundation of China and Centre national de la recherche scientifique de France (NSFC-CNRS) under Grant no. 61211130104 and National Science Foundation of China under Grant no. 61271220.

References

- [1] K. Finkenzeller, *RFID-Handbook*, Springer, Berlin, Germany; Carl Hanser, Munchen, Germany, 2nd edition, 2003.
- [2] J.-S. Chou, Y. Chen, C.-L. Wu, and C.-F. Lin, "An efficient RFID mutual authentication scheme based on ECC," *Cryptology ePrint Archive Report 2011/418*, IACR, 2011.
- [3] M. Feldhofer, S. Dominikus, and J. Wolkerstorfer, "Strong authentication for RFID systems using the AES algorithm," in *Proceedings of the Workshop on Cryptographic Hardware and Embedded Systems (CHES '04)*, Cambridge, Mass, USA, August 2004.
- [4] T. A. Pham, M. S. Hasan, and H. Yu, "A RFID mutual authentication protocol based on AES algorithm," in *Proceedings of the UKACC International Conference on Control (CONTROL '12)*, pp. 997–1002, Cardiff, UK, September 2012.
- [5] E.-J. Yoon, "Improvement of the securing RFID systems conforming to EPC Class 1 Generation 2 standard," *Expert Systems with Applications*, vol. 39, no. 1, pp. 1589–1594, 2012.
- [6] H. Li, Y. Ping, W. Xuan, and L. Pang, "A novel hash-based RFID mutual authentication protocol," in *Proceedings of the 7th International Conference on Computational Intelligence and Security (CIS '11)*, pp. 774–778, December 2011.
- [7] T.-C. Yeh, Y.-J. Wang, T.-C. Kuo, and S.-S. Wang, "Securing RFID systems conforming to EPC class 1 generation 2 standard," *Expert Systems with Applications*, vol. 37, no. 12, pp. 7678–7683, 2010.
- [8] Y. Liu, "An efficient RFID authentication protocol for low-cost tags," in *Proceedings of the IEEE/IFIP International Conference on Embedded and Ubiquitous Computing (EUC '08)*, pp. 180–185, December 2008.
- [9] C. De Cannire and B. Preneel, "TRIVIUM specifications," eSTREAM, ECRYPT stream Cipher Project 2005/030, April 2005, <http://www.ecrypt.eu.org/stream>.
- [10] H. Raddum, *Cryptanalytic Results on Trivium*, 2007, <http://www.ecrypt.eu.org/stream/papersdir/2006/039.ps>.
- [11] A. Maximov and A. Biryukov, "Two trivial attacks on TRIVIUM," in *SASC 2007: The State of the Art of Stream Ciphers*, pp. 1–16, 2007.
- [12] J. Borghoff, L. R. Knudsen, and M. Stolpe, "Bivium as a mixed-integer linear programming problem," in *Cryptography and Coding: 12th IMA International Conference, Cryptography and Coding 2009, Cirencester, UK, December 15–17, 2009. Proceedings*, M. G. Parker, Ed., vol. 5921 of *Lecture Notes in Computer Science*, pp. 133–152, Springer, Heidelberg, Germany, 2009.
- [13] Maplesoft, division of Waterloo Maple, Incorporation, <http://www.maplesoft.com/>.
- [14] International Organization for Standardization, "Information technology—security techniques entity authentication mechanisms part 2: entity authentication using symmetric techniques," ISO/IEC 9798-2, ISO/IEC, 1993.
- [15] S. E. Sarma, S. A. Weis, and D. W. Engels, "RFID systems and security and privacy implications," in *Cryptographic Hardware*

and Embedded Systems—CHES 2002: 4th International Workshop Redwood Shores, CA, USA, August 13–15, 2002 Revised Papers, vol. 2523 of *Lecture Notes in Computer Science*, pp. 454–470, Springer, Berlin, Germany, 2002.