CS553 - Term Paper

Group - Codebreakers

November 27, 2020

1 Abstract

In this paper, we talk about radiofrequency identification (RFID) cards, integrated circuit printing, or IC-printing and then consider the need and cryptographic implications of the Print Cipher. We have two versions of the Print Cipher depending on their block size: Print Cipher-48 and Print Cipher-96. And then we move towards security goals and security analysis where, concerning the key cryptanalytic methods known, we evaluate the protection of our proposal.

2 Introduction

The radiofrequency identification (RFID) system has an RFID reader and an RFID tag. RFID tags are a form of tracking system that identifies objects using smart barcodes. There are three types of RFID tags, namely, Active tag, Passive tag, and Semipassive tag based on their power supply. An active tag has its power supply, whereas a passive lag has no power supply, and a semipassive tag has its power supply but to transmit feedback signal to the RFID reader, it depends on the power supplied by the RFID reader. An active tag has a maximum range, whereas a passive and semi-passive tag has quite much the same range but less than that of an active tag. RFID cards or tags use radiofrequency technology to transmit data from a tag to a reader which then transmits the information to an RFID computer program.

The cryptographic algorithm provides confidentiality and maintains the integrity of the information, but due to exceptional physical and economic constraints, we must leave behind conventional cryptography because conventional cryptographic techniques require higher resources, but the problem with this RFID tag is that they have minimal resources like limited computation, limited memory, power, and small size. For example, the RSA algorithm of 1204 bits, which will consume most of the resources of RFID, which already have minimal resources. Therefore, for other functions, RFID won't have adequate resources. So it is not possible to implement the RSA algorithm of 1204 bits on RFID. For the proper functioning of the RFID, any cipher with more significant gate equivalence (GE) will create a problem. Only PRINT cipher and EPCBC cipher can be used for the RFID having a corresponding chip area of 402/726 GE and 1008 GE.

This paper takes into account technological advancements, such as integrated circuit printing or IC printing. The technology remains in its infancy and it has yet to fully understand its true potential. But the claimed advantages include the ability to print on thin and flexible materials and being much cheaper than silicon-based deployments because the conventional manufacturing process is bypassed. Since the main driver for IC printing is cost-effective, the typically mentioned application areas overlap closely with the typical lightweight cryptography domains. Indeed in the development of cheap RFID tags, one of the applications is IC-printing. Therefore, some of the strategies suggested for traditional RFID tags and those that will be used on printed tags have a lot in common.

For many factors, block ciphers create a natural starting point. They may not only be used in several different ways, but we feel a little more relaxed with them as a group. Analysis and design. That said we are working right at the edge of proven practice for such extreme environments as IC-printing and we are forced to consider and highlight some interesting problems. This is the objective behind the PRINT cipher of the block cipher.

3 The design approach to PRINT cipher

We can conceptually imagine that an "encryption computation" and a "subkey computation" are required inside a block cipher. The short-cuts we can build for the first are minimal, as we are restricted by the attentions of the cryptanalyst. For the most part, this means that proposals for a given security level and a given set of parameters for the block cipher will occupy almost the same area. If we were to reduce the space occupied by an execution, then the block size would most likely be reduced. Nevertheless, for the "subkey calculation," things are a little different and it is not always clear precisely how a key should be used. This emphasizes two different problems.

The first problem is whether in an application a key is likely to be modified. For RFID applications, it is very unlikely that one will want to change the key. Indeed, some other work on RFID implementation has shown that the overhead can be important in promoting a key shift.

The second problem is the exact shape of the key schedule. Some block ciphers, e.g. IDEA has a very simple key schedule in which subkeys are produced by user-supplied key sampling bits. The benefit of this method is that the subkey computations require no working memory. Some key schedule computation is available from other lightweight block ciphers, e.g. PRESENT, while another CGEN proposal proposes to use no key schedule, without any sampling or extra computation, the user-supplied key is used.

4 Implementation

PRINTcipher is a block ciher with b-bit blocks, b ϵ {48, 96}. PRINTcipher has an effective key length of 5/3 * b bits and a total of b rounds. For Ex- PRINTcipher-48 has 48-bit blocks, uses and (5/3*40) = 80-bit key and consists of 48 rounds. Similarly, PRINTcipher-96 operates on 96-bit blocks, uses a (5/3 * 96) = 160-bit key and consists of 96 rounds.

Each round of PRINTcipher consists of the following steps:

1) Bitwise exclusive-or (xor) of cipher state is done with a round key
2) The cipher state is mixed up using a fixed linear diffusion layer.
3) Bitwise exclusive-or (xor) of an n-least significant bit of cipher state is done with a round counter, where n = $log_2 r$, r is no of rounds.
4) Three-bit input to S-box is first permutated in a key-dependent manner.
5) The cipher state is passed through b/3 non-linear S-box layer.

1) **Key Xor:** Bitwise xor of the cipher state is done with a b-bit subkey(sk1) from a total of 5/3 * b bits effective key. This subkey is identical in all rounds.

2) **Linear Diffusion:** The permutation-layer is a simple bit permutation. In general, an SP-network with block size b and s bit S-boxes, bit permutation is given by:

$$p(i) = s \times i \; mod(b-1) \; ; for \; 0 \le i \le (b-2)$$
$$p(i) = b-1; for \; i = b-1$$

In case of PRINTcipher, we have 3 bit s-boxes, so s = 3. Hence in PRINT cipher bit *i* of the current state is moved to bit position *P(i)* in the following way:

$p(i) = 3 \times i\, mod(b-1)\; ; for\; 0 \le i \le (b-2)$
$p(i) = b-1; for\; i = b-1$

3) Round Counter RC_i**:** The round counter RC_i for $1 \le i \le$ r is combined using xor to the least significant bits of the current state. The values of the round counter are generated using an n-bit shift-register (n = $[log_2 r]$) in the following way. In case of PRINTcipher-48, r = 48. So, n = $[log_2 48]$ = 6. Let's denote the state of the shift register as Xn-1 || ....||X1 ||X0 (in case of PRINTcipher-48, state of shift register is denoted as X5||X4||X3||X2||X1||X0). The shift register is initialised to all zeros. In case of PRINTcipher-48, shift register is initialised as 000000. Round counter RCi is computed as:

t = 1 + Xn-1 + Xn-2
Xi = Xi-1 for n-1 ≥ i ≥ 1
X0 = t

Example:
Sequence of RCi for PRINTcipher-48 in hexadecimal notation:
RC1 :
X = 0 || 0 || 0 || 0 || 0 || 0
t = 1 + 0 + 0 = 1
X = 0 || 0 || 0 || 0 || 0 || 1
RC1 = 01

RC2 :
t = 1 + 0 + 0 = 1
X = 0 || 0 || 0 || 0 || 1 || 1
RC2 = 03

RC3 :
t = 1 + 0 + 0 = 1
X = 0 || 0 || 0 || 1 || 1 || 1
RC3 = 07

RC4 :
t = 1 + 0 + 0 = 1
X = 0 || 0 || 1 || 1 || 1 || 1

RC4 = 0F

RC5 :
t = 1 + 0 + 0 = 1
X = 0 || 1 || 1 || 1|| 1 || 1
RC5 = 1F

RC6 :
t = 1 + 0 + 1 = 0
X = 1 || 1 || 1 || 1 || 1 || 0
RC6 = 3E

RC7 :
t = 1
X = 1 || 1 || 1 || 1 || 0 ||
RC7 = 3E

4) Keyed Permutation: We have 3-bit S-boxes and hence 3-bit input to it. The set of 3-bit input is first permuted among themselves and then fed to s-box. And these permutations are done in a key-dependent manner.

The 5/3 b-bit user-supplied key k is considered as consisting of two subkey components k = sk1||sk2 where sk1 is b bits long and sk2 is 2/3 b-bits long. The first subkey is used, unchanged, within the xor layer of each and every round. The second subkey sk2 is used to generate the key-dependent permutations in the following way. The 2/3 b bits are divided into b/3 sets of two bits and each two-bit quantity a1||a0 is used to pick one of four of the available permutations. Specifically, the three input bits c2||c1||c0 are permuted to give the following output bits according to the value of a1||a0.

a1||a0	
00	C2 || c1 || c0
01	C1 || c2 || c0
10	C2 || c0 || c1
11	C0 || c1 || c2

5) S-box layer: A total of 3-bit to 3-bit S-box is used b/3 times in parallel. Each set of three input bit is applied to the S-box layer to get 3-bit output. The S-box is used in the following way:

x	0	1	2	3	4	5	6	7
S[x]	0	1	3	6	7	4	5	2

Keyed permutation with S-box layer:

Applying bit-permutation:

a1||a 0		**x**	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
	Binary		**000**	**001**	**010**	**011**	**100**	**101**	**110**	**111**
00	**c2||C1||C0**	**L0[X]**	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
01	**C1||C2||C0**	**L1[X]**	**0**	**1**	**4**	**5**	**2**	**3**	**6**	**7**
10	**C2||C0||C1**	**L2[X]**	**0**	**2**	**1**	**3**	**4**	**6**	**5**	**7**
11	**C0||C1||C2**	**L3[X]**	**0**	**4**	**2**	**6**	**1**	**5**	**3**	**7**

Applying the S-box layer:

L[x]	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
V0[x]	**0**	**1**	**3**	**6**	**7**	**4**	**5**	**2**
V1[x]	**0**	**1**	**7**	**4**	**3**	**6**	**5**	**2**
V2[x]	**0**	**3**	**1**	**6**	**7**	**5**	**4**	**2**
V3[x]	**0**	**7**	**3**	**5**	**1**	**4**	**6**	**2**

5 SECURITY ANALYSIS:

5.1 Differential and Linear Characteristics

$$q = (2p - 1)^2$$

p- linear characteristic probablity
q- correlation of the linear characteristic

S-box in PRINT cipher have remarkable differential properties as well as linear properties. These properties are taken by the other 3 virtual S-boxes, and if we combine the dependent key permutation and S-box operation on any differential characteristic over any S-box, it has a probability of at most 1/4, and any linear characteristic over a S-box has a correlation of at most 1/4. If a characterstic is having more than s rounds(s-bit Sbox) , then we would have atleast one active S box per round. An differential characteristic of s-round will have a probability of at most 2^{-2s} and any *s*-round linear characteristic will have a correlation of at most 2^{-2s-}.

Thus, conventional differential and linear characteristics are unlikely to play a role in the cryptanalysis of PRINTcipher with the specified 48 respectively 96 rounds.

Lets see the the 1-round iterative characteristic below:-

$$(0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1) \rightarrow (0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1)$$

We can see that there is only one active s box which is in the least significant bit. This has probability 1/4 when the active S-box is V_0 or V_1. The iterative characteristic above has an expected probability of 2 for 12 rounds.

We repeated this with twenty randomly chosen keys but made sure that either *V*0 or *V*1 is in the least significant bit. We generated 2^{28} pairs foe that difference. There were 16.6 number of pairs of the expected difference after we have done encryption of 12 rounds while 16 was expected for characteristic. For about 14 rounds and taking 2^{30} pairs, the pairs are found to be 4.5 on average, where it was expected to be 4. This iterative characteristic has expected probability of is 2^{-28}.

We can see here that there is not significant amount of differential effect here. If we want to calculate the exact differential effect, then it is a very difficult and complicated task for PRINT cipher. As the probability of the iterative characteristic is very low, *i.e.* 2^{-80} for fourty round, which leads to the expectation that good probability differentials are very less likely to exist for PRINT cipher.

5.2 High Order Differentials and Algebraic Attacks

The Algebraic degree of the S-box is 2 and as there are large number of rounds (48 or 96 rounds) in PRINT cipher so the overall degree of the cipher should be near maximum.

To prove this we did the following experiment:

We now that if a function has algebraic degree *m*, a m^{th} order differential will be constant, and the value of a $(m+1)^{st}$ order differential will be zero. if a m^{th} order differential over *s* rounds for one key is not zero, then the algebraic degree of this encryption function is at least $m-1$. For seven rounds of PRINT cipher and by using ten randomly chosen keys we computed the values of two different 25th order differentials and found the values to be non zero.

So this expermiment proves that the cipher reaches its maximum degree much before 48 rounds.

It is secure against higher order differential attacks.

We can see that there exist quadratic equations for all 3-bit S-boxes.

So, secret key of one particular encryption will be the solution to a number of quadratic equations. Such system of equations will be huge because of the large number of rounds and it can’t be solved faster than the time it will take to trying all the keys. The key dependent permutations make the equations even more complex and harder to solve.

5.3 Related-Key Attacks

The four S-boxes in PRINTcipher are very closely related for example S-box 0 and S-box 1 produce the same output for each of four inputs and similarly for S-boxes 2 and 3 and for S-boxes 4 and 5.

Now let’s consider the following thing:

Take two keys that varies only in the selection of one S-box, say, the leftmost one. Assume that one key selects S-box *V*0 and the other key selects S-box *V*1. For one round of encryption, the encryption function induced by the two keys will be equal for half the inputs. Similarly,the encryption functions over *s* rounds can be expected to produce identical ciphertexts for one in 2^s texts.

There are other related keys.

For example consider two keys different only in xor halves and only in the input to one S-box. For such two keys it may be possible that the differences in the texts are canceled by the differences in the xor key in every second round. If in all other rounds it is assumed that there is only one active S-box and that the difference in the inputs equal the difference in the outputs, then one gets an *s*-round differential characteristic of probability 2^{-s} (for even *s*).

The above observations can be used to devise related-key attacks which could recover a key for PRINT cipher using a little less than 2^{b} texts. It is clear, however, that if the keys of PRINT cipher are chosen uniformly at random it is very unlikely that one would find keys related as described above.

5.4 Statistical Saturation Attacks

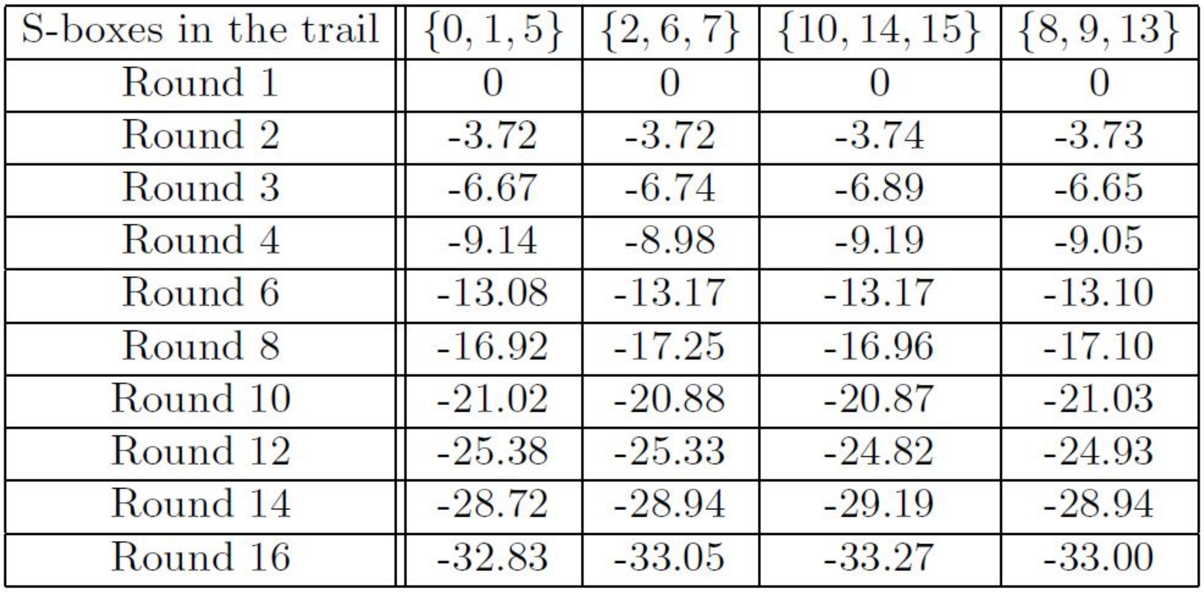
The Statistical Saturation attack applies to reduced round versions of PRINTcipher .We identified low diffusion trails for any number of S-boxes involved, see Table 1 for examples of the most promising ones using up to eight S-boxes in a trail.

Increasing the number of S-boxes in the trail makes complexity estimation of the attack very complicated.

So we focused only on the case of three active S-boxes in the trail. All four possible trails gave very similar results.

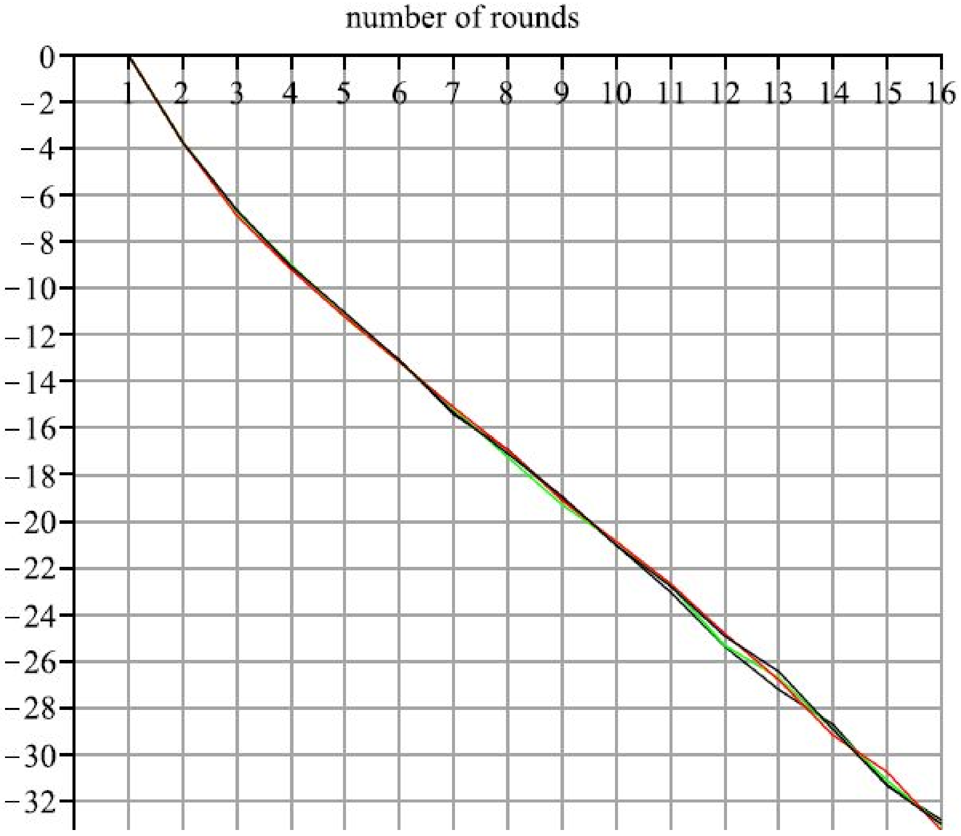
We estimated the bias for 50 randomly chosen keys for up to 10 rounds and for 20 randomly chosen keys for up to 15 rounds.

Below figures show the squared Euclidian distance between the distributions in the trail and the uniform distribution. The data complexity for attacking *n*+3, respectively *n*+ 4 rounds, depends on the number of key bits guessed and is nearly equal to the reciprocal of the square of Euclidian distance for *n* rounds. So, this shows us that more than 30 rounds of PRINT cipher can be broken using this attack.



S-boxes in the trail	{0, 1, 5}	{2, 6, 7}	{10, 14, 15}	{8, 9, 13}
Round 1	0	0	0	0
Round 2	-3.72	-3.72	-3.74	-3.73
Round 3	-6.67	-6.74	-6.89	-6.65
Round 4	-9.14	-8.98	-9.19	-9.05
Round 6	-13.08	-13.17	-13.17	-13.10
Round 8	-16.92	-17.25	-16.96	-17.10
Round 10	-21.02	-20.88	-20.87	-21.03
Round 12	-25.38	-25.33	-24.82	-24.93
Round 14	-28.72	-28.94	-29.19	-28.94
Round 16	-32.83	-33.05	-33.27	-33.00

Figure 1: Estimated squared distance (log2) for low diffusion trails with ratio 5⁄9



6 Conclusion

We have considered the technology of IC-printing in this paper and we have seen how it could impact the cryptography we use. We also specifically suggested the PRINT cipher lightweight block cipher, which directly takes advantage of this modern manufacturing strategy. Of course, it must be stressed that PRINTcipher-48 is intended not to be appropriate for deployment, but rather to be an object of study. It is also meant to be a catalyst for those who may be involved in this emerging technology being considered. We definitely agree that IC-features printing may be a fascinating line of work and we believe that it helps to highlight a variety of intriguing cryptographic design issues, most importantly how best to use a cipher key.

7 Parts done by members:

1) Rahul Kumar Mina (11840890): Abstract, Introduction, Conclusion

2) Prashant Kumar (11840820): Implementation

3) Meghna Singh (11840710): Security Analysis

8 References:

1) Knudsen L., Leander G., Poschmann A., Robshaw M.J.B. (2010) PRINTcipher: A Block Cipher for IC-Printing. In: Mangard S., Standaert FX. (eds) Cryptographic Hardware and Embedded Systems, CHES 2010. CHES 2010. Lecture Notes in Computer Science, vol 6225. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-15031-9_2

2) Cryptanalysis of PRINTcipher(a youtube video): https://youtu.be/wB7LxBLbBJo

3) Leander G., Abdelraheem M.A., AlKhzaimi H., Zenner E. (2011) A Cryptanalysis of PRINTcipher: The Invariant Subspace Attack. In: Rogaway P. (eds) Advances in Cryptology – CRYPTO 2011. CRYPTO 2011. Lecture Notes in Computer Science, vol 6841. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-22792-9_12