

Agence Nationale de la Recherche
ANR

DWP2.1

The Possibilities, and Limitations, of Cryptography in Constrained Devices

Date: 28.07.2009

Version: 1.0

J. Etrog*, M.J.B. Robshaw*, and O. Savry**, editors

*Orange Labs and **CEA-LETI

ANR RFID-AP Partners:

CEA-LETI, Eurecom, INRIA and Orange Labs

Deliverable for RFID-AP (ANR SESUR)

Table of Contents

1	Introduction	3
2	Components and Costs of RFID Tags	3
2.1	The power budget	4
2.1.1	The HF band	4
2.1.2	The UHF band	6
2.2	Analog components	7
2.3	Memory	9
2.4	Digital components	9
2.5	Manufacturing costs	10
2.6	Time cost	11
2.7	Cost summary	12
3	Overview of Cryptography Primitives	12
3.1	Symmetric primitive: The block cipher	12
3.1.1	A detailed example: PRESENT	13
3.2	Symmetric primitive: The stream cipher	15
3.2.1	A detailed example: Grain v1.0.	16
3.3	Symmetric primitive: The hash function	17
3.4	Symmetric primitive: The MAC	18
3.5	Asymmetric cryptography	18
4	Security of RFID Protocols	18
4.1	An example of a security model	19
4.2	Privacy properties	20
4.3	Authentication.	20
5	RFID Protocols	21
5.1	Hash based protocols	21
5.1.1	A detailed example: OSK	22
5.2	Protocols based on hard problems	23
5.2.1	A detailed example in public key: cryptoGPS	23
5.2.2	An detailed example in secret key: the HB family	25
5.3	Other crypto protocols	26
6	Future Directions	26
7	Conclusion	29

1 Introduction

The RFID-AP project is concerned with issues of authentication and privacy in the deployment of RFID tag-based applications. The security challenges in an RFID tag deployment are by now well-known, and the field of lightweight cryptography has developed massively in the last few years. To be clear on the scope of our work, we intentionally fix our focus to the low-cost framework defined by the EPCglobal standard.

In this report, we provide a detailed survey of the latest developments in lightweight cryptography, and the report falls into three different parts.

For the first part, we consider the technical background to lightweight cryptography. RFID tags are intended to be deployed widely and so they must be cheap. However, adding authentication or privacy solutions to RFID tags and/or readers consumes resources and this will increase the cost. The typical measure of space is the *gate equivalent*, the space on silicon that is occupied by a boolean NAND2 gate, and some early estimates in the literature suggested that no more than 2000 GE are available for security in low-cost RFID tags [70]. One goal of this report is to revisit this issue, and to make some detailed, contemporary estimates. Parts of the cost equation for an RFID tag deployment are non-technical and will depend on the application and the performance required for the different components. However a significant portion of the difficulties are due to purely technical issues, and it is on these factors that we will concentrate in this report. Thus, we will consider resources available to an EPCglobal system from the viewpoint of the available power; this is the over-riding technical limitation in passive RFID tags and after considering the analogue components, the memory, and the essential digital components, whatever power is left is available for additional luxuries such as security features.

The second part of the report focuses on the cryptography, and in particular on the cryptographic primitives. We survey the different advances in the field, and provide the latest information on some of the more promising developments in the field. In the third part we will consider how cryptographic primitives might be used and, in particular, we consider protocols for privacy and authentication that use cryptographic primitives. By looking at cryptographic progress in the second part, and by looking at the technical limits in the first part of the report, we can then assess whether any of the protocols that are proposed in the literature can really offer a contemporary viable solution to the problem of adding security measures to RFID tags.

2 Components and Costs of RFID Tags

An RFID tag consists of several components, each of which occupy some physical space and consume some power. A simple illustration is given in Figure 1 while a more complex version that will be needed for the power arguments is given in Figure 2. Since RFID tags are limited in their size and the amount of power that can be delivered by a reader, it is important to understand the limitations that

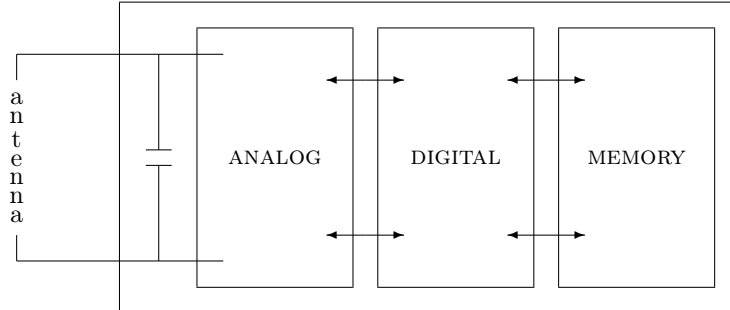


Figure 1: The general layout of an RFID tag.

these different components have on the overall design. It is, however, a difficult task to share out that part of the consumption that pertains the analog circuit, the digital circuit, and also the memory (particularly if EEPROM is used).

We use the available power budget as the basis for our estimates for the amount of space available on an RFID tag. The reason for this is that the maximum power that is available at a certain distance is, effectively, fixed. This gives us the maximum amount of power available to the tag and if we subtract the power consumed by the essential functional elements of the tags, then we arrive at the amount of excess power that might be available for added functionality. This can be converted into a number of logical gates for different manufacturing technologies.

2.1 The power budget

The power delivered by an RFID reader will depend on features of the reader such as its antenna characteristics, but also on the reading distance. While our focus is on EPCglobal applications and hence on UHF communications, it is illustrative to consider both the HF band, where antennas are inductive and absorb a magnetic flux, and the UHF band, where antennas are capacitive and absorb an electric field.

2.1.1 The HF band

In Europe electromagnetic emissions are regulated by the ETSI organization. For the HF (13.56 MHz) band, the power limits depend on EN 300-330 where a maximum magnetic field of 60 dB μ A/m at 10 meters is specified. This value implies a strong magnetic field near the reader. Indeed, the magnetic field $H(d)$ generated at the distance d by a coil antenna of radius R with a circulating current I and N turns is given by the well-known Biot and Savart law:

$$H(d) = \frac{N \times I}{2} \times \frac{R^2}{2(R^2 + d^2)^{\frac{3}{2}}}.$$

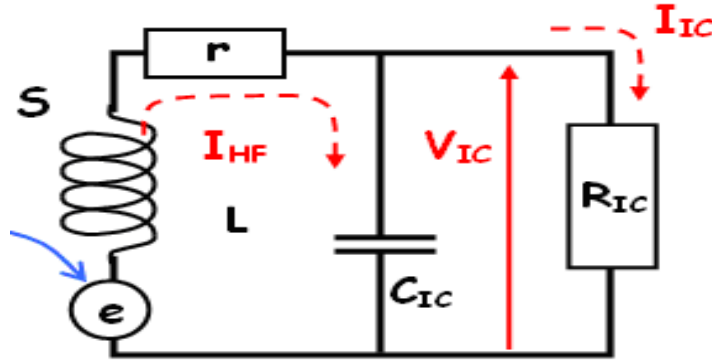


Figure 2: The electrical components of an RFID tag.

As an example, the legal threshold is reached with a current of 50 A in a classic antenna with a diameter of 40 cm and only 1 turn. However such a configuration is not practical and a tag closer than 50 cm to the reader would burn! In fact, tags are not designed to withstand more than 7.5 A/m, and so it is very misleading to look at the reader-side alone when considering the power that a reader can supply to a tag.

Instead we have to consider the tag, and a more restrictive parameter is given in the standards for so-called *vicinity cards* such as ISO 15693 and ISO 18000-3 which require that a card should still work with a 0.15 A/m magnetic field. The performance of the tag depends on a variety of factors:

- its coil with inductance L , turn section S , and resistance r ,
- its load R_{IC} ,
- its capacitance C_{IC} , adapted to be resonant with the inductance,
- the quality factor $Q_0 = \frac{L\omega}{r}$ of the antenna, and
- the quality factor $Q_{IC} = \frac{L\omega}{R_{IC}}$ of the tag.

With such information, we can calculate the power supplied to the tag P_{IC} by the reader using the following complicated expression:

$$P_{IC} = \left(\frac{\mu_0 S^2}{L} \right) \times Q_{IC} \times \left(1 + \frac{Q_{IC}}{Q_0} \right)^{-2} \times \omega \mu_0 H^2.$$

Using some typical values, such as a quality factor of 30 for a tag antenna and noting that a well-adapted tag would have $Q_{IC} \approx Q_0$, we can see that the power delivered to the tag under a 0.15 A/m magnetic field can be estimated to be about 2 mW. Such a magnetic field is delivered by a reader at a distance of 80 cm (with a 1 A current and a 40 cm diameter antenna, which are typical values).

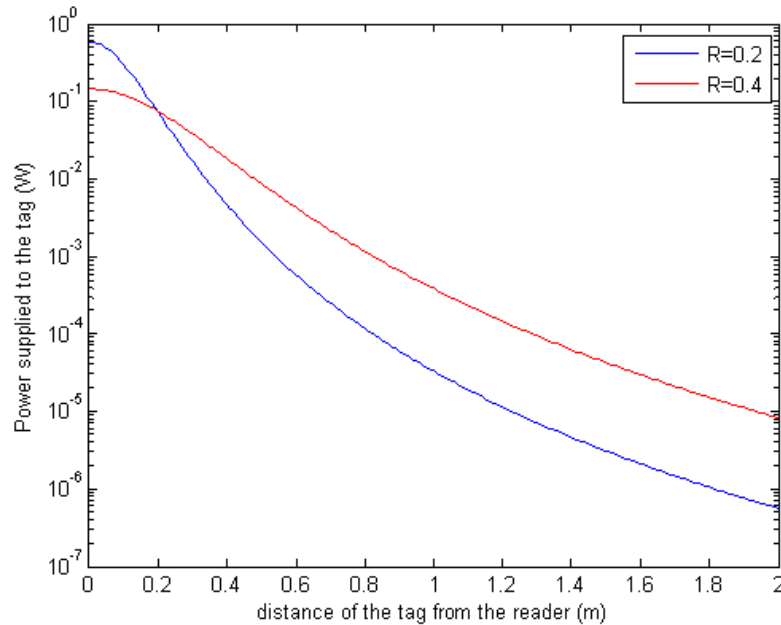


Figure 3: The power (W) supplied to an HF tag as a function of the distance (m) from the reader for two different antenna.

This estimate of 2 mW should be taken with some care since this does not mean that a tag will not be designed to function with less than 2 mW. If it can work with less power, it will work at a greater range.

To have an idea of the power available at a certain distance, we can consider the behavior of two antennas, one with R set to 0.4 m and the other with R set to 0.8m, and use the Biot and Savart law to obtain the results in Figure 3. Such a figure enables us to determine the power budget for a given maximum reading distance. This graph can be compared to the performances of some commercial products such as the NXP I CODE and Texas Instrument Tag-It which have power goals of 200 and 350 μW respectively with a range of a couple of metres. Generally speaking this matches the estimates in Figure 3 and can be taken as reference.

2.1.2 The UHF band

The UHF band (867 MHz in Europe) is also regulated by the ETSI organisation and the reader cannot emit more than 2 W. Using this figure for the maximum power emission of a reader, we can compute the power supplied to the tag $P(d)$

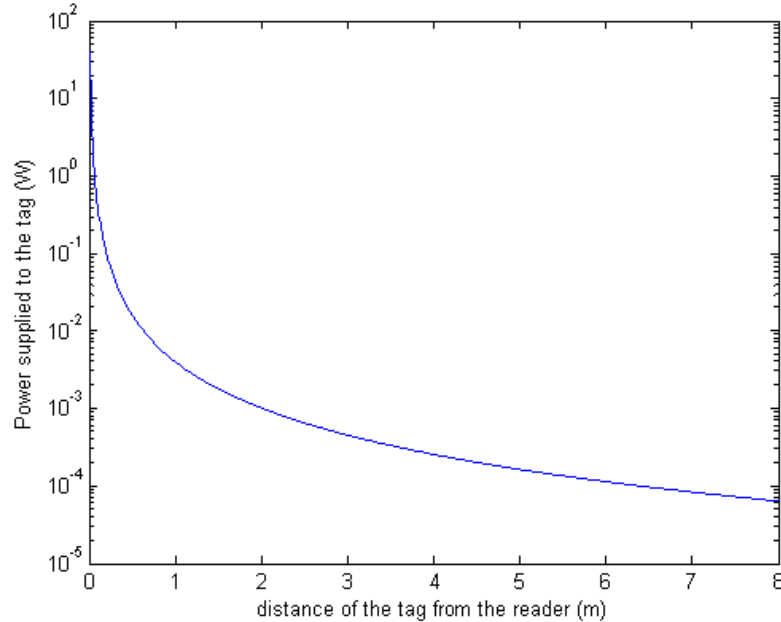


Figure 4: The power (W) supplied to a UHF tag depending on the distance from the reader (m).

at a distance d by means of Friis' formula :

$$P(d) = P_{bs} \times G_{bs} \times \left(\frac{\lambda}{4\pi d} \right)^2 \times G_{ant}$$

where P_{bs} is the power emitted by the reader through its antenna with gain G_{bs} and where G_{ant} is the gain of the tag antenna. Typical values are $G_{bs}=1.64$ for a dipole antenna and Figure 4 illustrates the behaviour of this power with the distance.

We notice that UHF can be more efficient than HF tag in terms of read-range since the decrease of the power supplied by the electric field follows a r^{-2} law whereas the power supplied by a magnetic field follows a r^{-3} law. This is reflected in the commercial products NXP U CODE and Texas Instruments RI tag where, indeed, UHF tags generally require less power (around $100\mu W$) and can be read at a greater distance than HF tags, around six to eight metres.

2.2 Analog components

The analog part consists of different functions, all of which are required for the basic functioning of the chip on the tag. These are essential components and cannot be avoided.

The resonant capacitor.

This capacitor is used to tune the antenna inductance to the resonance to improve the energy transfer from the reader to the tag. This capacitance can be high in the HF band from 30 to 100 pF. It has the drawback of generating losses of power but also of taking a lot of space on the circuit surface since a capacitor requires $5fF/\mu m^2$ in a 130 nm process. Thus, a capacitor in this process takes the surface of $0.1mm^2$ for a chip that is generally smaller than $1mm^2$.

In UHF band, this problem is less crucial since the resonance is not necessary. Nevertheless, an adaptation of impedance is often required in the circuit but it is assured by a specific design of the antenna. This fact explains, partially, why UHF tag chips are smaller than HF tag chips.

The rectifier, the charge pump, the voltage regulator.

All three components consume power. The rectifier rectifies the sinusoidal signal received by the antenna to give a constant voltage after filtering. This is an important part of the power consumption of the chip since filtering the harmonics of the rectified signal generates a loss of around 20% of the supplied current [49]. The charge pump that increase the voltage to feed the chip also generates source of loss due to current leakage in the diodes. Finally, the voltage regulator assures a constant voltage to feed the digital part no matter what magnetic field is emitted by the reader (within a certain range of course). This part also requires some filtering of the signal and so, as a consequence, wastes power.

The restoring of clock or the ring oscillator.

For HF tags, the clock that sets the digital sequencer is retrieved from the carrier of the signal that is sent by the reader. This circuit often requires an amplification of the carrier, but avoids the need to generate a clock from scratch. For UHF tags, however, the clock has to be built and while a PLL can be used, it is more efficient to generate the clock with a ring oscillator. The typical clocking frequency is around 100 KHz [117, 65].

The demodulator and the modulator.

The modulator modulates the load at the two terminals of the antenna. It will inevitably create a loss in the available power. Demodulation relies on the conventional approaches that involve peak and envelope detection together with some digital component implemented in logic.

On typical chips [49, 54, 62, 65, 117, 123], the analog and the digital part typically share roughly the same surface area of the die. Since power consumption is roughly proportional to the surface area, we might estimate that the power consumption of the chip can be divided in two equal contributions for the analog and digital parts. This is a rough approximation, but it suffices to remind us that not only the logic gates require power but also the analog elements.

2.3 Memory

A simple EPC tag only requires 128 bits of memory; 96 bits are used for the EPC code itself, the UID, while 32 bits are used for the `kill` password. This read-only memory will be fixed during the manufacturing process with a laser. Any memory that is required to enable the sequencer to monitor the data is designed with logical gates such as flip-flop registers. There may well be some applications that require read/write memory. If so, then EEPROM memory is mandatory, but this means that the manufacturing process is more expensive and considerably more power is required when writing to EEPROM [65].

2.4 Digital components

Even the most basic tag, the EPC tag, requires several digital components. The sequencer, interpreting the reader commands, building the answer, managing the anti-collision mechanism, and driving memory all rely on logical gates. This is all before we even begin to consider any additional luxuries such as security.

If we use the NAND2 gate as the typical unit of measurement, then the kind of digital components found on a basic RFID tag might occupy a space of up to 10,000 GE [49]. To estimate the number of gates that might be available to serve the digital components, we not only have to assess the available power but also understand the available energy during one clock cycle. This energy can be written with this following formula;

$$E_{\text{clock}} = \frac{P(d)}{f_{\text{clock}}}$$

where f_{clock} is the frequency of the clock of the digital part and $P(d)$ is the power supplied by the reader at the distance d (see Section 2.1). If we know the energy E_{GE} of the NAND2 gate then we can estimate the number of gates used by the digital part;

$$N_{\text{gates}} = \frac{E_{\text{clock}}}{E_{\text{GE}}} = \frac{P(d)}{f_{\text{clock}} \times E_{\text{GE}}}.$$

The energy E_{GE} can in turn be easily calculated by knowing the capacitance C of its CMOS transistors gates (there are four transistors per logical gate) and its supplied voltage V ;

$$E_{\text{GE}} = \frac{CV^2}{2}.$$

The values of C and V are dependent on the manufacturing process, and Table 1 give some values for some of the more common processes.

All the parameters for the equations are now known except the frequency of the clock. And here we need to choose our approach. By choosing a low clocking frequency we consume less energy and so, for the same fixed tag costs, our power budget allows us to support more gates. However, at the same time, higher-level protocol features might impose timing restrictions and so if we wish to add some processing, such as adding some cryptographic functionality, then there might

Process (μm)	Voltage (V)	Capacitance (fF)	Energy E_{ge} (fWs)
0.35	3.3	7.7	42
0.25	2.5	9.3	29
0.18	1.8	4.3	8
0.13	1.2	3.0	2

Table 1: Energy consumption E_{GE} for a NAND2 gate for different processing technologies.

Process (μm)	100 KHz			1 MHz		
	100 μW	15 μW	4 μW	100 μW	15 μW	4 μW
0.35	23,800	3,580	960	2,380	358	96
0.25	34,480	5,180	1,380	3,448	518	138
0.18	125,000	18,760	5,000	12,500	1,876	500
0.13	500,000	75,000	20,000	50,000	7,500	2,000

Table 2: The area available to digital components N_{gates} for different manufacturing processes, power budgets, and clocking frequencies. Typical passive RFID systems are clocked at 100 KHz and assumed to have around 15 μW of power available for added functionality.

not be the physical time to do so. To avoid this, we would need to increase the clocking frequency which, in turn, impacts our silicon budget.

We will consider this issue more in Section 2.6. Here however we will consider the typical clocking frequency of 100 KHz for EPC tags, as outlined in Section 2.2, as well as a higher clocking frequency of 1 MHz to illustrate its impact. In Table 2 we give an estimate for the number of gates that are available for different power budgets and different processes. Note that the 100 μW refers to the power arriving at the tag and most of this is consumed by all the essential analog and digital components. The power that is left is often assumed to be around 15 μW or even as low as 4 μW [37], and figures for these possibilities are given in Table 2. The values obtained are, inevitably, crude and the result of multiple approximations. They should therefore be taken with care. However the figures derived in this way do match other estimates in the field but, more importantly, they have been derived from purely technical and not economic arguments. We will now turn to some of the economic factors.

2.5 Manufacturing costs

In academic papers one often sees the target cost of an RFID tag being quoted as being around 10 dollar cents or even 5 dollar cents. However, this does not match current market realities with prices for EPCglobal tags being around 30–40 dollar cents for a large quantity. Some of the factors that influence the cost, and which are unlikely to go away in the short term, are considered below.

Silicon surface.

For a given process, the cost of a chip is proportional to the used surface of silicon. However, RFID tags are so small, for instance $300\text{ nm} \times 300\text{ nm}$ for the Hitachi μchip [117], that elements that are generally considered insignificant become significant; even the lines of the saw and the surface of the pads for the antenna or testing have to be taken into account. Moreover, roughly half the active surface consists of analog electronics. This means that the impact of an increase in the number of logical gates is somewhat reduced. Thus adding 2000 or 3000 logical gates might well not raise the tag price significantly. And since the surface of the analog part is unlikely to shrink in the same way that the digital part will, see Section 6, then the incremental cost of adding more logical gates can be expected to fall over time.

Design costs.

The introduction of cryptographic functionality, be it an algorithm or a protocol, will involve considerable design costs; developing the VHDL code and optimising the layout is expert work. Further, products providing security functionality are likely to require security certification, which avoids errors and provides reassurance to the customer, but increases the cost and the duration of the design and manufacturing cycle.

Inlay and packaging.

The price of a tag cannot be reduced beyond the price of the chip. But as well as the silicon area and design costs, a tag needs to be put together. As a result, the cost of the inlay needs to be taken into account. The inlay consists of the antenna and its substrate as well as the packaging and integration of the necessary components. Generally this is a flip-chip process, though particularly small tags might need an innovative process such as the Hitachi μchip [117] or that used in Alien FSA technology [1]. Currently, screen printing or ink jet printing technologies would allow a reduction in price, perhaps to around 6 dollar cents for a UHF inlay [113]. But unfortunately this technology doesn't currently yield compact designs.

2.6 Time cost

To make our estimates on the space available on-the-tag, we assumed a clocking frequency of 100 KHz. This is a typical value. However, it imposes some severe constraints if we wish to add some additional functionality to the tag.

To see this, consider the EPCglobal standard which requires that a tag respond to a reader command, a query for example, within $73.1\mu\text{s}$. At 100 KHz this corresponds to less than 10 clock cycles. There is not a lot that can be done in 10 clock cycles. For instance, depending on the implementation, see Table 3, DES will require much longer than that, as will Triple-DES [93]. To match our time constraint, a clock cycle should be smaller than $73.1\mu\text{s}/54 = 1.35\mu\text{s}$ which corresponds to a frequency of 738 KHz. Of course, other algorithms are faster,

for instance PRESENT requires 32 cycles and which corresponds to a frequency of 437 KHz. However, it is clear that the time constraint is important and if we want to add cryptographic functionality then this can impact the clocking frequency. In order to support additional algorithms, a frequency of 500 KHz or 1 MHz might be required, with the impacts that this would have on the power and space available.

The other option is to either (i) change the communication protocol or (ii) introduce some higher-level mechanism, as is proposed in [35], to share out the burden of a cryptographic computation across numerous communication slots.

2.7 Cost summary

The conclusion of our study in this first part of the report is somewhat mixed. When using older manufacturing processes, as is done now for RFID tags, then there is unlikely to be sufficient power budget available to do much, if any, cryptographic processing. For current generation RFID tags this means that no matter how much money one has available to make a larger chip, there won't be the power to run the additional functionality.

On the other hand, we can see that the situation changes rapidly as we move to smaller manufacturing processes and the power budget works in our favour. Thus, tags that are manufactured with more recent technologies shouldn't be limited purely by the available power and, instead, the limits to the amount of space on an RFID tag in the future are likely to be purely economical.

3 Overview of Cryptography Primitives

There are numerous sources for a basic overview of cryptographic primitives; among the best is the Handbook of Applied Cryptography [87]. In this report our focus is on the hardware costs of implementing cryptography, and so we will tend to avoid the detailed theoretical descriptions of these primitives and, instead, provide a brief description along with their performance characteristics.

As a starting point, it is typical to divide the class of cryptographic primitives according to how they use key material. When using symmetric cryptography, the participants in a cryptographic exchange share the same key material. In some cases, for instance when using a hash function, there is no key material and this situation is typically considered within the same symmetric classification.

3.1 Symmetric primitive: The block cipher

These are perhaps the most important building blocks since a good block cipher can be used to build all the other symmetric primitives. At the same time, our understand of how to design block ciphers is reasonably advanced. A block cipher encrypts a b -bit block of data under the action of a k -bit secret key, and as such it instantiates a family of permutations that are indexed by the key.

Table 3: The implementation results for a variety of hardware-oriented block ciphers. The throughput is measured when clocked at a typical RFID tag frequency of 100KHz.

	Key size	Block size	Cycles / block	Through. (Kbps)	Logic μm	Area GE
PRESENT-80 [108]	80	64	563	11.4	0.18	1,075
PRESENT-80 [15]	80	64	32	200	0.18	1,570
PRESENT-128 [15]	128	64	32	200	0.18	1,886
AES-128 [35]	128	128	1032	12.4	0.35	3,400
HIGHT [63]	128	64	1	6400	0.25	3,048
mCrypton [82]	96	64	13	492.3	0.13	2,681
Camellia [3]	128	128	20	640	0.35	11,350
DES [73]	56	64	144	44.4	0.18	2,309
DESXL [73]	184	64	144	44.4	0.18	2,168
TEA [125]	128	64	64	100	0.18	2,355

There have been considerable advances in the design and implementation of block ciphers in recent years, and some of the results are presented in Table 3 which is reproduced from [105]. As can be seen, the most promising current proposal is the block cipher PRESENT, and so with this in mind we provide some additional details on the design of this cipher.

3.1.1 A detailed example: PRESENT

The block cipher PRESENT was designed during a joint research project between Technical University Denmark, Ruhr-University Bochum and France Télécom R&D. The cipher was designed from scratch to be efficient to implement in hardware, and has numerous regularities and design features that help in this respect. The block length is 64 bits and while two key lengths of 80 and 128 bits are supported, only the 80-bit version is expected to be used for low-cost applications. Each of the 31 rounds of the cipher consists of an XOR operation to introduce a round key, a linear bitwise permutation and a non-linear substitution layer. The non-linear layer uses a single 4-bit S-box S which is applied 16 times in parallel in each round. The action of this box in hexadecimal notation is given by the following table.

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

Using a bit permutation for the linear mixing is very advantageous for hardware implementation and the one used in PRESENT is very regular.

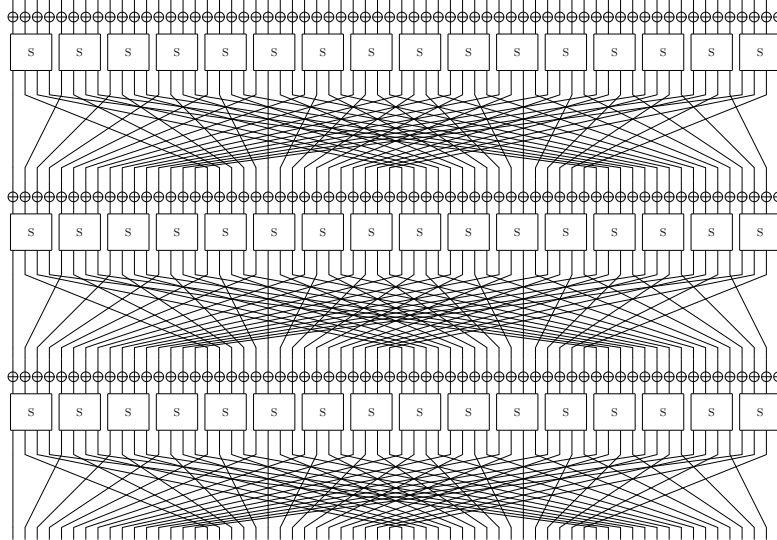


Figure 5: . Three rounds of PRESENT illustrating the role of the S-boxes and the bitwise permutation.

The key schedule for PRESENT is very lightweight, and more details about this along with all the design criteria and justifications, can be found in the original paper [15]. PRESENT has now been implemented and synthesized in several ways, and a variety of trade-offs exist. It's often interesting to look closely at the implementation and to identify the bulk of the hardware cost.

module	GE	%
data state	384.39	24.48
s-layer	448.45	28.57
p-layer	0	0
counter: state	28.36	1.81
counter: combinatorial	12.35	0.79
other	3.67	0.23
sub-total	877.22	55.88
KS: key state	480.49	30.61
KS: S-box	28.03	1.79
KS: Rotation	0	0
KS: counter-XOR	13.35	0.85
key-XOR	170.84	10.88
sub-total	692.71	44.12
total	1569.93	100

As can be seen, the bulk of the area is occupied by flipflops for storing the key and the data state, followed by the S-layer and adding the key. While the main goal is often a small hardware footprint, power-optimized implementations exist and for around 1,625 GE the power consumption is only $3.3\mu\text{W}$.

3.2 Symmetric primitive: The stream cipher

Just as in the field of block ciphers, there have been considerable recent advances in the design of stream ciphers. In contrast to block ciphers, the field of stream ciphers is very fragmented and the design of stream ciphers is acknowledged to be challenging.

The difficulty of designing a secure stream cipher is, to a great extent, a function of the way it operates. As well as a key the cipher typically uses an initialisation value. Thus the cipher can be repeatedly initialised with different IVs that are known to an attacker while the secret key remains unchanged. Also, the key and IV are typically used to initialise the state of the stream cipher, and after this time the state evolves without any influence from the secret key. These are very special attributes to a cipher and it is not surprising that they lead to very special design demands.

The eSTREAM project was part of the ECRYPT Network of Excellence, and the project ended in April 2008 with the publication of a range of promising new stream cipher proposals. In particular, three portfolio ciphers are intended for hardware efficient implementation, and, when looking beyond the space requirements, we find that these algorithms are particularly amenable to low-cost hardware implementation. Here we focus on Grain v1.0 which offers very good implementation flexibility [53] including some low-power implementations [34].

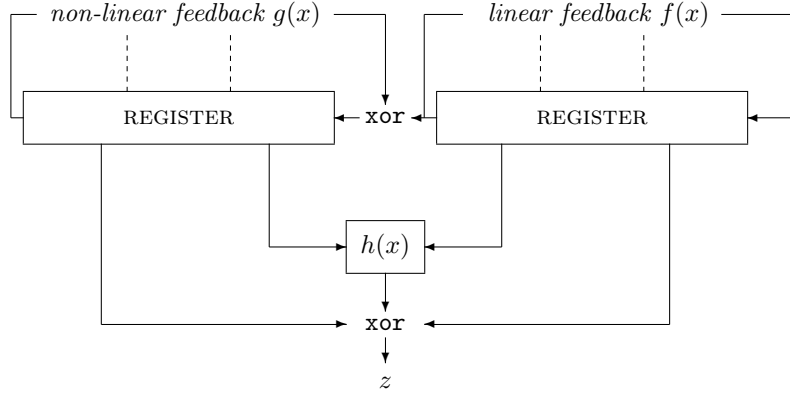


Figure 6: . An overview of the stream cipher Grain v1.0.

3.2.1 A detailed example: Grain v1.0.

Grain v1.0 consists of two feedback shift registers, one of which uses linear feedback while the other uses non-linear feedback. Grain v1.0 offers 80-bit security, though there is a variant—Grain-128 [58]—that offers 128-bit security as the name implies. A schematic overview of Grain v1.0 is given in Figure 6.

Each register is 80 bits in length and the linear feedback $f(x)$ and the non-linear feedback $g(x)$ are defined as follows;

$$\begin{aligned}
 f(x) &= 1 + x^{18} + x^{29} + x^{42} + x^{57} + x^{67} + x^{80} \\
 g(x) &= 1 + x^{18} + x^{20} + x^{28} + x^{35} + x^{43} + x^{47} + x^{52} + x^{59} + x^{66} + x^{71} + \\
 &\quad x^{80} + x^{17}x^{20} + x^{43}x^{47} + x^{65}x^{71} + x^{20}x^{28}x^{35} + x^{47}x^{52}x^{59} + \\
 &\quad x^{17}x^{35}x^{52}x^{71} + x^{20}x^{28}x^{43}x^{47} + x^{17}x^{20}x^{59}x^{65} + x^{17}x^{20}x^{28}x^{35}x^{43} + \\
 &\quad x^{47}x^{52}x^{59}x^{65}x^{71} + x^{28}x^{35}x^{43}x^{47}x^{52}x^{59}.
 \end{aligned}$$

The combining function $h(x)$ is a boolean function that takes five input bits from the two registers, namely positions 3, 25, 46, and 64 from the linear feedback shift register and position 63 from the non-linear register. It has been carefully chosen to provide particular cryptographic properties, more details can be found in [59, 57], and has the form;

$$h(x) = c_1 + c_4 + c_0c_3 + c_2c_3 + c_3c_4 + c_0c_1c_2 + c_0c_2c_3 + c_0c_2c_4 + c_1c_2c_4 + c_2c_3c_4$$

where the variables c_0 , c_1 , c_2 , c_3 and c_4 correspond to the five tap positions given above.

The hardware performance of Grain v1.0 has been studied closely in a variety of papers. One particularly nice feature is the wide-ranging performance profile. In [53], for instance, it is shown that implementations oriented towards RFID

Table 4: An overview of the performance of some current hash functions ordered by the size of the output. The throughput is estimated at a clocking frequency of 100 KHz.

	Hash size	Cycles / block	Through. (Kbps)	Efficiency (bps/GE)	Logic μm	Area GE
MD4 [36]	128	456	112.28	15.2	0.13	7,350
MD5 [36]	128	612	83.66	10.0	0.13	8,400
SHA-1 [97]	160	344	148.81	26.9	0.13	5,527
SHA-1 [97]	160	344	148.81	24.3	0.18	6,122
SHA-1 [36]	160	1,274	40.18	5.0	0.35	8,120
SHA-256 [36]	256	1,128	45.39	4.2	0.35	10,868
MAME [124]	256	96	266.67	32.9	0.18	8,100

tags, where reduced space is the driving metric, offer implementations requiring around 1,200 GE. Figures in [34] are derived from implementations that strive to minimize energy consumption; here the size might increase to 3,360 GE but with an electric current of merely $0.80\mu\text{A}$ when clocked at 100 KHz.

3.3 Symmetric primitive: The hash function

Informally, a cryptographic hash function takes an input of variable size and returns a hash value of fixed length while satisfying the properties of preimage resistance, second preimage resistance, and collision resistance [87]. For a hash function with n -bit output, compromising these should require 2^n , 2^n , and $2^{n/2}$ operations respectively. These properties make hash functions very appealing in a range of protocols.

As we will see in Section 5.1, many new protocols for RFID applications assume that a cryptographic hash function is used on the tag. However which hash function might be used in practice is rarely identified and, in fact, current hash functions are not at all suitable for constrained environments. They require significant amounts of state and the operations in current dedicated designs are not hardware friendly.

The design of a secure hash function is one of today's significant cryptographic problems. Rather than building a hash function from scratch, it is possible to build one from a block cipher, and work in [16] described how to use the compact block cipher PRESENT [15] as the basic building block. Work in that paper gave the implementation of two trusted constructions, *Davies-Meyer* and *Hirose*, that use PRESENT to give 64- and 128-bit outputs. The hardware costs of these constructions are given in Table 5

Table 5: The performance of different hash functions based on the direct application of PRESENT. All designs used $0.18\mu\text{m}$ technology and the throughput is estimated at a clocking frequency of 100 KHz.

	Hash size	Cycles / block	Through. (Kbps)	Efficiency (bps/GE)	Area GE
DM-PRESENT-80	64	33	242.42	109.5	2 213
DM-PRESENT-80	64	547	14.63	9.1	1 600
DM-PRESENT-128	64	33	387.88	153.3	2 530
DM-PRESENT-128	64	559	22.9	12.1	1 886
H-PRESENT-128	128	32	200	47	4 256
H-PRESENT-128	128	559	11.45	4.9	2 330

3.4 Symmetric primitive: The MAC

The *message authentication code* or MAC is a cryptographic checksum and it has a wide variety of authentication applications. In the literature, MACs are typically built out of a block cipher, *e.g.* *CBC-MAC* [94], or a hash function, *e.g.* *HMAC* [95]. However, if we build a MAC from a hash function then we inherit all the hardware disadvantages of a hash function. So instead it is likely that we would build a MAC out of a low-cost block cipher.

There are dedicated designs for low-cost MACs. Many of these are proprietary but some, such as SQUASH are public and have been proposed for use in RFID tags [112]. However in the case of SQUASH its not clear how to choose the best parameter sets so as to get good performance and security [104].

3.5 Asymmetric cryptography

Asymmetric, or public key, cryptography is built around the problems of mathematical hard problems [87]. The most commonly used schemes, such as those that depend on factorisation, discrete logarithms, or elliptic curve discrete logarithms are unsuited to RFID tag deployment. Even the simplest components for the schemes that are based on these problems will require many tens of thousands of logical gates. However, as we will see in Section 5.2 there are some possibilities to provide public-key functionality to RFID tags at a very low cost if we use the technique of *coupons*.

4 Security of RFID Protocols

In order to capture the requirements on cryptographic protocols in proper security definitions, one has to specify: (1) the goal of an adversary: in the case of an RFID protocol this goal can typically consist in defeating the correctness, or the security (by means of an impersonation attacks), or the privacy of the protocol.

One thus has to build an adversary model for each desired security property; such models are introduced for example in [118, 69], and (2) the resources of an adversary, e.g. whether it is a passive or active or "man-in-the-middle" adversary or more generally which kind of actions it can initiate, how many times it can interact with a target RFID tag and/or a legitimate reader, which amount of computing resources it can use for the attack.

If we want to prove that a cryptographic protocol, e.g. an RFID identification protocol, is secure under some of the above security definitions, we have to make assumptions on the security of the underlying cryptographic primitive(s). In other words, we have to relate the security of the protocol to the security of the underlying primitive(s) by means of a security reduction. The primitive has thus to satisfy some resistance in a certain security model. For example we want hash functions to achieve pre-image resistance or collision resistance or we want block ciphers to be indistinguishable for a random permutation under known plaintext attacks. We use such assumptions to prove the security of the protocol. This is why it makes sense to use known robust cryptographic primitives instead of ad hoc toy ciphers built from scratch.

4.1 An example of a security model

We give an example of a detailed security model developed in [118]. This security model is defined by first introducing the following types of actions.

- Create a legitimate or illegitimate tag.
- Make a set of tags available for acting on.
- Open a new protocol with the reader or some tags.
- Send messages to the reader or the tag and collect the answer.
- Access to the result of a protocol between the reader and a legitimate or illegitimate tag.
- Corrupt a tag, i.e. open a tag to access its internal state.

Then we define the different classes of adversaries.

- Strong is the class of adversaries who have access to all oracles.
- Destructive is the class of adversaries who never act on a tag after corrupting it.
- Forward is the class of adversaries who, after corrupting a tag, can only continue corrupting tags.
- Weak is the class of adversaries who never corrupt tags.

All these classes of adversaries define distinct embedded security classes as follows: a protocol is said to resist a given adversary if the advantage of this adversary is negligible as compared with an adversary who uses no action. Each of these four classes can be split into two subclasses by introducing the subclasses Narrow-Strong, Narrow-Destructive, Narrow-Forward, and Narrow-Weak, which represent adversaries who never access acceptance or rejection results and we can also prove that each of the two obtained groups of four subclasses forms distinct embedded security classes. It can also be proven that some subclasses cannot be reached, for example $\text{Destructive} \cup (\text{Narrow} \cap \text{Strong}) = \emptyset$.

4.2 Privacy properties

We recall here the different possible goals of a protocol in terms of security level in an intuitive manner (formal definitions require a formal description of the adversary model).

Anonymity

Anonymity is the minimum privacy level of an identification protocol, it means that no adversary is able to derive the identity of the tag using the information provided by the identification protocols in which the tag is involved.

Unlinkability

Unlinkability means that given several executions of the protocol involving at least two tags, no adversary is able to select two executions of the protocol which are related to the same tag with a good probability.

Forward secrecy

Forward secrecy means that given several executions of the protocol involving at least two tags and then, the internal state of one of these tags, an attacker cannot find a past execution of the protocol which is related to this tag with a good probability.

We recall here the obvious implications

$$\text{Forward secrecy} \Rightarrow \text{Unlinkability} \Rightarrow \text{Anonymity}$$

4.3 Authentication.

From a security point of view, in many applications the adversary does not need to find the identity of a tag, it only needs to identify himself as a legitimate tag or to impersonate a legitimate tag (this is called DoS attack i.e. denial of service attack). An identification protocol allowing the reader to corroborate the identity of the tag and to check that it is legitimate is called an authentication protocol. Authentication can also be extended to mutual authentication where both parts can trust each other. Of course it is possible to combine authentication with the

previous privacy properties and some weaknesses with respect to one property can affect the other, see for example the generic attack given in Section 5.1.1.

Theoretical analysis of the compatibility of privacy goals under different classes of adversaries and/or some kinds of authentication can be found for example in [118]. It is also possible to link some privacy properties with the cryptographic setting, it is proved in [14] that in private key cryptography it is not possible to achieve both DoS resistance and forward secrecy whereas in [118] it is proved that for two-round protocols Narrow-Strong privacy requires public-key encryption algorithm.

5 RFID Protocols

In this section we will classify the RFID protocols according to the underlying cryptographic primitives. Some of these primitives are adapted to the environment of RFIDs, and some are used as-is, without much improvement, though usually with carefully selected parameters.

5.1 Hash based protocols

Hash based protocols represent a large fraction of RFID protocols, due to the useful one-wayness property of hash functions. Hash based protocols are designed to achieve privacy. We can point out a few, important historically in the design of RFID protocols based on hash functions and then we will describe fully one of these protocols, OSK.

- [110] was the first to present an authentication protocol for RFID which preserves anonymity.
- [119] used a random value to achieve unlinkability and replay attacks resistance.
- [99] used an update for the internal state using another hash function to achieve forward-secrecy.
- [8] used a random value to achieve resistance against replay attacks.
- [21] showed how to achieve resynchronization (and thus resistance against DoS attacks) via mutual authentication.

There were numerous different attempts, using counters, timestamps, . . . , but there seem to be inherent weaknesses in the use of timestamps.

Much work has also be done on trying to improve the efficiency of protocols, we can mention [9, 122] among others.

We show on Figure 7 a survey on hash based RFID protocols and their links using the notation given in Table 6.

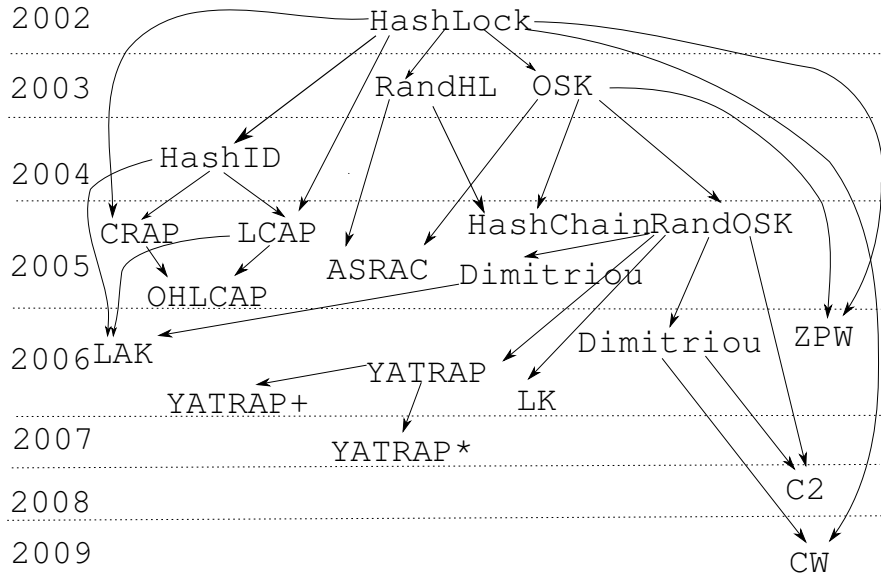


Figure 7: A schematic family tree of hash-based protocols.

5.1.1 A detailed example: OSK

We describe here the OSK protocol by Ohkubo *et al.* [99]. OSK is a privacy preserving identification scheme that uses two hash functions G and H . Each tag has a state s that is updated after each identification as $s_{i+1} = H(s_i)$. The original state s_0 of the tag is known to both the reader and the tag, thus evolving this state is easy for both entities. Identification is done by simply sending $a_i = G(s_i)$ and updating the state as previously shown. It is important to note that the two hash functions G and H do not need to be completely distinct: it is possible to derive them from a single hash function, for instance by simply padding out the input by 0-s for G and by 1-s for H . Eavesdroppers cannot reverse the hash function G to the state s_i , and therefore cannot predict s_{i+1} . Furthermore, even if an attacker can break into a tag and discover s_i , earlier tag identifications cannot be linked to this value as the function H is one-way. Thus the protocol achieves forward secrecy. A drawback of the OSK scheme is that any active attacker with sufficient time can simply repeatedly identify the tag n , times thereby evolving the tag to such a state that it becomes exceedingly hard for the reader to find the state s_{i+n} for each and every tag. Therefore, the tag becomes unable to perform identification, this is a generic DoS attack which can be used as a privacy threat in an adversary model given in [69] where the active adversary can access the output of the protocol by the reader as follows:

- The adversary selects two tags T_i and T_j .
- It challenges T_i n times.

Protocol	Reference	Protocol	Reference
HashLock02	[110]	OHLCAP05	[24]
RandHL03	[119]	Yatrap06	[114]
OSK03	[99]	ZPW06	[126]
HashID04	[61]	Dimitriou06	[29]
CRAP05	[107]	LAC06	[75]
LCAP05	[76]	YATRAPH+06	[20]
RandOSK05	[8]	LK06	[81]
HashChain05	[122]	YATRAPH*07	[115]
Dimitriou05	[28]	C208	[21]
ASRAC05	[78]	CW09	[23]

Table 6: Reference of protocols

- Then given one of the two tags T_b at random, the adversary submits it to a protocol with the legitimate reader, if the reader accepts the tag the adversary concludes that $T_b = T_j$ and if the reader rejects the tag the adversary concludes that $T_b = T_j$.

5.2 Protocols based on hard problems

In both secret and public key cryptography hard problems are a natural way to build zero-knowledge challenge-response protocols authentication protocols.

The idea of using public key cryptography in RFID tags was as old as [67] and many different schemes have been used as a basis of protocols. The Okamoto protocol [100] was derived in several protocols [11]. The Schnorr protocol [111] was also derived in several protocols [116, 77, 19]. The WIPR protocol [101, 102] based on the randomized variant of the Rabin cryptosystem, and its variants [121]. In [51] a technique named Universal Re-encryption was used to permit re-randomization of ciphertexts without knowledge of the corresponding private key and a RFID protocol based on ElGamal using this technique was proposed. Variants of this protocol were proposed in [109] to avoid some attacks point in [51]. We will give a detailed example of one of these protocols : the cryptoGPS protocol.

They are fewer secret key protocols based on hard problems; in [71] a protocol based on the difficulty of recovering the multiplicand or multiplier from the product of matrix multiplication is given and in [4], the authentication protocol is proven secure by reduction to the MQ-problem. Here, however, we will give a detailed example of the HB-family of protocols that are based on the LPN problem.

5.2.1 A detailed example in public key: cryptoGPS

The cryptoGPS public-key scheme is due to Girault, Poupard, and Stern [43, 47] and it has long been proposed for RFID tag authentication. One particular

Tag	Reader
PARAMETERS	
Curve \mathcal{C} , point P	Curve \mathcal{C} , point P
KEYS	
Secret $s \in_R \{0, 1\}^\sigma$ Public $V = -sP$	Public $V = -sP$
COUPON PRE-COMPUTATION WITH PRNG	
For $0 \leq i \leq t - 1$	
Let $r_i = \text{PRNG}_k(i)$ where $ r_i = \rho$	
Set $x_i = \text{hash}(r_i P)$	
Store coupon x_i	
PROTOCOL USING ON-TAG PRNG	
At time i fetch x_i	$\xrightarrow{x_i}$
Generate $r_i = \text{PRNG}_k(i)$	\xleftarrow{c} Pick $c \in_R \{0, 1\}^\delta$
$y = r_i + (s \times c)$	\xrightarrow{y} $\text{hash}(yP + cV) \stackrel{?}{=} x_i$

Figure 8: The elliptic curve variant of cryptoGPS. The parameters ρ , δ , and σ denote three particular bit lengths offering a range of security/performance trade-offs.

optimisation is the use of *coupons* which permit the RFID tags to avoid doing any number theoretic operation apart from the most basic [44]. Essentially, coupons are pre-computed data that are used once and excessive computation has been reduced at the cost of space on the tag. While the coupons are gradually used up, it is notable that this fits the established model for many RFID tag applications where tags are used several times, perhaps at different hops in the supply chain, and they are thrown away. However, for those that want to use the RFID tags for longer, there are external means to replenish the stock of the coupons.

There are several variants of cryptoGPS, but it is typically to restrict attention to the elliptic-curve based variant, as this is the one promising to be implemented in RFIDs. This is described in Figure 8, where h denotes the length of the cryptographic hash function $\text{hash}()$, and t denotes the number of coupons stored on the tag. The suggested parameter sizes are $|s| = \sigma = 160$ to provide 80-bit security and bit-length parameters σ and δ vary between $\sigma \in \{128, 160\}$ and $\delta \in \{8, 20, 32\}$ depending on what authentication security ($2^{-8}, 2^{-20}, \dots$) is needed from the protocol. The parameter ρ is set to $\rho = \sigma + \delta + 80$.

The performance of cryptoGPS has been examined in a series of papers [85, 86, 45]. There are several optimizations to the basic scheme [48, 46] and the implementation cost of the final cryptoGPS can be as little as 317 GE. However, a PRF is required and this increases the footprint by about 1000 GEs to give a total of about 1500 GEs. This is well within the notional barrier for cryptographic functionality, and the coupons can be stored in additional memory.

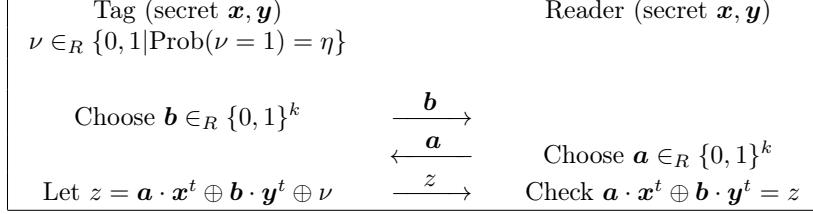


Figure 9: One round of the HB+ protocol.

5.2.2 An detailed example in secret key: the HB family

Introduced in [64], the HB protocol aims at authenticating RFID tags to a reader using lightweight operations (matrix multiplication and xor) while reducing its security to a well-known NP-hard problem: the learning parity with noise (LPN) problem [13]. But it was shown in [70] that HB is insecure against adversaries able to interact with tags. So in [70] HB+ which resists to impersonalization attack was proposed. Although these two protocols were initially studied in a scenario of a sequential executions, [72] extended both security proofs to a more general concurrent and parallel setting. In [42] it was shown that the security of HB+ is compromised if the adversary is given the ability to modify messages going from the reader to the tag (later known as the GRS security model). Many variants of HB were proposed as in [18, 91, 30] but all of them were proven to be insecure in the GRS model in [39]. Finally in [40, 41] a new variant Random-HB# and the optimization HB# were proven to be secure in the GRS model. But it was proven to be insecure if the attacker can modify the messages in both direction (called the MIM model) in [103]. Some current variants like [55] also failed to be secure in the MIM model while on the other side [17] is proved secure in the MIM model but uses a hash function which is sufficient to allow authentication.

We give a detailed description of HB+ where each tag has a secret (x, y) of two k -bit numbers, the protocol is composed with r following rounds : the tag generates a random k -bit value a and send it to the reader. Then the reader sends back a random k -bit value b . The tag selects a bit ν with a probability η to be equal to 1 and sends to the reader the bit $a \cdot x \oplus b \cdot y \oplus \nu$. At the end of the r rounds, the reader decides if the tag is legitimate or not.

We compare the security of both schemes, as the security of HB# is related to k_X, k_Y, m, η and t , k_X needs to be N bits to achieve N -bit security where k_Y is linked to the difficulty of the LPN problem. Some sets of parameters with different noise levels η were proposed : (80, 512, 1164, 0.25, 405) and (80, 512, 441, 0.125, 113) offer 80-bit security, false acceptance and rejection rates less than 2^{-80} and 2^{-40} respectively and communication requirements around 1500 bits. Similar parameters give error rates of 2^{-1} and 2^{-20} and transmission costs up to 48000 bits for HB+. The inconvenient of HB# which is his storage cost of $(k_X + k_Y + 2m - 2)$ bits comparing to HB+ one of $2k$ bits.

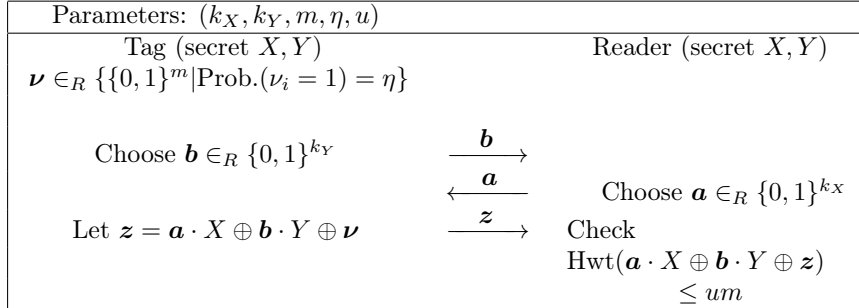


Figure 10: The RANDOM-HB# authentication protocol where the secrets X and Y are binary random matrices and the protocol has a single round. The verification step requires the comparison of two vectors and yields a PASS/FAIL verdict. For HB#, the random matrices are replaced by Toeplitz matrices [40].

5.3 Other crypto protocols

Some protocols use different primitives, for example S-protocol uses a *pseudo-random number generator* [74] while in [79] a protocol based on integer multiplication is presented and [26] uses Hamming distance as the basis of the protocol. Another family of protocols are tree-based protocols like [89, 96, 88] which authenticate a tag in $\log n$ operations using a tree structure. There is also an example of a protocol using universal hash functions [14].

6 Future Directions

The goal of this report has been to consider the current state of cryptography and RFID tags. However it is interesting to consider how then situation might change in the coming years.

First, when considering the future performance of digital technology it is typical to call upon the famous Moore's Law [90]. Indeed, in the realm of lightweight cryptography, some commentators have suggested that simply by waiting for Moore's Law to take hold, we will soon have all the computing power we might want on an RFID tag. On a large enough time-frame, this is probably true. However over the coming years it is not so clear-cut.

Moore's law is typically interpreted as suggesting that the density of transistors in an integrated circuit per unit cost doubles every two years, and not every 18 months as it is often claimed. Remarkably, this observation still holds and RFID tags will also take advantage of this progress. However, they will do so with a significant time-lag since RFID tags are only now using $0.13 \mu m$ process while the broader microprocessors industry is beginning to put $0.045 \mu m$ technology into production.

Thus the 0.09 , 0.065 , and the $0.045 \mu m$ technologies are not yet used to

develop tag chips and there are several reasons for this delay of three or four generations. First, new manufacturing processes are primarily used for pure digital components. Adding different kinds of memories and analog and radio-frequency electronics introduces an automatic delay of around three nodes. This is particularly the case since RFID tags are circuits that have a poor added-value when compared to microprocessors. There is no point dedicating the latest fabrication technology to a product that has such small profit margins. Thus RFID tags will only be developed in factories that are already profitable, or ones that have been left behind by market demands for newer and faster technologies. Indeed, another important law in the semiconductor industry, Rock's law, suggests that the cost of semiconductor chip fabrication plant doubles every four years. Economic arguments demand that a plant be made profitable as soon as possible and this is best done by producing circuits with high added-value. Once, those circuits can be improved with a new node, then capacity in the older factories is freed to produce circuits for RFID tags.

Two important issues arise when we consider this drive to miniaturisation. First, while the digital component can be made smaller as technology advances, at least half current chips consist of analog components. There is no foreseeable comparable improvement to this part of the RFID tag and the tag will always need to support the relatively high current and voltage requirements of big transistors that cannot be expected to change much in size.

So is there an interest in taking a better process node for the digital component? The answer is not obvious since new processes have a major drawback; the static power consumption through thinner gates is not necessarily a negligible source of power dissipation, even in running modes. While reducing the supply voltage would lead to a reduction of dynamic power consumption, it also results in a decrease of performance or speed. To compensate for this, the threshold voltage needs to be reduced too. However, lowering the threshold voltage exponentially increases the static power consumption. At a certain point, the increase in the static power consumption can become larger than any gain in dynamic power consumption and the total power consumption could, conceivably, become larger. To illustrate, the total energy consumed by a NAND2 gate has been calculated using data on the most recent node processes and available at LETI . The results are illustrated in Table 7 where we see that the static power consumption really increases with the miniaturization of the transistor. Indeed, at the slower clocking frequency the contribution from static power consumption outweighs the advantage of a more advanced process. However this is not the case at high clocking frequencies, and so what the implications are in practice is still not clear. However it is possible that moving to smaller fabrication technologies might not automatically give us the substantial performance gains we might expect.

Indeed, with smaller digital components other limitations begin to appear. For instance, just handling and manufacturing tags that are built using advanced fabrication technologies could be difficult; using flip-chip technology the size of the two pads that are required to interface with the digital component are relatively significant. Indeed, there are some drastic problems in cutting, manipulating

Process μm	NAND2 (μm^2)	E_{GE} (fWs)	Static power (pW)	Total Energy / cycle (fWs)	
				100 KHz	1 MHz
0.065	2.08	0.6	10	0.70	0.61
0.045	1.04	0.3	55	0.85	0.36

Table 7: Total energy consumption for a NAND2 cell from two advanced process nodes at two different clocking frequencies.

and integrating very small silicon surfaces and many of these do not appear to be currently solved.

7 Conclusion

This report provides a detailed overview of the state of RFID tag technology and the state of lightweight cryptography. Despite what has become an urgent research agenda, it is clear that there still remains a mismatch between the resources available on current RFID tags and the set of algorithms and protocols that would soon exhaust these resources.

However, for the future we feel confident in predicting that as more advanced technologies come on stream, and as a greater understanding of the security demands of different applications becomes clearer, then this mismatch is likely to disappear. Already there are many cryptographic primitives that achieve most of the goals we seek, and it would take very little in terms of technical advance for their deployment to become a practical reality.

References

- [1] http://www.aliantechnology.com/fsa_manufacturing.php.
- [2] M. Aigner and M. Feldhofer. Secure Symmetric Authentication for RFID Tags. In *TCMC 05*, Graz, Austria, March 2005.
- [3] K. Aoki, T. Ichikawa, M. Kanda, M. Matsui, S. Moriai, J. Nakajima, and T. Tokita. Camellia: A 128-Bit Block Cipher Suitable for Multiple Platforms - Design and Analysis. In D. Stinson and S. Tavares, editors, *SAC 00*, volume 2012 of *LNCS*, pages 39-56. Springer-Verlag, 2000.
- [4] D. Arditti, C. Berbain, O. Billet, and H. Gilbert. Compact FPGA implementations of QUAD. In F. Bao and S. Miller, editors, *ASIACCS 2007*. ACM Press, 2007.
- [5] Auto-ID Center. 860MHz 960MHz Class I Radio Frequency Identification Tag Radio Frequency & Logical Communication Interface Specification, v1.0.0. Research Report MIT-AUTOID-TR-007, 2002.
- [6] G. Avoine. Privacy Issues in RFID Banknote Protection Schemes. In J.-J. Quisquater, P. Paradinas, Y. Deswarte, and A. Abou El Kadam, editors, *CARDIS 2004*, pages 33-48. Kluwer, 2004.
- [7] G. Avoine. Adversarial model for radio frequency identification. Cryptology ePrint Archive, Report 2005/049, <http://eprint.iacr.org/>, 2005.
- [8] G. Avoine, E. Dysli, and P. Oechslin. Reducing Time Complexity in RFID Systems. In B. Preneel and S. Tavares, editors, *SAC 05*, volume 3897 of *LNCS*, pages 291-306. Springer, 2005.
- [9] G. Avoine and P. Oechslin. A Scalable and Provably Secure Hash Based RFID Protocol. In *PerSec 05*. IEEE Computer Society Press, 2005.

- [10] G. Avoine and P. Oechslin. RFID traceability: A multilayer problem. In A. Patrick and M. Yung, editors, *FC 05*, volume 3570 of *LNCS*, pages 125-140. Springer, 2005.
- [11] L. Batina, J. Guajardo, T. Kerins, N. Mentens, P. Tuyls, and I. Verbauwhede. Public-Key Cryptography for RFID-Tags. In *PerSec 07*, pages 217-222. IEEE Computer Society Press, 2007.
- [12] C. Berbain, H. Gilbert, and J. Patarin. QUAD: A Practical Stream Cipher with Provable Security. In S. Vaudenay, editor, *EUROCRYPT 06*, volume 4004 of *LNCS*, pages 109-128. Springer, 2006.
- [13] E. Berlekamp, R. McEliece, and H. Van Tilborg. On the Inherent Intractability of Certain Coding Problems. *IEEE Transactions on Information Theory*, volume 24, issue 3, pages 384-386. IEEE Computer Society Press, 1978.
- [14] O. Billet, J. Etrog and H. Gilbert. An Efficient Forward-Private RFID Protocol. *ACM CCS 09*, to appear.
- [15] A. Bogdanov, L.R. Knudsen, G. Leander, C. Paar, A. Poschmann, M.J.B. Robshaw, Y. Seurin, and C. Vikkelsoe. PRESENT: An Ultra-Lightweight Block Cipher. In P. Paillier and I. Verbauwhede, editors, *CHES 07*, volume 4727 of *LNCS*, pages 450-466. Springer, 2007.
- [16] A. Bogdanov, G. Leander, C. Paar, A. Poschmann, M.J.B. Robshaw, and Y. Seurin. Hash Functions and RFID Tags: Mind the Gap. In E. Oswald and P. Rohatgi, editors, *CHES 08*, volume 5154 of *LNCS*, pages 283-299, Springer, 2008.
- [17] J. Bringer and H. Chabanne. Trusted-HB: a low-cost version of HB^+ secure against man-in-the-middle attacks. *CORR*, page abs/0802.0603, 2008.
- [18] J. Bringer, H. Chabanne, E. Dottax, and S. Securite. HB^{++} : A lightweight authentication protocol secure against some attacks. In *SecPerU 06*, pages 28-33. IEEE Computer Society Press, 2006.
- [19] J. Bringer, H. Chabanne, and T. Icart. Cryptanalysis of EC-RAC, a RFID identification protocol. In M. Franklin, L. Hui, and D. Wong, editors, *CANS 08*, volume 5339 of *LNCS*, pages 149-161. Springer, 2008.
- [20] M. Burmester, T. v. Le, and B. d. Medeiros. Provably Secure Ubiquitous Systems: Universally Composable RFID Authentication Protocols. In *SecureComm 06*, pages 1-9. IEEE Computer Society Press, 2006.
- [21] S. Canard and I. Coisel. Data Synchronization in Privacy-Preserving RFID Authentication Schemes. In *Conference on RFID Security*, 2008.
- [22] CASPIAN. <http://www.spsychips.com>.

- [23] J.-C. Chang and H.-L. Wu. A Hybrid RFID Protocol against Tracking Attacks. Cryptology ePrint Archive, Report 2009/138, 2009.
- [24] E. Y. Choi, S. M. Lee, and D. H. Lee. Efficient RFID Authentication Protocol for Ubiquitous Computing Environment. In T. Enokido, L. Yan, B. Xiao, D. Kim, Y. Dai, and L. Yang, editors, *SecUbiq 05*, volume 3823 of *LNCS*, pages 945-954. Springer-Verlag, 2005.
- [25] K. Chung. Low Cost and Reliable RFID Tags for All Frequencies, www.avantetech.com.
- [26] J. Cichon, M. Klonowski, and M. Kutylowski. Privacy Protection in Dynamic Systems Based on RFID Tags. In *PerSec 07*, pages 235-240. IEEE Computer Society Press, 2007.
- [27] I. Damgård and M. Østergaard. RFID Security: Tradeoffs between Security and Efficiency. Cryptology ePrint Archive, Report 2006/234, 2006.
- [28] T. Dimitriou. A Lightweight RFID Protocol to protect against Traceability and Cloning attacks. In *SecureComm 05*. IEEE Computer Society Press, 2005.
- [29] T. Dimitriou. A Secure and Efficient RFID Protocol that could make Big Brother (partially) Obsolete. In *PerCom 06*. IEEE Computer Society Press, 2006.
- [30] D. Duc and K. Kim. Securing HB+ against GRS man-in-the-middle attack. In *Institute of Electronics, Information and Communication Engineers, Symposium on Cryptography and Information Security*, pages 23-26, 2007.
- [31] ECRYPT Network of Excellence. The Stream Cipher Project: eSTREAM. Available via www.ecrypt.eu.org/stream.
- [32] Electronic Product Code Global Inc. <http://www.epcglobalinc.com>.
- [33] M. Feldhofer. An Authentication Protocol in a Security Layer for RFID Smart Tags. In *MELECON 04*, volume 2, pages 759-762. IEEE Computer Society Press, 2004.
- [34] M. Feldhofer. Comparison of Low-Power Implementations of Trivium and Grain. State of the Art of Stream Ciphers 2007 (SASC 2007), Workshop Record, February 2007. Available via www.ecrypt.eu.org/stream/.
- [35] M. Feldhofer, S. Dominikus, and J. Wolkerstorfer. Strong Authentication for RFID Systems using the AES Algorithm. In M. Joye and J.-J. Quisquater, editors, *CHES 04*, volume 3156 of *LNCS*, pages 357-370. Springer-Verlag, 2004.
- [36] M. Feldhofer and C. Rechberger. A Case Against Currently Used Hash Functions in RFID Protocols. In *IS 06*, volume 4277 of *LNCS*, pages 372-381, Springer-Verlag, 2006.

- [37] M. Feldhofer and J. Wolkerstorfer. Hardware Implementation of Symmetric Algorithms for RFID Security. In P. Kitsos and Y. Zhang, editors, *RFID Security: Techniques, Protocols and System-on-Chip Design*, pages 373-415. Springer, 2008.
- [38] K. Finkenzeller. *RFID handbook: radio-frequency identification fundamentals and applications*. Wiley, 1999.
- [39] H. Gilbert, M. Robshaw, and Y. Seurin. Good variants of HB+ are hard to find. In G. Tsudik, editor, *Financial Cryptology 08*, volume 5143 of *LNCS*, pages 156-170. Springer, 2008.
- [40] H. Gilbert, M. Robshaw, and Y. Seurin. $HB^\#$: Increasing the Security and Efficiency of HB. In N. Smart, editor, *Eurocrypt 08*, volume 4965 of *LNCS*, pages 361-378. Springer, 2008.
- [41] H. Gilbert, M. Robshaw, and Y. Seurin. $HB^\#$: Increasing the Security and Efficiency of HB, full version, Cryptology ePrint Archive, Report 2008/028, 2008.
- [42] H. Gilbert, M. Robshaw, and H. Sibert. An active attack against HB+-a provably secure lightweight authentication protocol. IEE Electronic Letters, 41: 1169-1170. Technical report, See also Cryptology ePrint Archive, Report 2005/237, <http://eprint.iacr.org>, 2005.
- [43] M. Girault. Self-certified public keys. In D. W. Davies, editor, *Eurocrypt 91*, volume 547 of *LNCS*, pages 490-497. Springer-Verlag, 1991.
- [44] M. Girault. Low-Size Coupons for Low-Cost IC Cards. In J. Domingo-Ferrer, D. Chan, and A. Watson, editors, *Proceedings of the fourth working conference on smart card research and advanced applications on Smart card research and advanced applications*, pages 39-50. Kluwer Academic Publishers, 2001.
- [45] M. Girault, L. Juniot, and M. Robshaw. The Feasibility of On-the-Tag Public Key Cryptography. In *Conference on RFID Security 2007 - Workshop Record*, available via <http://rfidsec07.etsit.uma.es/slides/papers/paper-32.pdf>, 2007.
- [46] M. Girault and D. Lefranc. Public Key Authentication with One (Online) Single Addition. In M. Joye and J.-J. Quisquater, editor, *CHES 04*, volume 3156 of *LNCS*, pages 967-984. Springer-Verlag, 2004.
- [47] M. Girault, G. Poupard, and J. Stern. On the Fly Authentication and Signature Schemes Based on Groups of Unknown Order. *Journal of Cryptology*, volume 19, pages 463-487. Springer, 2006.

- [48] M. Girault and J. Stern. On the Length of Cryptographic Hash-Values Used in Identification Schemes. In Y. Desmedt, editor, *CRYPTO 94*, volume 893 of *LNCS*, pages 202-215. Springer-Verlag, 1994.
- [49] R. Glidden, C. Bockorick, S. Cooper, C. Diorio, D. Dressler, V. Gutnik, C. Hagen, D. Hara, T. Hass, T. Humes, et al. Design of ultra-low-cost UHF RFID tags for supply chain applications. *IEEE Communications Magazine*, volume 42, issue 8, pages 140-151. IEEE Computer Society Press, 2004.
- [50] O. Goldreich and L. Levin. A hard-core predicate for all one way-functions. In *STOC*. ACM Press, 1989.
- [51] P. Golle, M. Jakobsson, A. Juels, and P. Syverson. Universal re-encryption for mixnets. In T. Okamoto, editor, *CT-RSA 04*, volume 2964 of *LNCS*, pages 163-178. Springer-Verlag, 2004.
- [52] T. Good and M. Benaissa. ASIC hardware performance. In M. Robshaw and O. Billet, editors, *New Stream Cipher Designs*, volume 4986 of *LNCS*, pages 267-293. Springer, 2008.
- [53] T. Good and M. Benaissa. Hardware Results for Selected Stream Cipher Candidates. State of the Art of Stream Ciphers 2007 (SASC 2007), Workshop Record, February 2007. Available via www.ecrypt.eu.org/stream.
- [54] L. Guo, A. Popov, H. Li, Y. Wang, V. Bliznetsov, G. Lo, N. Balasubramanian, and D. Kwong. A small OCA on a 1×0.5 -mm 2.45 -GHz RFID Tag-design and integration based on a CMOS-compatible manufacturing technology. *IEEE Electron Device Letters*, volume 27, issue 2, pages 96-98. IEEE Computer Society Press, 2006.
- [55] G. Hammouri and B. Sunar. PUF-HB: A tamper-resilient HB based authentication protocol. In S. M. Bellovin, R. Gennaro, A. Keromytis and M. Yung, editors, *ACNS 08*, volume 5037 of *LNCS*, pages 346-365. Springer, 2008.
- [56] M. Hellman. A Cryptanalytic Time-Memory Trade-Off. *IEEE Transactions on Information Theory*, volume 26, issue 4, pages 401-406. IEEE Computer Society Press, 1980.
- [57] M. Hell. On the Design and Analysis of Stream Ciphers. Ph.D. Thesis, Lund University, 2007.
- [58] M. Hell, T. Johansson, A. Maximov, and W. Meier. A Stream Cipher Proposal: Grain-128. In *IEEE International Symposium on Information Theory-ISIT 2006*, 2006.
- [59] M. Hell, T. Johansson, and W. Meier. Grain - A Stream Cipher for Constrained Environments. In M. Robshaw and O. Billet, editors, *New Stream Cipher Designs: The eSTREAM Finalists*, pages 179-190. Springer, 2008.

- [60] J. E. Hennig, P. B. Ladkin, and B. Sieker. Privacy Enhancing Technology Concepts for RFID Technology Scrutinised. RVS-RR-04-02, Univ. of Bielefeld, 2004.
- [61] D. Henrici and P. Müller. Hash-Based Enhancement of Location Privacy for Radio-Frequency Identification Devices Using Varying Identifiers. In R. Sandhu and R. Thomas, editors, *PerSec 04*, pages 149-153. IEEE Computer Society, 2004.
- [62] Y. Hong, C. Chan, J. Guo, Y. Ng, W. Shi, L. Leung, K. Leung, C. Choy, and K. Pun. Design of passive UHF RFID tag in 130nm CMOS technology. In *APCCAS 08*, pages 1371-1374. IEEE Computer Society Press, 2008.
- [63] D. Hong, J. Sung, S. Hong, J. Lim, S. Lee, B.-S; 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. In L. Goubin and M. Matsui, editors, *CHES 06*, volume 4249 of *LNCS*, pages 46-59. Springer-Verlag, 2006.
- [64] N. Hopper and M. Blum. Secure human identification protocols. In C. Boyd, editor, *ASIACRYPT 01*, volume 2248 of *LNCS*, pages 52-66. Springer, 2002.
- [65] W. Jeon, J. Melngailis, and R. Newcomb. CMOS passive RFID transponder with read-only memory for low cost fabrication. In *IEEE International SOC Conference, 2005*, pages 181-184. IEEE Computer Society Press, 2005.
- [66] A. Juels. Minimalist Cryptography for Low-Cost RFID Tags. In C. Blundo and S. Cimato, editors, *SCN 04*, volume 3352 of *LNCS*, pages 149-164. Springer, 2004.
- [67] A. Juels and R. Pappu. Squealing Euros: Privacy Protection in RFID-Enabled Banknotes. In R. N. Wright, editor, *FC 03*, volume 2742 of *LNCS*, pages 103-121. Springer, 2003.
- [68] A. Juels, R. Rivest, and M. Szydlo. The Blocker Tag: Selective Blocking of RFID Tags for Consumer Privacy. In V. Atluri, editor, *ACM CCS*. ACM Press, 2003.
- [69] A. Juels and S. Weis. Defining strong privacy for RFID. Cryptology ePrint Archive, Report 2006/137.
- [70] A. Juels and S.A. Weis. Authenticating Pervasive Devices With Human Protocols. In V. Shoup, editor, *Advances in Cryptology - Crypto 05*, *LNCS*, volume 3126, pages 293-198, Springer-Verlag, 2005.
- [71] S. Karthikeyan and M. Nesterenko. RFID Security without Extensive Cryptography. In *SASN 05*, pages 63-67. ACM Press, 2005.

- [72] J. Katz and J. Shin. Parallel and concurrent security of the HB and HB+ protocols. In S. Vaudenay, editor, *EUROCRYPT 06*, volume 4004 of *LNCS*, pages 73-87, Springer 2006.
- [73] G. Leander, C Paar, A. Poschmann, and K Schramm. A Family of Lightweight Block Ciphers Based on DES Suited for RFID Applications. In A. Biryukov, editor, *FSE 07*, volume 4593 of *LNCS*, pages 196-210, Springer, 2007.
- [74] J. Lee and Y. Yeom. Efficient RFID Authentication Protocols Based on Pseudorandom Sequence Generators. Cryptology ePrint Archive, Report 2008/343, August 2008.
- [75] S. Lee, T. Asano, and K. Kim. RFID Mutual Authentication Scheme based on Synchronized Secret Information. In *Symposium on Cryptography and Information Security*, Hiroshima, Japan, January 2006.
- [76] S.-M. Lee, Y. J. Hwang, D. H. Lee, and J. I. L. Lim. Efficient Authentication for Low-Cost RFID Systems. In O. Gervasi, M. Gavrilova, V. Kumar, A. Laganaà, H. P. Lee, Y. Mun, D. Taniar, and C. J. K. Tan, editors, *ICCSA 05*, volume 3480 of *LNCS*, pages 619-627. Springer-Verlag, 2005.
- [77] Y. K. Lee, L. Batina, and I. Verbauwhede. EC-RAC (ECDLP Based Randomized Access Control): Provably Secure RFID Authentication Protocol. *IEEE International Conference on RFID*, pages 97-104, April 2008.
- [78] Y. K. Lee and I. Verbauwhede. Secure and Low-Cost RFID Authentication Protocols. In *AWiN 05*. IEEE Computer Society, 2005.
- [79] S. Lemieux and A. Tang. Clone Resistant Mutual Authentication for Low-Cost RFID Technology. IACR ePrint, Report 2007/170, 2007.
- [80] É. Leveil and P. Fouque. An improved LPN algorithm. In R. De Prisco and M. Yung, editors, *SCN 06*, volume 4116 of *LNCS*, pages 348-359. Springer, 2006.
- [81] C. H. Lim and T. Kwon. Strong and Robust RFID Authentication Enabling Perfect Ownership Transfer. In P. Ning, S. Qing and N. Li, editors, *ICICS 06*, volume 4307 of *LNCS*, pages 1-20. Springer-Verlag, 2006.
- [82] C. Lim and T. Korkishko. mCrypton - A Lightweight Block Cipher for Security of Low-cost RFID Tags and Sensors. In J. Song, T. Kwon, and M. Yung, editors, *WISA 05*, volume 3786 of *LNCS*, pages 243-258. Springer-Verlag, 2005.
- [83] T. Lohmann, M. Schneider, and C. Ruland. Analysis of power constraints for cryptographic algorithms in mid-cost RFID tags. In J. Domingo-Ferrer, J. Posegga and D. Schreckling, editors, *CADIS 06*, volume 3928 of *LNCS*, pages 278-288. Springer, 2006.

- [84] Y. Mansour, N. Nisan, and P. Tiwari. The computational complexity of universal hashing. In *STOC 90*, pages 235-243. ACM Press, 1990.
- [85] M. McLoone and M. J. B. Robshaw. Public Key Cryptography and RFID. In M. Abe, editor, *CT-RSA 07*, volume 4377 of *LNCS*, pages 372-384, Springer, 2007.
- [86] M. McLoone and M. J. B. Robshaw. New Architectures for Low-Cost Public Key Cryptography on RFID Tags. In *SecureComm 05*, pages 1827-1830. IEEE Computer Society Press, 2007.
- [87] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone. *Handbook of Applied Cryptography*. CRC Press, Boca Raton, Florida, USA, first edition, 1996.
- [88] D. Molnar, A. Soppera, and D. Wagner. A Scalable, Delegatable Pseudonym Protocol Enabling Ownership Transfer of RFID Tags. In B. Preneel and S. Tavares, editors, *SAC 05*, volume 3897 of *LNCS*, pages 276-290. Springer-Verlag, 2005.
- [89] D. Molnar and D. Wagner. Privacy and security in library RFID: Issues, practices, and architectures. In B. Pfitzmann and P. Liu, editors, *ACM CCS*, pages 210-219. ACM Press, 2004.
- [90] G.E. Moore. Cramming more components onto integrated circuits. *Electronics*, 38(8), April 1965.
- [91] J. Munilla and A. Peinado. HB-MP: A further step in the HB-family of lightweight authentication protocols. In J. Domingo-Ferrer, J. Posegga, F. Seb e, V. Torra, G. Karetsos, A. Rouskas, B. Jabbari and B. Walke, editors, *Computer Networks*, volume 51 of *Computer Networks*, issue 9, pages 2262-2267, 2007.
- [92] National Institute of Standards and Technology. FIPS 197: Advanced Encryption Standard, November 2001. Available via csrc.nist.gov.
- [93] National Institute of Standards and Technology. Special Publication 800-67: Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher, May 2008. Available via csrc.nist.gov.
- [94] National Institute of Standards and Technology. NIST Special Publication 800-38B: Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication. May 2005. Available via csrc.nist.gov.
- [95] National Institute of Standards and Technology. FIPS 198-1: The Keyed-Hash Message Authentication Code (HMAC), July 2008. Available via csrc.nist.gov.
- [96] Y. Nohara, S. Inoue, K. Baba, and H. Yasuura. Quantitative Evaluation of Unlinkable ID Matching Schemes. In *WPES 06*, pages 55-60. ACM Press 2006.

- [97] M. O'Neill (nee McLoone). Low-Cost SHA-1 Hash Function Architecture for RFID Tags. *RFIDSec 08*, pages 41-51, 2008.
- [98] M. Ohkubo, K. Suzuki, and S. Kinoshita. Cryptographic Approach to "Privacy-Friendly" Tags. In *RFID Privacy Workshop*, 2003.
- [99] M. Ohkubo, K. Suzuki, and S. Kinoshita. Efficient hash-chain based RFID privacy protection scheme. In *Ubiquitous Computing - Privacy Workshop*, 2004.
- [100] T. Okamoto. Provably secure and practical identification schemes and corresponding signature schemes. In E. F. Brickell, editor, *CRYPTO 92*, volume 740 of *LNCS*, pages 31-53. Springer, 1993.
- [101] Y. Oren and M. Feldhofer. WIPR - a Public Key Implementation on Two Grains of Sand. In *Conference on RFID Security*, Budapest, Hungary, July 2008. <http://iss.oy.ne.ro/WIPR>.
- [102] Y. Oren and M. Feldhofer. A Low-Resource Public-Key Identification Scheme for RFID Tags and Sensor Nodes. In *WiSec 09*. ACM Press, 2009.
- [103] K. Ouafi, R. Overbeck, and S. Vaudenay. On the Security of HB# against a Man-in-the-Middle Attack. In J. Pieprzyk, editor, *ASIACRYPT 08*, volume 5350 of *LNCS*, pages 108-124. Springer, 2008.
- [104] K. Ouafi and S. Vaudenay. Smashing SQUASH-0. In A. Joux, editor, *EUROCRYPT 09*, volume 5479 of *LNCS*, pages 300-312. Springer, 2009.
- [105] C. Paar, A. Poschmann, and M.J.B. Robshaw. New Designs in Lightweight Symmetric Encryption. In P. Kitsos and Y. Zhang, editors, *RFID Security: Techniques, Protocols and System-on-Chip Design*, pages 349-372. Springer, 2008.
- [106] D. Paret. *RFID en Ultra et Super Hautes Fréquences UHF-SHF, Théorie et mise en oeuvre*. DUNOD, 2005.
- [107] K. Rhee, J. Kwak, S. Kim, and D. Won. Challenge-Response based RFID Authentication Protocol for Distributed Database Environment. In D. Hutter and M. Ullmann, editors, *SPC 05*, volume 3450 of *LNCS*, pages 70-84. Springer-Verlag, 2005.
- [108] C. Rolfes, A. Poschmann, G. Leander, and C. Paar. Ultra-Lightweight Implementations for Smart Devices - Security for 1000 Gate Equivalents. In *CARDIS 2008*, to appear. Springer-Verlag.
- [109] J. Saito, J.-C. Ryou, and K. Sakurai. Enhancing Privacy of Universal Re-encryption Scheme for RFID Tags. In L. Jang, M. Guo, G. Gao, and N. Jha, editors, *EUC 04*, volume 3207 of *LNCS*, pages 879-890. Springer-Verlag, 2004.

- [110] S. Sarma, S. Weis, and D. Engels. RFID Systems and Security and Privacy Implications. In B. Kaliski, C. Koç, and C. Paar, editors, *CHES 02*, volume 2523 of *LNCS*, pages 454-469. Springer-Verlag, 2002.
- [111] C. Schnorr. Efficient identification and signatures for smart cards. In J. J. Quisquater and J. Vandewalle, editors, *CRYPTO 89*, volume 434 of *LNCS*, pages 239-252. Springer, 1990.
- [112] A. Shamir. SQUASH - a New MAC With Provable Security Properties for Highly Constrained Devices Such As RFID Tags. In K. Nyberg, editor, *FSE 2008*, volume 5086 of *LNCS*, pages 144-157. Springer, 2008.
- [113] Supply chain Digest. January 2009.
- [114] G. Tsudik. YA-TRAP: Yet Another Trivial RFID Authentication Protocol. In *PerCom 06*. IEEE Computer Society Press 2006.
- [115] G. Tsudik. A Family of Dunces: Trivial RFID Identification and Authentication Protocols. Cryptology ePrint Archive, Report 2006/015, 2007.
- [116] P. Tuyls and L. Batina. RFID-Tags for Anti-Counterfeiting. In D. Pointcheval, editor, *CT-RSA 06*, volume 3860 of *LNCS*, pages 115-131. Springer-Verlag, 2006.
- [117] M. Usami. An Ultra-small RFID Chip: μ -chip. In *Proceedings of 2004 IEEE Asia-Pacific Conference on Advanced System Integrated Circuits 2004*, pages 2-5, 2004.
- [118] S. Vaudenay. On Privacy Models for RFID. In K. Kurosawa, editor, *ASIACRYPT 07*, volume 4833 of *LNCS*, pages 68-87. Springer, 2007.
- [119] S. Weis, S. Sarma, R. Rivest, and D. Engels. Security and Privacy Aspects of Low-Cost Radio Frequency Identification Systems. In D. Hutter, G. Müller, W. Stephan, and M. Ullmann, editors, *SPC 2003*, volume 2802 of *LNCS*, pages 454-469. Springer, 2003.
- [120] J. Wolkerstorfer, S. Dominikus, and M. Feldhofer. Strong authentication for RFID systems using the AES algorithm. In M. Joye and J.-J. Quisquater, editors, *CHES 04*, volume 3156 of *LNCS*, pages 357-370. Springer, 2004.
- [121] J. Wu and D. Stinson. How to Improve Security and Reduce Hardware Demands of the WIPR RFID Protocol. In *IEEE International Conference on RFID - RFID 2009*, Orlando, Florida, USA, April 2009.
- [122] S.-S. Yeo and S.-K. Kim. Scalable and Flexible Privacy Protection Scheme for RFID Systems. In R. Molva, G. Tsudik, and D. Westhoff, editors, *ESAS 05*, volume 3813 of *LNCS*, pages 153-163. Springer-Verlag, 2005.

- [123] W. Yeoh, Y. Choi, K. Tham, S. Diao, and Y. Li. A CMOS 2.45-GHz radio frequency identification tag IC with read/write memory. In *2005 IEEE Radio Frequency integrated Circuits (RFIC) Symposium, 2005. Digest of Papers*, pages 365-368, 2005.
- [124] H. Yoshida, D. Watanabe, K. Okeya, J. Kitahara, J. Wu, O. Kucuk, and B. Preneel. MAME: A Compression Function With Reduced Hardware Requirements. In P. Paillier and I. Verbauwhede, editors, *CHES 07*, volume 4727 of *LNCS*, pages 148-165. Springer, 2007.
- [125] Y. Yu, Y. Yang, Y. Fan, and H. Min. Security Scheme for RFID Tag. Auto-ID Labs white paper WP-HARDWARE-022. Available from <http://www.autoidlabs.org/>.
- [126] J. Zhai, C. Mok-Park, and G.-N. Wang. Hash-Based RFID Security Protocol Using Randomly Key-Changed Identification Procedure. In M. L. Gavrilova, O. Gervasi, V. Kumar, C. J. K. Tan, D. Taniar, A. Laganà, Y. Mun, and H. Choo, editors, *ICCSA 06*, volume 3983 of *LNCS*, pages 296-305. Springer-Verlag, 2006.