Lightweight Cryptography and RFID: Tackling the Hidden Overheads

Axel Poschmann[1], Matt Robshaw[2], Frank Vater[3], and Christof Paar[4]

[1] Division of Mathematical Sciences, Nanyang Technological University, Singapore
aposchmann@ntu.edu.sg
[2] Orange Labs, 38-40 rue du Général Leclerc, Issy les Moulineaux, France
matt.robshaw@orange-ftgroup.com
[3] Innovations for High Performance Microelectronics, Frankfurt/Oder, Germany
vater@ihp-microelectronics.com
[4] Horst Görtz Institute for IT Security, Ruhr-University Bochum, Germany
christof.paar@rub.de

Abstract. The field of lightweight cryptography has developed significantly over recent years and many impressive implementation results have been published. However these results are often concerned with a core computation and when it comes to a real implementation there can be significant hidden overheads. In this paper we consider the case of CRYPTOGPS and we outline a full implementation that has been fabricated in ASIC. Interestingly, the implementation requirements still remain within the typically-cited limits for on-the-tag cryptography.

1 Introduction

Radio-frequency identification (RFID) tags are becoming a part of our everyday life and a wide range of applications from the supply chain to the intelligent home are often described in the literature. Yet, at the same time, security and privacy issues remain a major issue, not least in the battle against counterfeit goods, pharmaceutical products, and even engine components in the automotive and aeronautic industries [16].

It has long been recognised that cryptographic techniques might be used to help alleviate these problems. However they have all too often been considered as too expensive to implement, or too unsuited to the enviroment of use. Over recent years this view has begun to change and there have been substantial advances in cryptographic design, for instance in new block ciphers such as PRESENT [3]. And as well as the advances we might have expected in symmetric cryptography—which is typically viewed as the lightweight choice—there has been a growing understanding of which asymmetric techniques are available and how they might best be implemented. Indeed, given the essential nature of an RFID-based deployment with many (potentially unknown) players being involved—*i.e.* we have an *open* rather than a *closed* system—lightweight public-key cryptography should be viewed as a particularly attractive technology.

D. Lee and S. Hong (Eds.): ICISC 2009, LNCS 5984, pp. 129–145, 2010.
© Springer-Verlag Berlin Heidelberg 2010

Some of the more recent implementation results in the literature have been very impressive. The oft-cited opinion is that there are around 2 000-3 000 gate equivalents (GE) available for on-tag security features,[1] and despite this representing a formidable challenge several algorithms claim to achieve this.

In this paper we highlight a problem with many of these estimates and we observe that figures are often given for the cryptographic core of a computation. For instance, estimates for the feasibility of elliptic curve cryptography might consider just the elliptic curve operation while implementation results for cryptoGPS [21,22] are focused on the protocol computations. This means that when it comes to a real implementation there can be significant hidden overheads.

The main purpose of this paper is to highlight this issue, but also to re-examine the case of one particular proposal in particular, that of cryptoGPS. To do this we will describe a full implementation of cryptoGPS which includes all the additional functionality that would be required in a real deployment. Further, noting that implementation results for lightweight cryptography are often derived from an FPGA implementation or ASIC synthesis tools, we have gone one step further and we report on the results of the full ASIC fabrication of a fully-supported version of cryptoGPS.

1.1 Related Work

Over recent years a lot of work on public key cryptography for RFID tags has centered around elliptic curves. A comparison between different ECC implementations is not always easy because the choice of the underlying curve determines both efficiency and security of the algorithm. However no implementation has been published so far that comes under 5 000 GE which would, even then, be too great for passive RFID-tags. Instead several elliptic curve implementations with a significantly lower security level than 80-bit exist, but their size lies in the range of 10 000 GE or above [2,6,8].

Gaubatz *et al.* [9] have investigated the hardware efficiency of the NTRUencrypt algorithm [18,26] with the following parameter set $(N, p, q) = (167, 3, 128)$ that offers a security level of around 57 bits. Though their implementation requires only 2 850 GE, it takes 29 225 clock cycles, which translates to 292 ms for the response to be computed at the typical clocking frequency of 100 KHz. Further, it is noteworthy that more than 80% of the area is occupied with storage elements and that already a bit serial datapath is used. This implies that the opportunities for future improvement are very limited. Oren *et al.* propose a public key identification scheme called wipr [27]. Their ASIC implementation requires 5 705 GE and 66 048 clock cycles, though a proposed optimisation [32] suggests a reduced area requirement of around 4 700 GE.

In this paper, however, we will concentrate on the cryptoGPS scheme. The name GPS is derived from the inventors Girault, Poupard, and Stern, but the term cryptoGPS is increasingly used to avoid confusion with the geographical

[1] The gate equivalent (GE) is a unit of area and is equivalent to the physical space occupied by a logical NAND gate for the given manufacturing process.