Journal of Network and Computer Applications 58 (2015) 73–93

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

A survey on lightweight block ciphers for low-resource devices: Comparative study and open issues

Bassam J. Mohd [a], Thaier Hayajneh [b], Athanasios V. Vasilakos [c]

[a] Computer Engineering Department, Hashemite University, Zarqa, Jordan
[b] School of Engineering and Computing Sciences, New York Institute of Technology, New York, USA
[c] Deptartment of Computer Science, Electrical and Space Engineering, Lulea University of Technology, Lulea 97187, Sweden

ARTICLE INFO

Article history:
Received 8 July 2014
Received in revised form
3 August 2015
Accepted 7 September 2015
Available online 16 September 2015

Keywords:
Security
Cryptography
Cipher
Encryption
Low-resource devices
FPGA

ABSTRACT

This paper investigates the lightweight block ciphers' implementations, which have received a fair amount of research for their essential security role in low-resource devices. Our objective is to present a comprehensive review of state-of-the-art research progress in lightweight block ciphers' implementation and highlight future research directions. At first, we present taxonomy of the cipher design space and accurately define the scope of lightweight ciphers for low-resource devices. Moreover, this paper discusses the performance metrics that are commonly reported in the literature when comparing cipher implementations. The sources of inaccuracies and deviations are carefully examined. In order to mitigate the confusion in the composite metrics, we developed a general metric which includes the basic metrics. Our analysis designated the energy/bit metric as the most appropriate metric for energy-constrained low-resource designs. Afterwards, the software and hardware implementations of the block cipher algorithms are surveyed, investigated, and compared. The paper selects the top performing ciphers in various metrics and suggests the Present cipher as a good reference for hardware implementations. What transpires from this survey is that unresolved research questions and issues are yet to be addressed by future research projects.

1. Introduction

Lightweight ciphers are designed for rapidly growing applications that extensively employ smart, low-resource devices. The applications include wireless sensor network (WSN) (Rolfes et al., 2008), wireless body area network (WBAN) (Zhang et al., 2011; Chen et al., 2011), radio-frequency identification(RFID) (Rolfes et al., 2008), internet of things (IoT) (Atzori et al., 2010), and smartcards (Rolfes et al., 2008) among others. Typically, applications exchange sensitive or private data, and therefore assuring an adequate level of data security is a fundamental requirement. For example, in WBAN stringent security methods must be implemented to protect medical data privacy (Latré et al., 2011). Hence, data encryption is an essential design requirement to ensure the security of the data. However, due to their computational and energy limitations (powered with small batteries), lightweight ciphers must not burden the low-resource devices and impact their lifetime.

A low-resource device refers to a class of implementations characterized by low computing power (e.g. 8-bit microcontroller), limited battery supply, small (gate) area, and/or small memory requirements. Generally, ciphers targeted for low-resource devices are referred to as lightweight ciphers. Thus, the designers of lightweight ciphers must cope with the trade-offs between security, cost and performance. The fundamental problem of providing security in such low-end devices is the extremely constrained environment; the design must have a small footprint, low power and energy consumption, and satisfactory speed (Canniere et al., 2009).

Ciphers are classified as asymmetric (public-key) and symmetric. Asymmetric ciphers offer more security features; however, they are computationally more demanding and relatively more expensive (Law et al., 2006; Carman et al., 2000; Piedra et al., 2013; Hayajneh et al., 2014). For example, using an 8-bit controller platform, the software implementation of elliptic-curve cryptography (ECC) performs 100–1000 times slower than the Advanced Encryption Standard (AES) (Eisenbarth et al., 2007).

Symmetric ciphers are further classified as block ciphers and stream ciphers. Stream ciphers can be easily constructed by block ciphers (Menezes et al., 2010). Some protocols cannot be designed with stream ciphers (Canniere et al., 2009). Typically, a cipher algorithm consists of three sub-algorithms: encryption, decryption, and key expansion. The key expansion expands the cipher key to process all the cipher text bits. In most ciphers, key expansion is performed once for both encryption and decryption (Law et al., 2006); however, in the case of AES the key expansion is

http://dx.doi.org/10.1016/j.jnca.2015.09.001
1084-8045/

performed separately for encryption and decryption. The main block cipher parameters are key-size, block-size, and number of rounds.

Based on the algorithm structure, block ciphers are classified into Substitution Permutation Network (SPN), Feistel, stream, and Lai-Massey (Cazorla et al., 2013). In the context of lightweight cryptography (e.g. WSNs and WBANs), the essential cryptographic primitives are block ciphers, hash functions, and message authentication codes (MACs) (Law et al., 2006). The hash functions are relatively inexpensive and MACs can be designed using block ciphers (Law et al., 2006; Preneel, 1998; Crosby and Wallach, 2003). Therefore, most of the research on lightweight cryptography focused on block ciphers.

Another class of the encryption schemes is the chaos-based encryption, which is based on properties of chaotic system e.g. dependence on initial conditions and pseudo-random property (Kocarev, 2001). The chaos-based encryption algorithms are better fit for image compression as they offer confusion and diffusion (François et al., 2012). However, chaos-based encryption algorithms have several challenges including weak security and slow execution (Kocarev, 2001; He et al., 2010). Several research works were published to improve security and reduce complexity, e.g. (François et al., 2012; Fawaz et al., 2013). However, research work on chaos-based lightweight block ciphers targeted for low-resources devices is not significant.

Several surveys examining cipher implementation were published lately, albeit with different objectives. A group of surveys studied cipher designs for high-performance systems (e.g. (Bossuet et al., 2013)), while others focused on small embedded platforms (e.g., Law et al., 2006; Eisenbarth et al., 2012). Some articles addressed the software implementations (e.g., Cazorla et al., 2013; Malina et al., 2014; Weis and Lucks, 2000), others concentrated on hardware designs (e.g., Kerckhof et al., 2012; Batina et al., 2013)) with a few articles covering both (e.g., Eisenbarth et al., 2007). Further, many cipher-related studies focused on specific performance aspects, such as throughput issues (Järvinen et al., 2005) and security problems in FPGAs (Wollinger et al., 2004).

The primary goal of this article is to sketch out a roadmap of existing research in the area of lightweight cipher implementations. Furthermore, we seek to identify the gaps and challenges in the current research, which are potential research opportunities. Our approach is more comprehensive and addresses critical issues in greater depth. The issues include; presenting novel cipher taxonomy; investigating performance metrics' challenges; developing a general metric to mitigate the metric confusion; covering and analyzing a large set of ciphers; and determining the top performing ciphers.

The main contributions of this paper can be summarized as follows:

- Present a precise definition for the term "lightweight cipher for low-resource device" by analyzing cipher implementation space in the context of the system performance and targeted platform. A novel taxonomy of cipher implementations is discussed.
- Examine the performance metrics cited in the literature, and highlight the confusion and lack of a uniform platform to measure performance. We focus on various metric issues, sources of errors, and the significance of the energy/bit metric for lightweight block ciphers. To mitigate the metric confusion, we propose a general metric which combines the basic metrics.
- Analyze and compare the existing research on lightweight block cipher implementations. Some of the ciphers presented are not lightweight in the strictest sense, but are included for comparison. Both software and hardware implementations are investigated. While conventional ciphers have received extensive research work, lightweight ciphers have received less attention (Kerckhof et al., 2012). The set of ciphers that are selected for comparison varies from one research to another; however, AES is cited in most of the studies. The list of ciphers that are covered in this survey is shown in Table 1. The table also shows the key size, block size, structure type, and number of rounds. We made a great effort to include most of the block ciphers reported in the literature.
- Investigate the reported data from existing studies and extract conclusions. Direct comparisons of reported numbers may lead to misinterpretation, as noted by researchers in Bossuet et al. (2013). The reports are used to extract intuitive observations and draw a map of the best performing ciphers in various metric categories. This should drive the development of new algorithms by cross-pollinating the finest cipher designs.
- Finally, we discuss potential directions for open research issues based on what transpires in the discussions.

The rest of the paper is organized as follows: Section 2 discusses background information about lightweight ciphers and security in low-resource devices. Section 3 presents our taxonomy of the cipher implementation space. Section 4 examines the performance metrics typically cited in the software and hardware implementations. Section 5 provides a brief overview of the cipher software implementations. Section 6 explores the cipher hardware implementations. Open research issues are discussed in Section 7. Finally, Section 8 summarizes the paper and provides concluding remarks.

2. Background

The term “lightweight” is overused by researchers and thereby different definitions, albeit converging, abounds in the literature. In this section, the lightweight cipher definition is discussed. Security challenges for low-resource devices are examined as well.

2.1. Lightweight block cipher

The researchers in Fan et al. (2013) defined a lightweight cipher as a cryptographic algorithm that is tailored for low-resource devices and must address three challenges: minimal overhead (silicon area or memory footprint), low-power consumption, and adequate security level.

Others defined lightweight ciphers as those targeted for low-cost designs. For instance, (Eisenbarth et al., 2007) designated ciphers targeted for WSN and WBAN (i.e. with limited resources) as lightweight cryptography/ciphers. It is even more challenging to accurately describe the low-cost definition because of its strong dependency on the targeted platform (i.e. software or hardware) (Kerckhof et al., 2012; Eisenbarth et al., 2012).

Other researchers adopted a more quantitative approach to define lightweight ciphers. For instance, to designate a cipher as lightweight, researchers in Cazorla et al. (2013) suggested the following properties: block-size, key-size, operations, and key scheduling. "Lightweight" implies small block size (32, 48, or 64 bits) compared with a conventional cipher, which has a larger block size (64 or 128 bits). The key size also tends to be smaller in the case of lightweight ciphers. Additionally, lightweight ciphers may simplify the key schedule (Cazorla et al., 2013) and employ elementary operations with a larger number of rounds. To illustrate the quantitative definition, Table 2 compares AES, a conventional cipher, and Ktantan, a lightweight cipher.

The lightweight ciphers are targeted for low-resource devices, and therefore, their implementations (software or hardware) should optimize the resource utilization. One of the critical issues for low-resource devices is power and energy consumption. The main power supply often is a limited battery, which in some cases