Journal of Research in Engineering and Applied Sciences

APPLICABLE LIGHT-WEIGHT CRYPTOGRAPHY TO SECURE MEDICAL DATA IN IOT SYSTEMS

[1]Norah Alassaf, [2]Basem Alkazemi, [3]Adnan Gutub

[1]Master Graduate Student in Computer Science & Engineering, Umm Al-Qura University (UQU), Saudi Arabia;
[2]Associate Professor, Computer Science Department, Umm Al-Qura University (UQU), Saudi Arabia;
[3]Professor, Computer Engineering Department, Umm Al-Qura University (UQU), Saudi Arabia
Email: {1noura.alassaf@hotmail.com; 2bykazemi@uqu.edu.sa; 3aagutub@uqu.edu.sa}

Abstract

Different applications require different level of security where the scarce of resource plays effective role. In order to protect the private medical data in the internet of things (IoT) field, the search for the optimal encryption algorithm is a must. Electronic sensors are used to collect medical data from the patient's body acquiring its transmission to the healthcare system securely. It is essential to ensure trust and data secrecy from the starting point-sensors throughout the medical treatment to prevent any unauthorized access or unneeded interruption. Thus, data encryption from the beginning sensors is necessary but facing all limitations in computing complexity, power consumption and communication bandwidth, where the normal available crypto-algorithms are considered heavyweight is completely unpractical. This work study several realistic lightweight encryption algorithms suitable for IoT medical systems. The paper outlines a comparison between ten applicable cryptographic algorithms resulting fair analysis in terms of memory utilization and speed. The investigation deduces the optimal candidate algorithm for the proposed health care system considering the balance between the optimal requirement and the future threats.

Key Words : Healthcare e-system security; Data encryption; Internet of medical things; Light Weight Cryptosystems.

1. Introduction

Internet of things (IoT) is considered as a hybrid network that merges between the resources constrained networks and the Internet, where different objects can communicate and exchange information [1]. It is a promising field that integrates different technologies as well as different communication solutions. There are many IoT applications targeting several domains such as e-health, e-commerce, e-home etc [2]. The IoT revolution in the e-health will improve the quality of healthcare services as well as reduce the healthcare cost [3]. These factors helped optimize the healthcare field by innovating new devices and solutions. In other words, the IoT is remodelling the healthcare field in order to improve the social benefits through offering a continuous monitoring of patients and services as well as updating healthcare medical records [4]. The healthcare data provide real sensitive information that should be protected by law, in most countries, via Health Information and Portability Accountability Association (HIPAA) [5].

The data security is known as one of the most critical issues [6], it is an indispensable requirement especially for the operations and transactions that are based on data [1]. Therefore, data encryption is required before transmitting the data into the Internet public network [7]. Developing new technologies for healthcare environment without considering security will put privacy of patient's info vulnerable, so the integration of the IoT with the security protocols is the original known challenge [3]. In addition, the biggest problem facing most security measures is that the encryption algorithm consumes lots of time [8], this may result in dangerous delay putting the patient health potentially at risk or it may lead, in the worst case, to lose the patient life [9].

Investigating Cryptographic Algorithms for Securing Data in IoT Health-Care Systems (HCS) is considered a crucial task [5]. In fact, the HCS need cryptography to provide the required level of security entirely in order to be trusted [10]. HCS is commonly utilizing different crypto-algorithm, selected from asymmetric public key cryptography, symmetric key cryptography, and/or hash functions based on the application requirements and the available communication [9]. Merging between crypto methods from symmetric, asymmetric, and hash functions within same HCS may be found as medical hybrid crypto procedures [11]. Lately, selected crypto techniques are chosen and tuned to fit limited resources within HCS named as light-weight cryptography (LWC) making modern classification of IoT cryptography classified as shown in Figure 1. In this introduction, a briefing is given about main cryptography, i.e. asymmetric public key crypto, symmetric key cryptography and hash functions; where our focus of this work will be on LWC to be covered in a separate section later.

1.1. Asymmetric Public Key Cryptography

Asymmetric public key cryptography (PKC) is an encryption technique that depends on two asymmetric keys, i.e. public key and private key. PKC system uses a mathematical operation based on factorization or discrete logarithm problem, where calculating the keys (public and private key) from each other is almost impossible to compute [10]. The advantage of PKC is that it doesn't require any secure channel in order to exchange the secret data making it very suitable for the ad hoc and wireless sensor network [12]. The most common current algorithms of PKC known are RSA and Elliptic Curve Cryptography [11]. RSA taken from the names of its inventors: Rivest-Shamir-Adleman (RSA) depends on the product of two large random prime numbers building its security on complexity of number factorization.

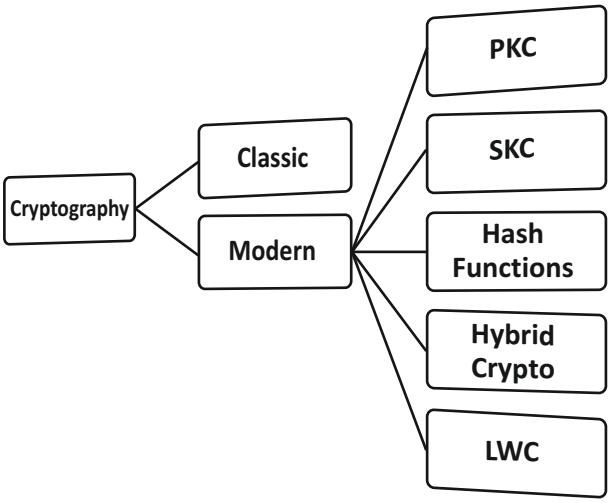

Fig.1: Modern cryptography classification methods

The other competing promising PKC method is known as Elliptic Curve Cryptography (ECC), which depends on finding the discrete logarithm from a random elliptic curve element. In fact, ECC is claimed to be more efficient than RSA providing same level of security [11]. However, both RSA and ECC work slowly when encrypting and decrypting large amount of data delivered via medical electronic sensors making the PKC still unpractical IoT crypto tools. Some attempts have been tried out minimizing PKC requirements but they remain suffering slower operation especially for our essential medical high demand cryptography [13]. It is noted that a single symmetric key cryptography encrypting tens of megabits using the same secret key ciphers will consume same amount of energy and time of a one single public key crypto operation [12]. In other words, higher speed of PKC algorithms demands a higher usage of RAM and a complex long code size, making symmetric key cryptography still being the tuned applicable solution, as described next.

1.2. Symmetric Key Cryptography

Symmetric key cryptography (SKC) is an encryption technique that uses one same secret key for both encryption and decryption. It is more preferred due to the resource constraints [14]. Symmetric encryption can be classified into two basic categories: Stream cipher and block cipher based on the data-bits grouping. The stream cipher encrypts the bits individually. It is accomplished by adding a plaintext bit into a bit of key stream. Block cipher encrypts a block of bits together immediately by using the same key. Both stream cipher and block cipher are widely used with current medical sensors although block cipher is believed to be more secure [15]. RC5, for example, is a block cipher claimed to be more secure and three times faster than the stream cipher RC4. However, involving the security protocols on top of the sensors normal operation increases the overhead of the data transmission and energy conservation. This can make using stream cipher for encryption seams more preferable than block cipher whereas the size of plaintext is equal to the cipher text size. This case is found in the Media Access Control (MAC) using only 16 bytes from the data frame of 60 bytes. Also, MAC will achieve the data integrity without the need for cyclic redundancy check (CRC) which is also used to detect errors in the received frame [15]. Indeed, for the block ciphers a mode of operation is required in order to achieve the semantic security. Semantic security means preventing the passive adversary from recognizing some information from the plaintext by just recognizing the cipher text except the message length. Thus, operation modes and the initialization vector (IV) are used to overcome the problem of data pattern by providing some proper randomization [16]. One common SCK algorithm implemented in medical sensors is known as the Advanced Encryption Standard (AES). It, AES, is an iterative block cipher procedure that depends on substitution and permutation network structure. It uses data blocks of size 128 bit with three optional key lengths that determine the number of rounds in the AES main algorithm, i.e. AES 128 bits needs 10 rounds, AES 192 bits requires 12 rounds, and AES 256 bits requests 14 rounds. It has been noted for some time that AES is the most energy efficient cipher [15] with high speed and excellent security; it is practical in its implementation for hardware [11] and software as well as its efficient running on different platforms [14].

1.3. Hash crypto algorithm

Cryptographic hash algorithm is a one way function that is impossible to invert used normally for integrity purposes. It takes data with various lengths as input then extracts the output with fixed length. Different hash functions are used to find whether a message have been changed by the attacker or not. The most common cryptographic hash functions used are Message Digest (MD) and Secure Hash Algorithm (SHA) [1]. In general,

hash functions are used to emphasize the integrity of the exchange messages, but increase the communication cost and usually considered as insufficient to detect message changes [17].

The high demand of saving energy, memory and computation requirements, used via the traditional algorithms like RSA, ECC, AES, MD, and SHA, emphasized the need for lightweight cryptography (LWC) as special tuned algorithms suitable for medical IoT sensors. It is important to realize that, investigating a security protocol for healthcare application is an open research major concern that is addressed in this work.

The flow of the paper is as follows. Next section, Section 2, will present the considered healthcare system model and its need for this lightweight cryptography (LWC) security. Section 3 presents the related work of the IoT and healthcare systems requirements for lightweight cryptography (LWC). Section 4 discusses ten candidate lightweight cryptography algorithms that are considered practical for securing our IoT medical applications. Section 5 describes the implementation and evaluation criteria followed by Section 6 detailing the comparisons and analysis study of the algorithms in terms of performance as well as memory utilization. Section 7 concludes this paper opening the research for attractive future work ideas that is promising for interesting expected results.

2. Healthcare System

Nowadays, it became an easy task to asses, monitor and track the patient's activities regardless of the patient's location and without needing for actual direct connection between doctor and patient [1]. The wireless medical sensors can play a big role in the Internet of Things (IoT) and the healthcare field [12]. Sensors can be used to collect the physiological data like blood pressure, body temperature, heart rate, sleep patterns…etc [16]. Medical data will be sent via the gateway device or smartphone to the hospital system for testing, evaluation and analysis. As shown in Figure 2, Remote Health Care Monitoring System (RHCMS) architecture is mainly made of five elements: Sensors, Smart phone, Smart-home unit (SHU), Internet connection, and Healthcare care services (HCS).

Normally the RHCMS is known to monitor the patients while they are participating in their normal life at home. Also, the RHCMS includes an emergency notification system in case needed. The data collected from the patient by sensors are sent through a wireless link to his mobile via the main blue line, as shown in Figure 2. Then, the data is transferred through the Internet and finally end up at the hospital HCS. The red line is considered a backup line for data to be sent through SHU often connected to the internet then to the HCS through fiber optic link to keep the records correctly updated and synchronized. The green line is used by the hospital if there is any feedback or emergency situation that HCS needs to alert the patient. The RHCMS model of Figure 2 is consisting of 4-basic operations including data collection, transmission, analysis, and evaluation, as detailed below.

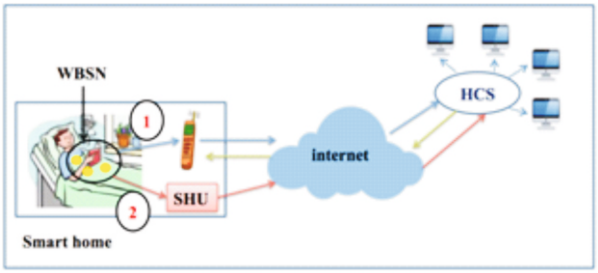

Fig. 2 : Remote health care monitoring system (RHCMS) architecture components

RHCMS Scenario basic operations:

1) Collect the patient data:
   a. Activate the sensor to collect the patient data.
   b. Store the patient data within sensors.
   c. Encrypt the data and place it in the appropriate format to prepare it for delivery (the transmission process).
2) Transmit the patient data:
   a. Send the data from the mobile phone through the internet.
   b. Receive the patient data at the hospital end.
   c. Decrypt the data received and check its completion.
3) Analyze the patient data:
   a. Test the data received by a personnel applying HCS specific algorithm to classify it.
   b. Observe testing results by responsible medical doctor and evaluate it documenting the feedback.
   c. Transmit feedback to the patient or professional agency as needed.
4) Action toward the patient situation and HCS:
   a. Emergency alert is announced, if there is any urgent situation or risk needing the patient's immediate action.
   b. Request ambulance sending to the patient home registering the situation to the hospital in charge.
   c. Notify the family members about the emergency.

Healthcare field is already a targeted industry for attackers [18] where the sensors are considered as an easy

target to exploit [19]. Also, the medical fraud is a big risk intruding the health care data that may lead to many challenging mysterious problems. Medical fraud problem can lead to illegally increase in financial cost as well as it can provide wrong treatment causing loss of lives. In fact, some strange medical bills have been reported while patients are in the queue or waiting room watching a movie with the claim of receiving group therapy. Medical fraud, in general, is becoming a risky huge industry. It is easily providing financial unnecessary interest especially when incorrect reporting or alerting is given [3]. Cryptography can help in reducing the effect of the critical situation with data, i.e. where medical data appears in unreadable format, and the encryption algorithms can be employed to protect this medical information [1].

3. Lightweight Cryptography

Lightweight cryptography (LWC) is defined as a crypto algorithm suitable for limited resource constrained environment such as medical sensors, RFID tags, and portable health care devices [20]. Its data security can be stream based or block based, but must be keeping acceptable level of immediate usage security [21]; i.e. researchers can misunderstand the lightweight concept as less secure protocol, but it's not the case. The lightweight crypto security is only lightweight in resource consuming for instantaneous treatment and not reduction in the security or privacy weight [22]. In general, the LWC offers the adequate amount of (80 to 128 bit) security for direct usage of IoT medical application tuned for the significant reduced amount of resources [23]. Although 80 bit security can be adequate for constrained devices, like RFID tags and 4-bit micro-controllers, 128 bits is representative for typical medical applications. For one way trust authentication, 64 to 80 bit security is reported acceptable. In general, any selected implementation of LWC algorithms must be compact, fast, low power consuming while preserving adequate required security [23].

Choosing the appropriate encryption/decryption algorithm is difficult due to the resource constraints of the medical sensors and their immediate connection concerning patient's health and life. Obviously, complex crypto algorithm demands heavy computation as well as large amount of resource. Where in the IoT and especially for a time constrained environment like e-health, there is an essential need for reasonable amount of complex security while consuming minimum resources. The communication and computation capabilities of the sensor node play a big role in determining the suitable crypto algorithm. For the focus of this work, the specific considerations addressed to maintain the appropriate crypto for IoT medical sensors depend on the energy consumption, memory occupation, and execution time as briefly described next.

3.1. Energy consumption

Most portable medical devices are built with limited energy or battery resources [9]. Applying specific security solutions will depend on the sensors communication and computation which affect the energy utilization of the limited power available [5]. The energy consumed by sensor transducer, by the microprocessor for the computation, and by the communication between sensor nodes, are more expensive than computation with respect to inside the sensors [24]. It is noted in the literature the approximation that transmitting one bit in WSN consumes as much power as implementing eight hundred to one thousand instructions [18]. This made our study of the quantity of energy needed for cryptography not to be measured separately; it can be estimated proportional to the clock cycles needed for the operations noted as the encryption cycles.

3.2. Memory occupation

As the energy resources, the memory is also very limited within the healthcare sensors. For instance, in the smart dust project tiny-OS occupied 3500 bytes of memory, allowing for remaining bytes of about only 4500 bytes for the application and the security which are insufficient [5]. This made our exploration focus on studying the amount of memory, i.e. RAM and ROM, needed by the cryptography algorithm.

3.3. Execution time

The healthcare field used to ignore the security part while focusing on the emergency and direct life saving. However, this lead to information leakage and security breach causing severe medical problems [1]. The correct scenario is to consider the security as well as the medical emergency need, i.e. by achieving a balance between security and requirement of medical emergency. For our study, the execution time of the encryption algorithm should consider the cryptography key set up time added to the encryption time.

In fact, the fast execution of crypto algorithm can minimize the energy consumption and maximize the lifetime of the battery. All three parameters, namely energy consumption, memory occupation, and execution time, are affecting each other and the optimization in all is required as described later in the section about implementation and evaluation.

4. Considered Crypto Algorithms

Many crypto techniques are developed for data security purposes. However, not all of them are trusted to be practical for medical applications due to the sensitivity of the health situations [17]. For our study, this section selects ten crypto algorithms understood most suitable implemented for IoT medical purposes. It is to be noted that most encryption algorithms applicable for

constrained device applications are built based on tuning the known Advanced Encryption standard (AES) improved as required. The chosen ten algorithms found suitable are symmetric lightweight cryptographic procedures with all needed security features for IoT HCS introduced briefly as follows.

4.1. Advanced Encryption standard (AES)

Advanced Encryption standard (AES) was announced by National Institute of Standards and Technology (NIST) in 1998 [20]. The AES is a block cipher with substitution and permutation network structure. It has 128 bits presented by 4*4 matrixes where four basic operations are applied to the cipher state as follows: Substitute Bytes, Shift Rows, Mix Columns and Add Round key [21]. It is noted that the AES shifting operation is very helpful in improving the speed of the algorithm. The best single-key cryptanalysis for AES-128 was met in the middle attack on seven out of ten rounds according to the published paper: Triathlon of Lightweight Block Ciphers for the Internet of Things [20] which still makes its security acceptable.

4.2. Camellia

Camellia is a block cipher with 128 bit data block and variable key size of 128, 192, 256 bits designed to work as required for both software and hardware implementations [25]. It is a feistel cipher with processing capabilities and security levels considered very similar to AES [23]. Camellia is accepted for use by the ISO/IEC and the projects CRYPTREC and NESSIE. Its software implementation is faster than hardware making the hardware implementation reported not recommended [23]. Camellia can work for different applications were it depends on some logical operations on top of 8*8 S-boxes substitutions. Its' security is based on the fact that no linear attacks or differential attacks exceeding 128-bits is reported successful making its security applicable [23].

4.3. International Data Encryption Algorithm (IDEA)

International Data Encryption Algorithm (IDEA) is a block cipher designed to replace the data encryption standard (DES), which has been common in the last decade [11]. Its main improvement of IDEA over DES is enlargement of small key size which was contradicting current standards.

The IDEA is a specially developed software architecture optimization announcing it as standard and distributed widely to be used for commercial applications. The IDEA manly contains the simple arithmetic computations of addition, multiplication and XOR operations. Its applicability for software more than hardware comes from the fact that it does not use any S-boxes or lookup tables [21]. Indeed, its security is good enough for HCS since no major algebraic or linear attack was stated against it except the attack known as the impossible differential attack, reported for 3.5, 4 and 4.5 rounds, making IDEA still a valid crypto system to be used [21].

4.4. Leak Extraction (LEX)

Leak Extraction (LEX) is software oriented stream cipher from eSTREAM project [26]. LEX can be described as an AES cipher with some modifications to gain faster operations, i.e. LEX is faster than AES, by at least 2.5 times. LEX utilizes the standard AES key schedule, but implemented via a modified key stream extracted from the internal state of certain rounds. LEX setup phase encrypts 128 bit key and the initial vector (IV) of 128 bit to create 256 bit to be starting the secret state. Then, the secret state is changed during round functions using the key that is adjusted after around 500 AES encryptions in order to improve security [26].

4.5. Light Encryption Device (LED)

Light Encryption Device (LED) is lightweight block ciphers described as an extension of AES based ciphers. It is a more friendly algorithm operating on implementations of hardware more than software, but still running with reasonable software performance [27]. This cipher is distinguished by having no strong dependence on the key schedule and the round keys have been replaced by part of the master key. The LED allows deriving very simple bounds on the number of active S-boxes during a block cipher encryption. Since the key schedule is very simple, this analysis can be done in a related-key model as well; i.e. the bounds apply even when an attacker tries to mount a related-key attack. Although AES-based approaches are well-suited to software, they don't always provide the lightest implementation in hardware, where LED is using special techniques resolving this contradiction. It is been noted that differential attacks cover 16 rounds out of 32 related to LED-64 and 24 rounds out of 48 related to LED-128. However, no attacks appeared (up to our knowledge) on LED-80 until this work [20].

4.6. Rabbit

Rabbit is an efficient stream cipher published in 2003 dedicated for software implementation. Rabbit was inspired from the complex behavior for the real values related to chaotic maps. It depends on bitwise operations like shifting, bitwise XOR and concatenation, which resulted in real fast performance [23]. The algorithm takes 128 key bit lengths as an input and for every iteration it produces 128 bits pseudo random from mixing the internal states. Also, there is no need for S-boxes or lockup tables. The best attack against Rabbit is exhaustive key search attack [28] making LED still considered secure for our applications.

4.7. RC5

RC5 was developed by Ronlad Rivest in 1994. It is a block cipher where the number of rounds, the block size and the key size is variable and left free to be chosen [22]. Its functions depend on general operations of the

microprocessors such as XOR, cyclic shift and modular addition. RC5 is suitable for medical applications since its attraction features are fast execution with less memory usage. However, its negative drawback is that RC5 may face some security degradation due to its noted weak diffusion [27]. In this work, our study of RC5 is considered for 32 bit words, 12 rounds and a 16 byte key. Its disadvantage problem of differential cryptanalysis attack is ignored (for our medical application) assuming that it will need at least the complete codebook to achieve the answers, which estimated to be equal to 2^64 cipher-texts [20]. This assumption makes RC5 relatively secure for data of our IoT LWC health-care systems.

4.8. Salsa20

Salsa20 is a software oriented stream cipher, similar to IDEA, based on AES simple arithmetic operations. In fact, Salsa20 basic functions can be represented as merging expansion operations with hash function. The expansion combines cryptographically 16 bytes data to 32 bytes secret key, as well as 8 byte nonce to 8 byte of block counter, all mixed up selectively to form 64 byte block.

The Salsa20 involves a hash function in its operation to constitute a key stream, where the hash output is entered to XOR operation with the 64 byte block [26]. The best attack reported to affect Salsa20 security was using 256 bit brute force search which needs more time than affecting medical security, i.e. Salsa20 is considered still acceptable for our IoT LWC health-care systems.

4.9. SIMON

SIMON is a feistel block cipher created by National Security Agency (NSA). Its main objective is similar to our application for protecting data in very constrained HCS environments. SIMON procedure core operations can be observed as simple round functions of bitwise AND, bitwise OR, and left circular shifts, affecting the sensitive data to be secured. It is developed aiming hardware implementations, i.e. to improve the performance running on hardware, but found achieving acceptable results on both the software as well as hardware data crypto systems.

The attack affecting this method is found meaningful after rounds 48 out of 69, especially related to SIMON-128 [29], which made SIMON block cipher good enough to be included in our study.

4.10.SPECK

SPECK is designed for the software execution on the microcontrollers [29]. However, its optimization is found achieving high performance in both software and hardware [23]. SPECK uses feistel cipher structure, similar in principle to Camellia and SIMON cryptography. SPECK procedure runs branches of bitwise process developed for every round using crucial shifts in both directions, modular addition, and bitwise XOR operations, which made its main modification variation among other AES similar crypto algorithms. In fact, both SPECK as well as SIMON are not just normal single version block ciphers, but they are considered families of multi block ciphers where each of them contain ten different block ciphers with different keys and block sizes. The real benefit to this work found in SIMON and SPECK is them having more flexibility in their specifications operation that helped in optimizing the speed and memory RAM utilization. SPECK security is acceptable, i.e. the cryptography attacks on it reported reached around 70 percent but couldn't make more exceed. This percentage can be reduced by increasing the number of rounds but this may affect the performance efficiency. To be precise, the best effective attack stated against SPECK46/128 is on rounds 19 out of 27, which is making it satisfactory for our IoT LWC protection of health-care systems.

5. Implementation Evaluation Criteria

The ciphers, previously discussed in Section 4, are implemented in software running on 8-bit AVR ATmega128 microcontroller ready for software evaluation. The AVR microcontroller is designed by Atmel using 8 bit RISC microcontroller that offers 128-Kbytes of flash memory and 4-KBytes of SRAM. AVR devices supports 133 instructions with 32 general purpose registers connected immediately to the arithmetic logic unit, as described in [20]. In fact, the RAM and ROM utilization affects the performance of cryptosystems [30] making the analysis and comparison evaluation criteria involve mainly speed and memory occupation. The speed is measured by the number of clock cycles per byte and the memory occupation is defined by the RAM and ROM used, making the evaluation independent and fair for all ten considered LWC algorithms.

It should be mentioned that, for most encryption algorithms, the evaluation metrics of the software execution is found different than them for the hardware implementation estimating the memory utilization. The software operation memory occupation focus on the requirements of RAM and ROM, while the hardware focus on the area and specifically the hardware chip size [30]. However, both software and hardware implementations pay similar attention to the performance speed, which can be unified by estimating the required number of rounds running the algorithms affecting the clock cycles and speed [21].

6. Comparison & Analysis

In this section, we evaluate the ten candidates LWC encryption algorithms software implementations based on previously defined (in Section 5) memory and speed, as found in Table 1. The memory utilization evaluation of the lightweight ciphers is extracted from RAM and ROM usage measures. Consider the RAM utilizations in bytes

for all ten LWC listed in Table 1. This RAM usage can be classified into four categories based on RAM used in bytes, i.e. below 35 bytes, between 35 and 150 bytes, between 150 and 300 bytes, and above 300 bytes. Observe that the first category evaluation involves four crypto algorithms, namely IDEA, Camellia, SPECK, and RC5, showing the best RAM usage, respectively. The next group has two crypto procedures, SIMON and AES. The third category also involves two crypto algorithms, Rabbit and LED, while the two remaining procedures, Salsa20 and LEX, are in the lowest (fourth) preferred rank, i.e. largest memory RAM utilization.

As a matter of fact, the RAM utilization requirements for the first two categories are to be considered winning assuming them acceptable for our selections, i.e. for the first two categories algorithms considered are IDEA, Camellia, SPECK, RC5, SIMON, and AES.

To elaborate our study of the memory utilization fairly, we furthermore pay attention to the ROM used bytes in addition to the RAM utilization study. Consider the evaluation of lightweight ciphers ROM deployment measure listed in Table 1.

Table 1
Speed and Memory Comparison Results

Considered Crypto Algorithm	Speed	Memory Occupation	
	Clock Cycles per byte	RAM used (bytes)	ROM used (bytes)
AES (128/128)	24695	54	1708
Camelia (128/64)	Not in range	12	1262
IDEA (128/64)	Not in range	0	1140
LEX (128/320)	8061	432	21398
LED (80/64)	1074961	242	2600
Rabbit (128/128)	Not in range	216	1714
RC5 (128/64)	75871	33	1614
Salsa20 (128/128)	90802	322	4478
SIMON (96/64)	39313	40	752
SPECK (96/64)	28401	29	628

Similar to RAM study classifications, the ROM usage can be classified into four categories in terms of ROM used measured in bytes, below 1000 bytes, between 1000 and 2000 bytes, between 2000 and 5000 bytes, and above 5000 bytes. As illustrated in Table 1, the first classification involves SPECK and SIMON which shows the best cost in term of memory ROM utilization. The second group has five crypto procedures, IDEA, Camelia, RC5, AES, and Rabbit, ordered by most efficient ROM usage respectively. The Third category involves LED and Salsa20 while the last algorithm in terms of this ROM utilization exploration is LEX. Most practical LWC procedures need memory size in the first two categories which were of ROM with less than 2000 bytes, i.e. the preferred ROM efficiency selected ciphers are ordered as SPECK, SIMON, IDEA, Camelia, RC5, AES, and Rabbit.

The evaluation of speed of the LWC is measured by the clock cycle per byte involving the LWC algorithms, where seven procedures have been selected as applicable providing fair comparison as listed in Table 1.

The comparison of speed shows that LEX is the fastest cipher in this study; then, AES, SPECK followed by SIMON, respectively, are in the next rank. The AES-128 on the AVR 8bit microcontroller is slightly faster than SPECK(128,128) with 17%, but SPECK uses less memory with higher speed than AES-128, which when its key size increase, SPECK will overtake the AES throughput due to the attractive rounds number. In addition, SPECK 64/128 which has the same key size as AES-128 but with a smaller block size is smaller and faster than AES. Furthermore, MSP430 platform in the literature [18], the SPECK is the reported as fastest algorithm in both efficiency and throughput. In fact, SPECK is documented as 23% faster than AES using negligible RAM and 81% less ROM resulting the energy consumption of 35% fewer than AES.

Indeed, lots of efforts have been performed reshaping the AES block cipher as a candidate solution for the LWC for both hardware and software implementations. However, AES can operate perfectly on some devices like laptops and smart phones, but not as good for the most constrained environment where there is some limitations, i.e. especially for current and tomorrow's HCS needs. The AES is expected to be modified more and more to be fast, which will suffer also in its larger size and complex design making it out of track for LWC. Of Course, different applications needs different level of security, but not all of them will require a high level of security exactly as the AES guarantees. In other words, the scarce of resource can determine the optimal solution in most LWC IoT devices.

It is to be noted that most available researches were focusing on the challenges that are related to optimizing for a specific platform. It is important to realize that if the block cipher achieves the highest performance on dedicated platform, this one can be out of the comparison on most other platforms. Moreover, it may face a limit in usability on the end of the platform's life which is continually changing very fast. From the other side, it will not be sufficient to delay the discussion until the future devices or the IoT devices appear, but we can study the performance of ciphers to find simple algorithm that can be efficiently everywhere. With all this in mind, the SIMON and the SPECK algorithms were found the best. Both of them are designed especially to improve the

security on the very constrained environments such as our IoT medical application. They are currently built as general block ciphers expected to be important involved in many future applications of the IoT area. In addition, the two of them, the SIMON and the SPECK, are practically flexible working well on many different platforms that can be adjustable for the innovative future use. As a matter of fact, our study found that the SPECK algorithm is better than SIMON procedure in the software implementation; that's because the SIMON needs some preparation bit moves and that is due to the fact that several operations made by using single word of the intermediate cipher-text. Also, SIMON requires several copies to be registered making it less preferred when compared to the SPECK which can be executed completely by in-place operations and no need for any move operations which is detailed in [29].

7. Conclusion

Healthcare e-field is becoming an innovative targeted industry for data security attackers. The medical sensors are considered as an easy target to exploit where its affect is very catastrophic reaching murderous life death problems. On the purpose of protecting the medical data in the IoT field, data encryption is required before transmitting the data into the Internet public network.

The high need of energy, memory and computation requirements of the traditional algorithms like RSA and AES, emphasized the need to adapt the LWC algorithms. This work focus on ten LWC encryption algorithms implemented in software suitable for constrained medical applications. All ten LWC algorithms have been studied and analyzed in terms of speed and memory utilization. Based on our evaluation, two algorithms the SPECK and the SIMON have been found efficient and suitable although they are currently built as general cryptosystems. They both are anticipated to be central in many potential IoT applications. The main advantage of those two LWC algorithms is their flexibility functioning on different adjustable platforms for innovative today and future use.

This study concluded that the SPECK algorithm is further better than SIMON procedure in the software implementation. The SPECK is found the best, fast, small, simple and flexible. Last but not least, it supports broad range of block and key sizes which made it perfect for the different IoT devices. In the final analysis, the SPECK is the optimal choice for the software implementation promising for attractive future research.

Acknowledgments

Thanks to Umm Al-Qura University (UQU) for encouraging this research work.

References

[1] L. Yehia, A Khedr, A. Darwish, Hybrid Security Techniques for Internet of Things Healthcare Applications. *Advances in Internet of Things, 5(03), 2015.*

[2] A. Gutub, Exploratory Data Visualization for Smart Systems, *Smart Cities 2015 - 3rd Annual Digital Grids and Smart Cities Workshop,* Burj Rafal Hotel Kempinski, Riyadh, Saudi Arabia, May 2015.

[3] S. Niranjana, A. Balamurugan, Intelligent E-Health Gateway Based Ubiquitous Healthcare Systems in Internet of Things, *International Journal of Scientific Engineering and Applied Science (IJSEAS),* 1(9), December 2015.

[4] A. Gutub, Social Media & its Impact on e-governance, *ME Smart Cities 2015 - 4th Middle East Smart Cities Summit,* Pullman Dubai Deira City Centre Hotel, Dubai, UAE, December 2015.

[5] G. H. Zhang, C. C. Y. Poon, Y. T. Zhang, A review on body area networks security for healthcare. *ISRN Communications and Networking,* 2011.

[6] S. Al-Nofaie, M. Fattani, A. Gutub, Merging Two Steganography Techniques Adjusted to Improve Arabic Text Data Security, *Journal of Computer Science & Computational Mathematics (JCSCM), 6(3),* doi: 10.20967/jcscm.2016.03.004, pp. 59-65, September 2016.

[7] A. Gutub, H. Tahhan, Efficient Adders To Speedup Modular Multiplication For Cryptography, *5th IEEE International Workshop on Signal Processing and its Applications (WoSPA),* University of Sharjah, U.A.E. 18 - 20 March 2008.

[8] A. Gutub, A. Tabakh, A. Al-Qahtani, A. Amin, Serial vs. Parallel Elliptic Curve Crypto Processor Designs, *IADIS International Conference: Applied Computing,* Fort Worth, Texas, pp. 67-74, October 2013.

[9] A. Gutub and E. Khan, Using Subthreshold SRAM to Design Low-Power Crypto Hardware, *International Journal of New Computer Architectures and their Applications (IJNCAA),* 1(2), pp. 474-483, 2011.

[10] A. Gutub, High Speed Low Power GF(2k) Elliptic Curve Cryptography Processor Architecture, *IEEE 10th Annual Technical Exchange Meeting,* KFUPM, Dhahran, Saudi Arabia, March 2003.

[11] A. Gutub and F. Khan, Hybrid Crypto Hardware Utilizing Symmetric-Key & Public-Key Cryptosystems, *International Conference on Advanced Computer Science Applications and Technologies (ACSAT),* Palace of the Golden Horses, Kuala Lumpur, Malaysia, November 2012.

[12] S. Rehman, M. Bilal, B. Ahmad, K Yahya, A. Ullah, O. Rehman, Comparison based analysis of different cryptographic and encryption techniques using message authentication code (mac) in wireless sensor networks (wsn). *arXiv preprint arXiv:1203.3103,* 2012.

[13] A. Gutub, Merging GF(p) Elliptic Curve Point Adding and Doubling on Pipelined VLSI Cryptographic ASIC Architecture, *International Journal of Computer Science and Network Security (IJCSNS),* 6(3A) , pp. 44 – 52, March 2006.

[14] K. Chaudhari, M. Borole, A Survey on Various Cryptographic Algorithms for Security Enhancement. *International Journal of Computer Applications,* 110(7), January 2015.

[15] L. Gupta, R. Jain, Security in low energy body area networks for healthcare, online, available at: http://www.cse.wustl.edu/ ~jain/cse 571-14/ftp/ban/index.html ; 2014.

[16] L. Casado, P. Tsigas, ContikiSec: A Secure Network Layer for Wireless Sensor Networks under the Contiki Operating System, *14th Nordic Conference on Secure IT Systems (NordSec), Lecture Notes in Computer Science,* 5838, Springer-Verlag, pp. 133-147, 2009.

[17] A. Trad, A. Bahattab, S. Othman, Performance trade-offs of encryption algorithms for Wireless Sensor Networks. *IEEE World Congress on Computer Applications and Information Systems (WCCAIS)*, pp. 1-6, January 2014.

[18] H. Baldus, K. Klabunde, G. Muesch, Reliable set-up of medical body-sensor networks. *European Workshop on Wireless Sensor Networks,* Springer Berlin Heidelberg, pp. 353-363, January 2014.

[19] S. Aly, T. AlGhamdi, M. Salim, H. Amin, A. Gutub, Information Gathering Schemes for Collaborative Sensor Devices, *Procedia Computer Science, 32, Elsevier,* pp. 1141-1146, June 2014.

[20] D. Dinu, Y. Le Corre, D. Khovratovich, L. Perrin, J. Großschädl, A. Biryukov, Triathlon of Lightweight Block Ciphers for the Internet of Things. *IACR Cryptology ePrint Archive,* 209, 2015.

[21] T. Eisenbarth, S. Kumar, C. Paar, A. Poschmann, L. Uhsadel, A survey of lightweight-cryptography implementations. *IEEE Design & Test of Computers,* 24(6), pp. 522-533, 2007.

[22] T. Guneysu, Lightweight Cryptography for Security and Privacy, Springer, 2016.

[23] C. Manifavas, G. Hatzivasilis, K. Fysarakis, K. Rantos, Lightweight cryptography for embedded systems - a comparative analysis, *Data Privacy Management and Autonomous Spontaneous Security,* Springer Berlin Heidelberg, pp. 333-349, 2014.

[24] S. Aly, T. Alghamdi, M. Salim, A. Gutub, Data Dissemination and Collection Algorithms for Collaborative Sensor Devices Using Dynamic Cluster Heads, *Trends in Applied Sciences Research,* 8(2), Academic Journals Inc., USA, pp. 55-72, 2013.

[25] K. Aoki, T. Ichikawa, M. Kanda, M. Matsui, S. Moriai, J. Nakajima, T. Tokita, Camellia: A 128-bit block cipher suitable for multiple platforms - design andanalysis, *International Workshop on Selected Areas in Cryptography,* pp. 39-56, 2000.

[26] G. Meiser, T. Eisenbarth, K. Lemke-Rust, C. Paar, Efficient implementation of eSTREAM ciphers on 8-bit AVR microcontrollers, *IEEE International Symposium on Industrial Embedded Systems,* pp. 55-66, June 2008.

[27] J. Guo, T. Peyrin, A. Poschmann, M. Robshaw, 2011, The LED block cipher, *International Workshop on Cryptographic Hardware and Embedded Systems,* Springer Berlin Heidelberg, pp. 326-341, September 2011.

[28] M. Boesgaard, M. Vesterager, T. Christensen, E. Zenner, The stream cipher rabbit. *ECRYPT Stream Cipher Project Report,* 6, 2005.

[29] R. Beaulieu, D. Shors, J. Smith, S. Treatman-Clark, B. Weeks, L. Wingers, SIMON and SPECK: Block Ciphers for the Internet of Things, *IACR Cryptology ePrint Archive,* 2015.

[30] A. Gutub, Subthreshold SRAM Designs for Cryptography Security Computations, *2nd International Conference on Software Engineering and Computer Systems (ICSECS),* Universiti Malaysia Pahang, Kuantan, Malaysia, June 2011.