Multimedia Tools and Applications (2019) 78:32633–32657
https://doi.org/10.1007/s11042-018-6801-z

Enhancing speed of SIMON: A light-weight-cryptographic algorithm for IoT applications

Norah Alassaf[1] · Adnan Gutub[1] · Shabir A. Parah[2] · Manal Al Ghamdi[3]

Received: 11 April 2018 / Revised: 13 October 2018 / Accepted: 23 October 2018 /
Published online: 5 November 2018

Abstract

Multimedia communication is revolutionizing all major spheres of human life. The advent of IoT and its applications in many fields like sensing, healthcare and industry, result exponential increase in multimedia data, that needs to be shared over insecure networks. IoT driven setups are however constrained in terms of resources as a result of their small size. From data security point of view a conventional algorithms cannot be used for data encryption on an IoT platform given the resource constraints. The work presented in this paper studies the performance of SIMON cryptographic algorithm and proposes a light-weight-cryptography algorithm based on SIMON for its possible use in an IoT driven setup. The focus is on speed enhancement benefitting from software prospective, making it different than common studies mostly reflecting hardware implementations. To achieve performance in practical prospective, the contribution looks into SIMON cipher's characteristics considering utilizing it for internet of things (IoT) healthcare applications. The paper suggests further improvement to implement the original SIMON cryptography in order to reduce the encryption time and maintain the practical trade-off between security and performance. The proposed work has been compared to Advanced Encryption Standard (AES) and the original SIMON block cipher algorithms in terms of execution time, memory consumption. The results show that the proposed work is suitable for securing data in an IoT driven setup.

Keywords Internet of things · Medical data · AES · SIMON cipher · Light weight cryptography

✉ Shabir A. Parah
shabireltr@gmail.com

[1] Department of Computer Engineering, Umm Al-Qura University (UQU), Makkah, Saudi Arabia

[2] Department of Electronics and Instrumentation Technology, University of Kashmir, Srinagar, JK, India

[3] Department of Computer Sciences, Umm AL-Qura University, Makkah, Saudi Arabia

1 Introduction

Multimedia communications play a pivotal role in Internet of Things (IoT) based applications like environmental monitoring, healthcare, home automation and surveillance. Multimedia-IoT is presently receiving huge attention from research community round the globe. Since Internet of Things (IoT) is supposed to play an effective role in supporting the everywhere computing, permitting different devices and machines to communicate and interact, and to facilitate exchange of data; a multi-pronged research mechanism is globally underway to improve various aspects of IoT devices and networks. One of the main areas that needs attention is the security of data in IoT driven multimedia networks. Network architecture of the IoT has three basic layers: perception layer, network layer, and application layer [46]. The perception layer can be defined as the source of information collection. The network layer is used to connect the perception layer to the user application layer. Finally, the application layer is used to involve users into the scenario. Implementing IoT play increasing role impacting different fields such as smart transport, energy, cities, and healthcare applications [23]. Today, the spread of many small devices forces publicizing the internet of things (IoT). In fact, IoT devices are manufactured by large number of companies and are being used for important and critical objectives. In other words, the most important advantage of Internet of Things (IoT) is achieving full communication between heterogeneous devices, heterogeneous networks and heterogeneous processing bandwidths. Thus, if the IoT can't merge the multimedia in an organized way, it will not be able to benefit from multimedia-based services and applications. In other words, involving multimedia in IoT is an important research direction that will develop new architectures, protocols as well as scenarios such as adapting security in health monitoring environment. The presence of remote healthcare monitoring systems has led to reduction in the cost of treatment while enhancing the quality of services. Studies show that the number of elderly people is steadily growing [49]. Thus, the need for more hospitals to tackle the health care issues is increasing and so is the treatment costs. In such a scenario, electronic healthcare can reduce the burden and share an ample workload of the conventional care. However, successful deployment of healthcare systems depends on having the adequate security and privacy of the patient's data [22].

In the case of transmitting medical data also referred to as Electronic patient record (EPR), IoT security and privacy are very important to make people trust these systems. Encryption is the proven solution to be efficient in data security and effective retention [1]. However, it is still considered as major challenge for IoT raising valid research questions especially in the case of determining the appropriate algorithm to be used [4]. A common solution is to secure data via trusted cryptography, i.e. symmetric-key or public-key cryptography [26]. Many research works have been presented earlier to secure data via RSA or more advanced elliptic curve cryptography [27]. However, when it comes to the highly constrained portable devices, these traditional cryptographic algorithms needs significant high resources in order to execute [20]. Some research proposed constrained solution via hardware special arithmetic implementation involving efficient extraordinary adders [28] or redesigning SRAM sub-threshold crypto hardware for low-power utilizations [25]. Others even further presented investigation slightly modifying the crypto algorithm by merging its arithmetic on pipelined VLSI cryptographic ASIC architecture [21], which is found currently unpractical for healthcare mobile devices demanding more innovative research. In fact, many proposed cryptographic algorithms have appeared for this purpose, but found to have disadvantages like, high computational cost, increased implementation time and computation resources [18]. In order to take care of the discussed disadvantages, light-weight-cryptography (LWC) is being used now-a-days. This approach of LWC algorithms are found more suitable to help secure information wherever portable systems like those requires in healthcare are used [1]. Light-weight cryptography (LWC) algorithms take the IoT device limitations in full consideration. It puts priority for available ROM and RAM memory, computation limitations and energy consumption [38]. As a matter of fact, the software implementation is depending on the evaluation metrics that are distinct than hardware. The software implementation is mainly based on memory requirements and the number of clock cycles. Optimizing or designing a particular lightweight encryption algorithm requires the researcher to pay attention to the point of balance between the competing objectives, i.e. security, cost and performance [37]. The execution time, memory consumption and speed for the algorithms SIMON, SPECK and AES are implemented and evaluated previously [8] showing interesting features. Multimedia IoT, connects devices to medical equipment and service networks with an aim to provide effective services in the health sector. The critical nature of data being transferred over such networks requires that it should be secured before its transit. A number of security approaches for such a data could be found in [13, 15–17, 33, 39, 40, 42]. However most of the literature does only talk about securing the data without taking various constraints of IoT scenario in consideration [35, 43]. In this work we intend to improve the original SIMON algorithm to work better with regard to the software performance. Indeed, a large number of reported works have been focused on the challenges of modifying an algorithm based on a specific platform. But, the critical issue here is that, the same algorithm may work on two different IoT devices and give different performance. Thus, it was better to look for algorithms that works well and is easy to implement on different platforms, such as SIMON and SPECK. Both they were flexible, secure and perform efficiently in multiple platforms rather than single one.

The structure of the rest of the paper is as follows: Section 2 discusses the security scenario and the remote healthcare monitoring system in the IoT. It further discusses the importance of the IoT architecture. Section 3 describes the data confidentiality in IoT medical sensors. Section 4 provides a brief description of AES, and original SIMON followed by our improvement prospective to gain the enhancement version of SIMON in Section 5. Section 6 evaluates the performance comparing the three encryption algorithms: AES, original SIMON and our optimized version of SIMON expressing the results in terms of execution time, memory occupation, and the speed performance. Finally, Section 7 concludes the work.

2 Security within IoT remote healthcare monitoring system

The integral architecture of IoT allows the easy connectivity, communication, control and useful applications to be developed. The architecture of IoT has three basic layers, as shown in Fig. 1. The perception layer has a different nature belongs to the heterogeneous equipment's and the collecting way. Especially, with the existence of resource-constrained nature found in IoT devices, like the sensors, radio frequency identification (RFID), barcode etc. The security solutions must achieve practical performance regarding energy consumption, i.e. number of round trips, bandwidth requirements, memory and computing needs. As clarified in Fig. 1, there is no direct communication between the outmost layer (perceptual layer) and the human (application layer). Also, the number of constrained devices surpasses any other number. With this consideration, the IoT structure outer layer (perceptual layer) is covered by billions of controlled devices, which is a feature that can't be skipped easily, making medical doctors and IoT devices are not able to communicate directly and that because the limitations in networks and devices within the perceptual layer despite the fact that the IoT devices e.g. sensors can communicate with the network directly.

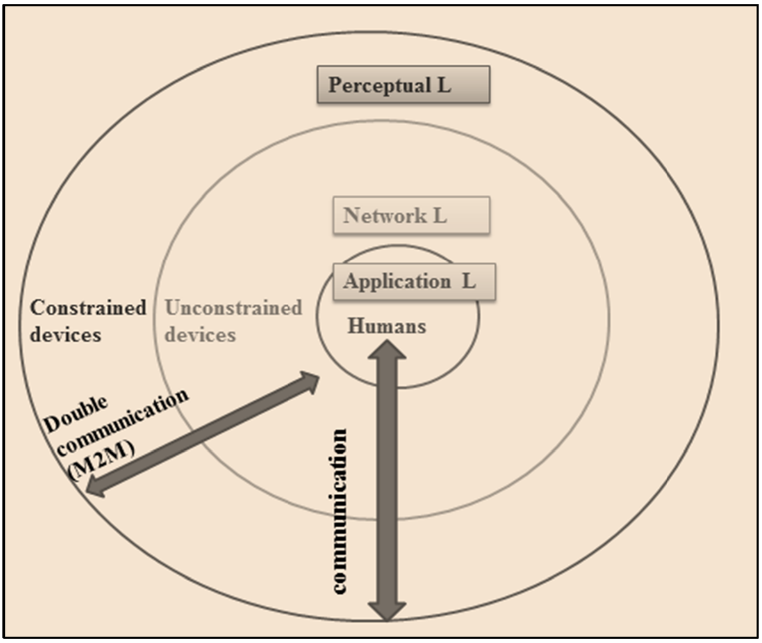


Fig. 1 IoT architecture

A typical IoT architecture showing massive deployment of machine to machine communication has been depicted in Fig. 1. It could be seen that the IoT setup works utilising a huge deployment of sensing and controlling capabilities. The IoT is considered as the large-scale deployment of machine type communication (MTC) devices that implement sensing actuation operations through minimal human interaction. The number of objects connected to the internet is supposed to exceed the number of people in the world.

This is due to the fact that IoT is based on rationale of connecting various physical devices through internet, e.g. the number of medical devices being connected to internet is rapidly increasing with each passing day [2]. Actually, the healthcare systems are considered as the biggest potential market for IoT with the heart rate monitoring determined as the top benefit [41]. The IoT devices used to apply emergency notification system and remote health care monitoring system by using simple devices such as blood pressure heart rate monitors accessing data into more sophisticated devices. These particular devices can monitor specialized implants used for "pacemakers", as a known example [24]. Creating such a system will be very useful especially for the elderly people and whomever with chronic disease. Medical data will be sent via the Smartphone to the hospital system for testing, evaluation and analysis as shown in Fig. 2.

In order, for these systems to be reliable and successful they must maintain high performance, work quickly, monitor the patient's condition immediately and respond correctly safely accordingly [36]. In addition, these systems must earn the patient confidence in maintaining his medical information and privacy.

The sensors, smart phone, smart-home network, public communication network and the hospital network are exposed to different security threats and most of the time the main reason was the inherited vulnerabilities from the wireless communication. There are a various

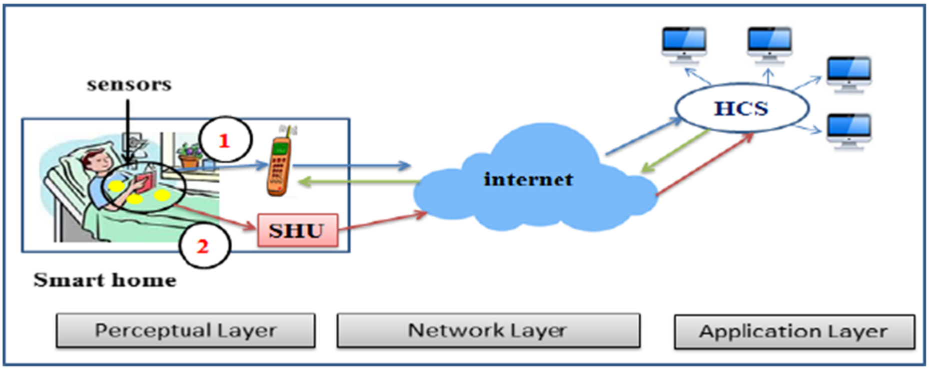


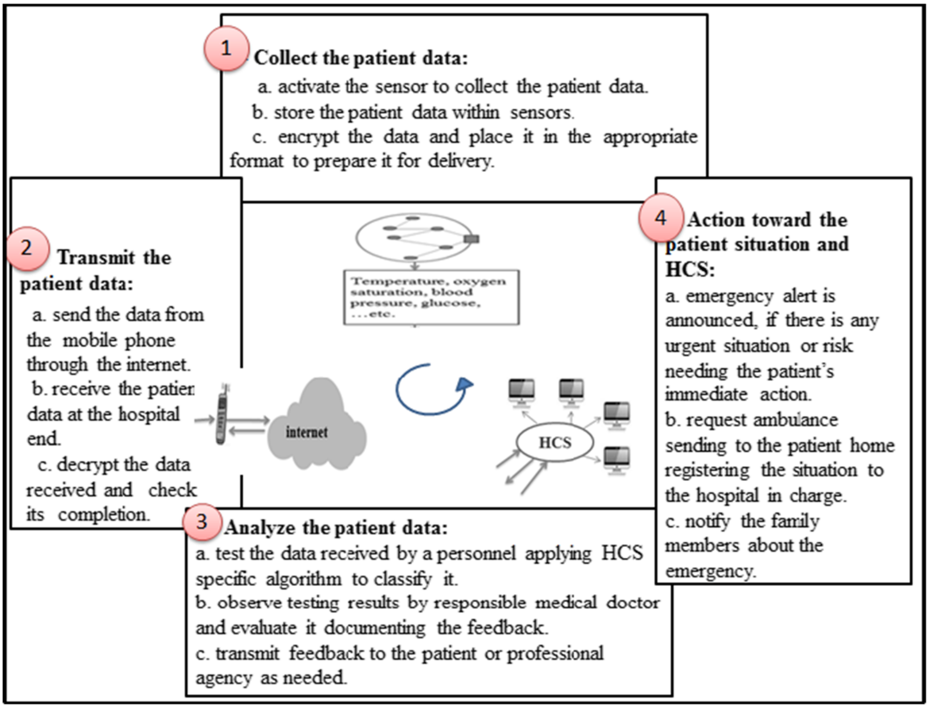
Fig. 2 Remote healthcare monitoring system (RHCMS)

vulnerabilities, threats and attacks that help to determine the security requirements to the chosen RHCMS scenario shown in Fig. 3.

Some of the prominent areas where these vulnerabilities occur are presented as follows:

A. *Collecting the patient's data by sensors*

Sensors are a core part in IoT. It is important to realize that communication links within the sensors are built around various technologies including the many wireless technologies so the security requirements are important to discuss. The data confidentiality must go on even in


Temperature, oxygen saturation, blood pressure, glucose, …etc.
internet
HCS

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.

2 **Transmit the patient data:**
a. send the data from the mobile phone through the internet.
b. receive the patier 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.

Fig. 3 Remote healthcare monitoring system scenario

case of a security breach related to the nodes or in transmitting between nodes. Any data leak may tell about the patient disease information. Thus, determining the limitations, threats and the appropriate security requirements are important to improve a trustworthy system. In fact, different scenarios may create different challenges in terms of security and performance that need to be solved by adaptive and the appropriate security methods.

In conclusion, data security, integrity and availability and authenticity are important in IoT. Security will prevent the private information to be intercepted by an attacker. The route of the message from source to destination contains a lot of intermediate entities that can be attacked to release the private information. This could be prevented if the message in these intermediate entities were encrypted [45]. Integrity will prevent the data alteration by attacker in either the transmission route or the storage nodes, which would lead to incorrect diagnosis results. Finally, availability of data to the hospital would support continuous treatment procedures for the patient.

B. *Transmitting patient's data through communication networks*

The data confidentiality is a must to prevent the data from being revealed if the interception happened in the communication session. Moreover, the data integrity is required to protect the data from modification when transmit from the sensors into the hospital. Data reliability is needed to guarantee that the collected data from sensors will receive no matter if there is a node or link failure. The data accuracy will guarantee the freshness of data and that the data is not replayed by an attacker.

C. Analyzing the data in hospital.

The medical records of patients must be surrounded by physical security that will restrict the access of the medical servers. At the same time using a weak physical security may permit and cause the system or records altered by unauthorized people. Data confidentiality would be used to limit the access of data monitoring at the system servers into authorized people. Also, data integrity is needed to protect the data from change by unauthorized persons. Data availability must be guaranteed to the hospital personnel even if any system failure occurs. The authentication is needed for two reasons. First, to authenticate the hospital personnel. Second, to ensure the data was received from the intended patient.

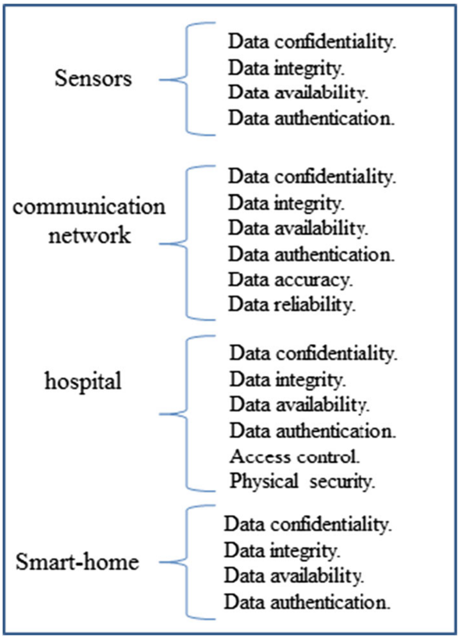
D. Using smart home environment.

The authentication process is indispensable and it's important to adapt the suitable authentication process to prevent the adversary from getting an access to the network operations. The functionality of the smart-home must be separated into public and private in order to prohibit the functionality exposure. Also, the availability is important to provide the direct connection between the smart-home and the hospital.

As discussed earlier, this paper focus is mainly involving the perceptual layer. We are discussing the data confidentiality of the medical data residing in sensors of the perceptual layer, as illustrated within interesting RHCMS classification overview of Fig. 4. Thus, the discussion would be on the nature of the medical sensors in IoT as it is believed to consider encrypting the data from the starting point-sensors, which is main factor to overcome lot of damages [37].

Usually, the performance of the application would be affected by involving cryptography [21]. In the case of medical applications, the response to the patient's condition is essential in

Fig. 4 the RHCMS needs from a security point of view



order to maintain patient's lives and health. Hence, the speed of encryption of the medical data and its transmission to the hospital is a very important point [22]. The longer the time of encryption the longer the treatment gets delayed, causing catastrophic outcomes [23]. Based on our case of study, there are four important speed parameters, namely: 1. Speed of data collection and encryption, 2. Speed of data transfer, 3. Speed of data analysis to determine the appropriate action, and 4. Speed of implementing the taken decision, as declared in Fig. 5.

In the proposed work, we target the speed of data encryption after collection. In fact, LWC uses fewer resources and saves time conserving the necessary security measures. Also, from a practical point of view, reducing the encryption time is essential to maintain the patient's life knowing his/her condition in a measurable acceptable time [42]. On the other hand, increasing

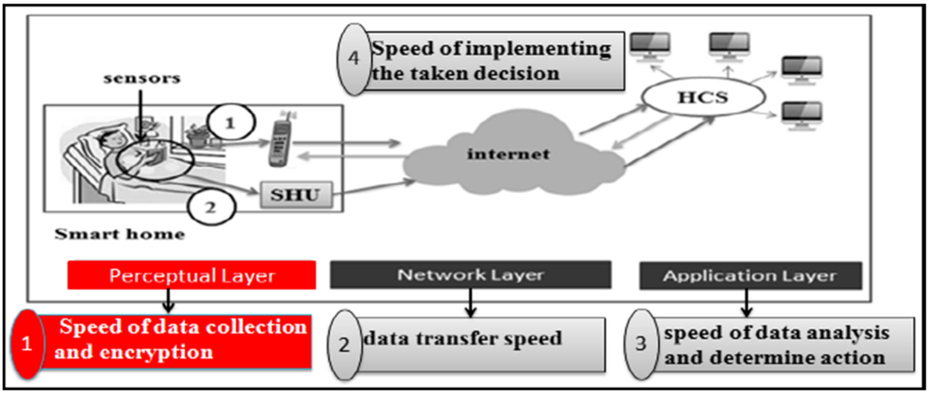

4
Speed of implementing the taken decision
sensors
1
2
SHU
Smart home
internet
HCS
Perceptual Layer
Network Layer
Application Layer
1
Speed of data collection and encryption
2
data transfer speed
3
speed of data analysis and determine action

Fig. 5 Remote healthcare system

the crypto-computation time may lead to disastrous and opposite results such as complications of the health status which maybe leading to the undesired worst case scenario as death of the patient.This has been our motivation and fuel to study the SIMON encryption algorithm in more depth, i.e. in a way that focuses on reducing execution time, speeding up the algorithm while maintaining the same security level of the algorithm.

3 Data confidentiality in IoT medical sensors: Requirements

Most of the practical IoT devices and applications are suffering low computing power, limited memory, restricted batteries and little operating frequency range [12]. Due to these constraints many important factors must be considered before adapting the security algorithms [5]. In short, the hardware limitations specifically involve the energy, computational constraints, and ROM-RAM memory restrictions [34]. Also, the software limitations include the embedded software boundaries as well as the dynamic security as discussed below.

3.1 Energy and computational constraints

Most of the IoT medical sensors have been developed based on portable battery driven requiring special attention [20]. Those devices are also using low power CPU which has a degraded clock rate [25] on top of this energy limitation. As a practical study, complex computations or expensive encryption cannot be realistic to involve in everyday handy low powered devices [30].

3.2 Memory limitations

Adapting a strong and long encryption algorithm may lead to a negative impact on the battery consumption or the implementation performance i.e. ROM and RAM sizes. In general, the medical sensors suffer from a limitation of memory sizes. In sensors RAM as well as its small flash memory can’t be compared with other traditional digital systems, i.e. laptop, iPad, mobile phones, etc. [29]. For these reasons, security schemes on IoT medical devices must be memory efficient. The traditional security algorithms are proven not preferable and may not be taking the memory efficiency in mind [12].

3.3 Embedded software restrictions

The medical IoT operating system, which is embedded in applicable devices, is having small network protocol stacks [30]. Beside that the security model must be thin. It should also be robust considering fault tolerance, at the same time, to reduce error rate in human health treatments [49].

3.4 Dynamic security patch

The remote reprogramming for the IoT devices is currently believed to be impossible considering the available operating systems [29]. In fact, the protocol stack may not be able to receive and modify the code or the library while in operation, causing systems update to be a real challenge [49], i.e., to be addressed besides all other restrictions.

3.5 Mobility

Nowadays, supporting the mobility feature for the healthcare services is a must. But, pervasive mobility and security at the same time is a challenge. That belongs to the fact that different networks may need different security requirements [14].

The focus of the work carried out in this paper work is to improve the SIMON algorithm with respect to three primary metrics: execution time, memory occupation and speed. In our medical case study, the point of preference for performance is security, but it must consider the practical speed of the implementation because the task is to ensure patient safety and survival. However, increasing security may increase the implementation time, thus putting the patient's life at risk, which is the realistic performance-efficiency base for our software implementation of the cipher IoT medical system.

4 Studied block ciphers

The shift of focus from the PC or desktop computing system into the resource constrained devices has led this investigation for research in LWC. Adapting normal, i.e. heavy-weight, encryption algorithm increases the overhead making the process completely unpractical. However, adapting lightweight crypto algorithms will achieve acceptable balance between the security and practical overhead [1]. Despite the large number of lightweight algorithms currently present or proposed they are lack of flexibility. We have used SIMON for this study because of various advantages associated with it: flexibility, simplicity, efficiency, besides security and that kind of balance is hard to get in many LWC algorithms [7]. In contrast, a large number of researches have continued for years targeting the AES algorithm in order to reshape it for the lightweight applications. From the other side, there are a number of researches that have found that SIMON algorithm may be an alternative for the AES algorithm under the circumstances of resource constrained platforms, as SIMON offers more options of blocks and key sizes [19]. In fact, having this feature may allow a tradeoff between security, performance and risks. Choosing a 32 bit block or 128 bit block depending on the application needs, where reducing the security level can maintain energy and save computation time. As a result, improving the node's life time.

Furthermore, these targeted encryption algorithms have different architecture base where as SIMON is dependent on the Feistel structure; which uses a simple logic and arithmetic operations such XOR, AND, and shifting. On the other hand, AES depends on the Substitution Permutation Network (SPN) structure with S-box based design where each bit in the intermediate cipher-text is changed for each round. And it's to be mentioned that the software platform performance is out of the scope of this work, considering it is not important to be discussed at this stage. Thus, comparing the focus algorithms is to check how fast, efficient and simple they are in order to conclude the optimal algorithm for the Remote Healthcare Monitoring System.

4.1 AES

The Advanced Encryption Standard (AES) is one of the most famous algorithms in the cryptographic field. It is involved in a lot of research as it is, as well as tuned to make it more practical or lightweight. Also, it can allow flexibility, i.e. it can integrate the SPN structure of the AES with the ARX based structure instead of the S-box based structure, to gain more simplicity design for future work. The size of the cipher block is 128 bits allowing three optional key lengths. In fact, the key length determines the number of rounds in the algorithm. It is known that128 key length AES needs 10 rounds, 192 key length requires 12 rounds, and 256 key length necessitates 14 rounds [10]. Furthermore, according to [11] AES has new block and key size called 512-AES which requires 22 rounds. In this work, the 128 bits key length AES model is used and is presented by 4*4 matrix. AES requests four basic operations that must be applied to the cipher state, namely Sub Bytes, Shift Rows, mix columns, and add round key. These four operation functions are repeated ten times in ten rounds, based on key length, except the Mix column that would not be executed in round ten. The reader is directed to reference [10] for more in depth description.

4.2 SPECK

SPECK was proposed by Ray Beaulieu [6], as family of ciphers designed with flexibility allowing some different key and block sizes. The cipher was developed to achieve higher performance especially optimized for execution on microcontrollers. SPECK is similar to SIMON, i.e. it is also Feistel structure cipher. It runs with two branches which develop every round by using crucial shifts in both directions, modular addition and bitwise XOR [6].

The Round function of SPECK depends on the Feistel structure as illustrated in Equation (1).

$$Rk(x,y) = \left[(S^{-a}x+y)\oplus k, S^{-b}y\oplus(S^{-a}x+y)\oplus k\right] \tag{1}$$

In this section, we discussed SPECK algorithm, because the optimization on SIMON was based on SPECK. Also, SIMON and SPECK are very related. Recall that SIMON is originally built for hardware utilization where SPECK built for the software.

4.3 SIMON

Original SIMON algorithm is considered as a Feistel block cipher which is designed by National Security Agency (NSA). Its cipher operations consist of simple round functions that uses bitwise AND, bitwise OR and left circular shifts. It has been developed to improve the performance of hardware and at the same time to achieve good computation crypto results [36].

The round function in SIMON is depends on two stage of the Feistel map just as illustrated in Equation (2):

$$Rk(x,y) = (y\oplus f(x)\oplus k, x) \tag{2}$$

Where k is the round key and

$$f(x) = \left(Sx\&S^8x\right)\oplus S^2x \tag{3}$$

SIMON basic operations are:

- Bitwise AND: Bitwise AND between two random bits of n bits words which can be described as: (x &y)
- Left and circular shifts: (x < < y) is the cyclic shift of x to the left by y bits, where as (x> > y) is the cyclic shift of x to the right by y bits.
- Bitwise XOR: is the exclusive-or (XOR) which is represented like $x \oplus y$.

In general, the speed performance and security strength running the crypto-algorithms are considered challenging conflicting goals [28]. To focus our discussion, SIMON is typically not preferred when it comes to software performance in comparison with SPECK [1] because SIMON requires more rounds to execute, as its main dedication is for hardware rather than software. In this regards, we are trying to benefit improving SIMON benefitting from features of another algorithm known as SPECK giving us room for improvement and speeding up, as clarified in the next section.

5 Proposed SIMON LWC improvement

As mentioned earlier, the light weight crypto algorithms considered in this work are AES, and SIMON. This section discusses improvements to original features of SIMON, benefiting from the understanding of another algorithm, namely SPECK. Recall the scope prospective to enhance SIMON speed.

To benefit this work, consider that both SIMON and SPECK lightweight block ciphers are designs relayed on the ARX philosophy [6].This means that the ARX philosophy is based on avoiding the use of the look-up tables related to the S-box based designs. Also, it benefits from the decrease of the number of operations that are used to perform the encryption. Thus, helps to improve the software performance providing its normality of the code to be acceptable for small medical devices as IoT inspiration [41]; where both SIMON and SPECK belongs to the same family of lightweight block ciphers [8]. Both block ciphers contain twenty variants which supports range of key sizes. The key sizes can be from 64-bits to 256-bits. Each block size can vary, starting from 32-bit and ending at 128-bit block size.

As declared earlier, SPECK algorithm is considered as pure ARX design and can be described as an algorithm that combines two Feistel structures [7]. The round function of SPECK, as shown earlier in Equation 1 and Fig. 6, noting a is 8 and b is 3 for all the block sizes except for the 32-bit size holding special condition of a as 7 and b as 2.

The value parameters of $Rk(x,y)$ is affected by a and b, if $n = 16$ then a is 7 and b is 2, otherwise a is 8 and b is 3. Note interestingly that within describing the SPECK round functions, the SPECK$2n/mn$ can be described as a SPECK with the $2*n$ bit block size and $m*n$ bit as the key size. For example, SPECK32/64, means the word size is 16 bits which is called (n), and the 32 bit is the $2*n$; while the number of words is 4 which is noted as m. As a result, the total key is $m*n$ which equals to 64. For more details on SPECK, the reader is referred to reference [7].

From the other side, any algorithm needs to be scheduled for the key where a master key can generate a group of keys for each round as illustrated in Fig. 7.

The key schedule of SPECK depends on the idea of re-using the SPECK round function. As $m = (2, 3 \text{ and } 4)$, representing the number of words of the master key schedule such that $(l_{m-2}, \ldots.., l_0, k_0)$, all the round key n are extracted as declared in the code shown in Fig. 8. In other words, The key schedule of SPECK use its own round function in order to generate the round keys K_i. Note that the round keys K_i are generated to be used in the SPECK round function. The round keys can be written such that $(l_{i+m-1}, \ldots., l_i, k_i)$ where m *is in* $(2,3, 4)$, representing the number of words of the master key schedule.

The counter i in SPECK is issued to eliminate the slide properties. However, SIMON's key schedule is independent of its round function. Assume m is the number of key words, the key rounds functions can be written as clarified in the three following equations.

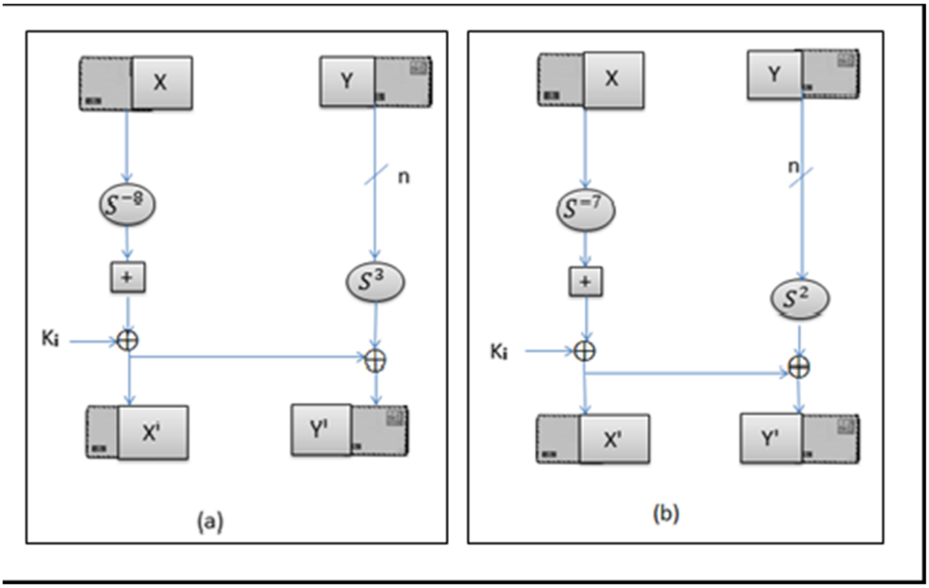


Fig. 6 **a** SPECK round function for 32 block size **b** SPECK round function the all the block sizes of SPECK except 32

$$K_{i+2} = K_i \oplus \left(I \oplus S^{-1}\right) S^{-3} K_{i+1} \oplus C \oplus \left(Z_{j\ i}\right) \tag{4}$$

illustrated in Fig. 9 section a.

$$K_{i+3} = K_i \oplus \left(I \oplus S^{-1}\right) S^{-3} K_{i+2} \oplus C \oplus \left(Z_{j\ i}\right) \tag{5}$$

illustrated in Fig. 9 section b.

$$K_{i+4} = K_i \oplus \left(I \oplus S^{-1}\right) S^{-3} K_{i+3} \oplus C \oplus \left(Z_{j\ i}\right) \tag{6}$$

illustrated in Fig. 9 section c.

As illustrated in Figure 9, Round key k_i is derived from the master key (K). Where the master key K is divided into different lengths of words which is (2, 3 or 4) as illustrated in Table 1. The $C \oplus (Z_{ji})$ are used to eliminate the slide properties and the circular shift

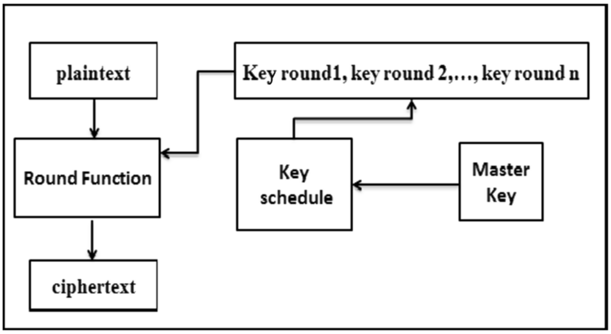
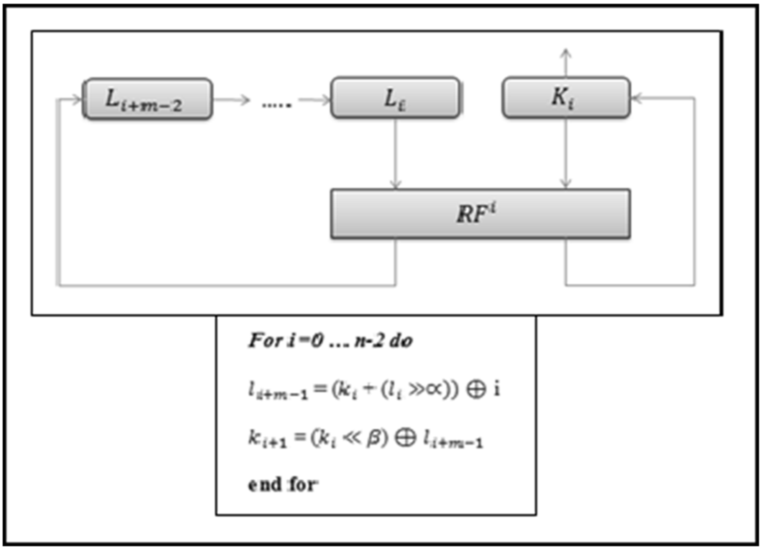


Fig. 7 Composition of the algorithm



```
For i=0 ... n-2 do
    $l_{i+m-1} = (k_i + (l_i \gg \propto)) \oplus i$
    $k_{i+1} = (k_i \ll \beta) \oplus l_{i+m-1}$
end for
```

Fig. 8 Key schedule of SPECK where RF is the round function

similarities at the same time. Thus, this explains the benefit of adding it to the key expansion process. The following Table describes the SIMON's basic parameters.

The performance and security are considered as a competing goals in regard to cryptography [28]. SPECK overcomes SIMON, when it comes to software performance [1].This is because SIMON requires more rounds to execute due to its use of weaker non-linear function of bitwise AND operation, which is found preferred for hardware more than software. On the other hand, SPECK gets its non-linearity from the complex modular addition operation that is preferred for software. It is to be noted that both SPECK and SIMON operations work functionally acceptable in hardware and software.

Usually the software implementation of encryption algorithm is concerned with bit permutation. The bit permutation of SPECK and SIMON are both depending on the circular shifts [8]. In this research work, we are targeting the software performance of the SIMON algorithm since it is found appropriate for enhancement. Specifically, we try to reduce the execution time and the speed cost of the algorithm means reach to a high performance while preserving an optimal trade-off between the security and the speed. Thus, we are highlighting the design decisions that may affect the SIMON's software performance.

In general, the execution time is the amount of processing time that is required to perform the security mechanism. In order to improve the software performance, we need to take some points into consideration like word size, byte ordering. Also, the point that can't be ignored is the quality of the compiler, which may affect the performance in software simulations, i.e. implementing the encryption algorithm in C language is normally slower than implementing it in Assembly. The reason is related to the existence of processor instructor such as operation: 'rotate'. Note that our work is dedicated as proof of concept for academic interest, all the simulation codes were written in C programming language. Each implementation based on assembly would be expressed as a processor dependent and thus will miss the probability feature. Although, there is no doubt about the importance of implementation in Assembly language but the nature of IoT programming is mainly flexible having no specific hardware

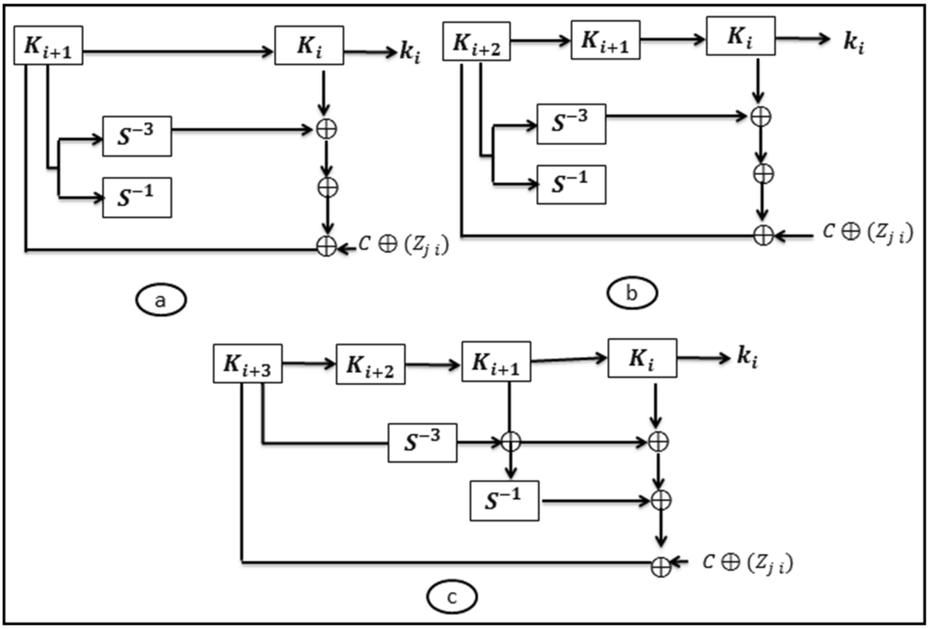


Fig. 9 SIMON's key schedule

platform dominated. This has made the IoT devices make C a valid priority for acceptance. In other words, IoT contains a huge number of distinctive devices, different processors and different operating systems. Thus, relying on assembly is unpractical causing a lot of codes variations to be written, tested, debugged and maintained separately and so a lot of errors will appear to be debugged.

Indeed, in order to improve the speed performance of SIMON algorithm, it was necessary to study the optimization in the number of rounds, the rounds is function and the key extension function. Reducing the number of rounds considered as the easiest and most effective solution to reduce the execution time and thus improve the speed and energy. But, at the same time, it can directly impact the security of the algorithm, where the adequate number of rotation will help to protect against several traditional attacks [45]. It was attempted to avoid changing the number of rounds to maintain the same level of security of the original SIMON, however, some attacks affected algorithm successfully due to the reduction of number of rounds. Also, in regard to the round function process, we found that it was necessary to study the basic operations that formed the round function of the algorithm to determine the optimal changes. In fact, changing the logical or arithmetic operations, re-change the order of execution, or

Table 1 SIMON's basic parameter

Block size (2n)	Key size (mn)	Rounds	Constant sequence
16*2 = 32	16*4 = 64	32	Z_0
42*2 = 48	42*3 = 72,42*4 = 96	36,36	Z_0,Z_1
32*2 = 64	32*3 = 96,32*4 = 128	42,44	Z_2,Z_3
48*2 = 96	48*2 = 96,48*3 = 144	52,54	Z_2,Z_3
64*2 = 128	64*2 = 128, 64*3 = 192,64*4 = 256	68,69,72	Z_2,Z_3, Z_4

change the number of rotation, without taking into account the software performance lead to a negative impact in the software implementation and complicated our situation.

To clarify our modification, consider the SIMON digital round operations. SIMON's round function depends on some basic logical operations namely: AND bitwise operation, XOR operation, and three shift left circular operations (shift by one, shift by eight and shift by two). The problem is that the shift operation may be easy and cheap for hardware than software, which is really expensive highly dependent on the targeted platform. The number of shifts affect the speed so accuracy in the optimization is needed to determine the appropriate efficient shift numbers. As an illustration, the shift 32-bit on 8-bit microcontroller or 16-bit microcontroller will slower the implementation of the algorithm. Indeed, in SIMON's software performance there is a penalty due to the execution of all shift operations on a single word. Unlike SPECK where there is some balance in its round function design. Unfortunately, we could not be able to remove this SIMON penalty because it maintains the encrypt and decrypt symmetry [7]. But in return, we could reduce its impact by changing the shift numbers from ({1,8}, 2) to ({0,8}, 2) so the first rotation is removed. Where this rotation by 8 is almost considered free on 8-bit microcontroller and inexpensive in 16-bit or 32-bit microcontroller, as noticed in SPECK. Also, the rotation by 2 is inexpensive too, since the digital number is multiplied by power of 2's, i.e. 8, 16 or 32, which are just shifting of binary bits.

Our new version of the algorithm will follow the standard Feistel structure. The plaintext is splitting into two words x_0, y_0 where zero refers to the first number of round. The x_0, y_0 will enter the round function procedure for a specific number of rounds until the last round number appeared, then $x_{round\ number}$, $y_{round\ number}$ are extracted and finally concatenated in a cipher-text format, as illustrated in Figure 10.

In addition, our case of study is using SIMON 64/96 reducing the number of iterations of the round function loop into 21 times to handle the 42 rounds of SIMON 64/96 by partially unrolling 2 rounds of the code. Thus, the speed and the execution time will be optimized and the space-time trade off would appear too, but in unnoticeable manner. Furthermore**,** we can simplify the SIMON key schedule process by using the idea of re-using our modified round function to generate the key schedule like SPECK did. In other words, the optimized SIMON Key schedule use its own round function and the C ⊕ (Zji) to generate the round keys. Indeed, The master key will be divided into different number of words depending on the number of registers similar to the case of the original SIMON which shown in Figure 11.

Doing that change in the key schedule will not lead to weakness so far, according to a similar idea presented in [47]. Also, making this choice will keep the code size smaller using optimized ROM and nothing from RAM, which is an extra benefit. This optimized key schedule of SIMON is illustrated in Figure 9.

6 Comparison and analysis

This section presents the output of the simulation that has been carried out using Cooja simulator with Contiki Operating System. Cooja emulation tool is considered as most used one for the sensors/IoT domain. IoT normally uses multiple mode of communication networks besides it supports many platforms and flexibility options [43]. As a matter of fact, the development of non-intrusive sensing, open source, wireless and wire-able solutions are still considered as an open issue [35]. From the other side, one of the most important challenges facing the medical applications is the conditional time data flow. Considering the IoT concept,

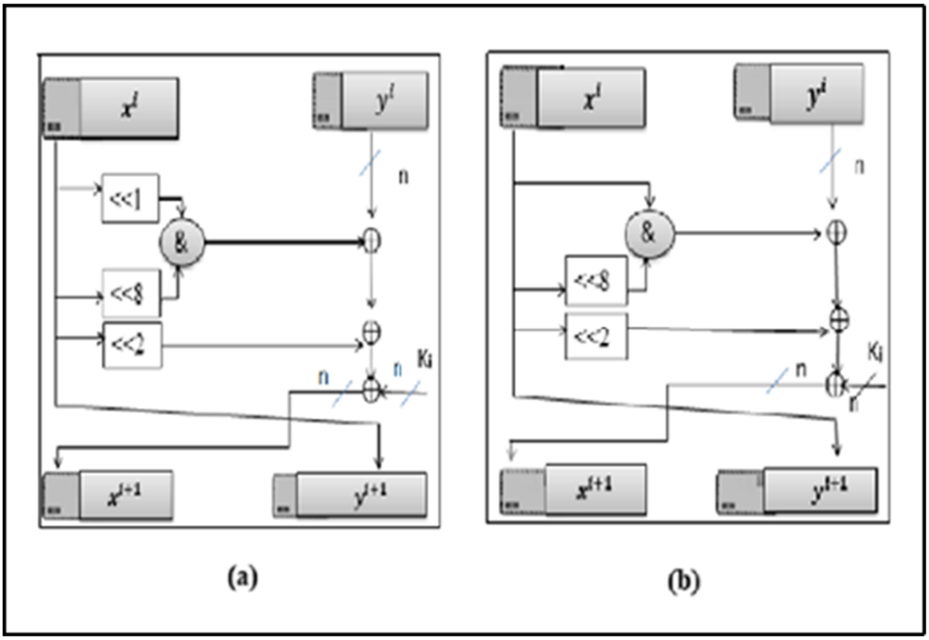


Fig. 10 **a** Original SIMON round function, and **b** the optimized round function

where processing a huge amount of data is required to allow a large communication between different devices. In this case, the traditional simulators do not fit. Sensor nodes need an operating system, routing protocol and simulation for the purpose of operating in the appropriate way. The problem that is encountered in the sensor networks is the absence of the hardware standardization architecture. Further, the normal simulation tools don't pay attention for the need of on-node processing algorithm like Cooja does. It is pertinent to mention that many state of art medical monitoring system shave been simulated using Cooja [9, 32, 44]. This work simulation utilizes T-mote sky platform, defined as MSP430 based board involving IEEE802.15.4 wireless module. The radio chip CC2420 is used to enable less power, with wakeup-fast from sleep feature. The platform affords a reliable wireless interesting communication that run smoothly for all simulation. The basic parameters for simulation have been depicted in Table 2. The implementation board contains MSP430 F1611 microcontroller that involves 48 KB ROM and 10 KB RAM. This T-mote sky hardware requires similar to real life portable battery of type AA solid batteries, with the additional feature to be attached to the computer to run using the USB port.

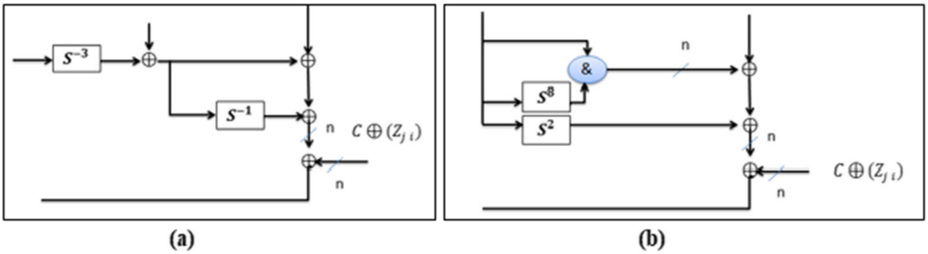


Fig. 11 **a** Original key schedule of SIMON **b** Optimized key schedule of SIMON

Table 2 Basic parameters for simulation

Parameter value	Value
The simulated mote platform	T-mote Sky
Operating system	Contiki
CPU	MSP430
Contiki version	3.0
Computer OS	UBUNTU 14.04 LTS
Encryption Algorithms	(AES,SPECK,SIMON)
Compiling GCC	MSP430-gcc 4.7.0
Node transmission range	50 m, 50 m

Two t-mote sensors are included where the first sensor encrypts the medical data and send it to the second sensor while considering an area of 50*50 mm^2.

In order to measure the CPU cycles, we used the MSP-cycle-watcher built as a quality checking tool to count the CPU cycles of each encryption code. The measurement of the number of cycles within the code of the encryption part is extracted by subtracting it out of the complete cycles of the normal entire code, i.e. involving setup and post computation overhead. It is assumed that comparing the exact encryption cycles is leading to fair comparison study of power consumption. Also, this step (execution encryption cycles) have been performed precisely by adding a check point to determine the start and end point of counting the encryption cycles. In this work, we measure the ROM use by a “size A” command present in the used MSP430-gcc compiler. This “size A” measure calculates the memory consumption automatically from the compiled code file. Typically, the ROM consumption of the program is the considered collection of text and data bytes adopted to determine the ROM consumption used for the dedicated encryption part only. In fact, the RAM includes data consumption and stack consumption. The data consumption is computed using the “size A” command and the implementation information file; but for the stack usage during the execution, we used the MSP stack watcher, i.e. to check the stack usage.

This study has been performed to compare and evaluate the optimized version of SIMON with the performance of the original encryption algorithms, i.e. SIMON, and AES all on software platform. The evaluation focus is on two aspects namely: execution time and the memory usage. The comparison results of execution time are shown in Figure 12.

Note that SIMON is flexible, offers a range of block and key sizes and as mentioned earlier that is the reason that makes SIMON an attractive solution for IoT and for different multimedia applications. However, AES supports only two block sizes 128 bits and 512 bits. Obviously, the second option is too big to consider for the lightweight algorithms. Additionally, AES-128 guarantees a sufficient data protection for a small IoT/healthcare devices. The comparison results of execution time and, ROM consumption are shown in Table 3.

Figure 12 shows the comparison of speed with regard to the execution time of the original SIMON, optimized SIMON with respect to all the block size versions of the algorithm and the AES (128/128). It shows that AES consumes clock cycles less than SIMON with block size 64, 96 and 128 bits. It indicates that the presented optimization allows the SIMON’s algorithm, which originally well suited to hardware implementation more than software, to overcome AES. The original SIMON speed lower than AES is justifying its normal avoidance to be chosen which is resolved in this work, i.e. software implementation of LWC algorithms.

In regard to the optimized SIMON, to quantify the execution time comparison, the difference percentages were calculated. The values presented interesting feedbacks on how

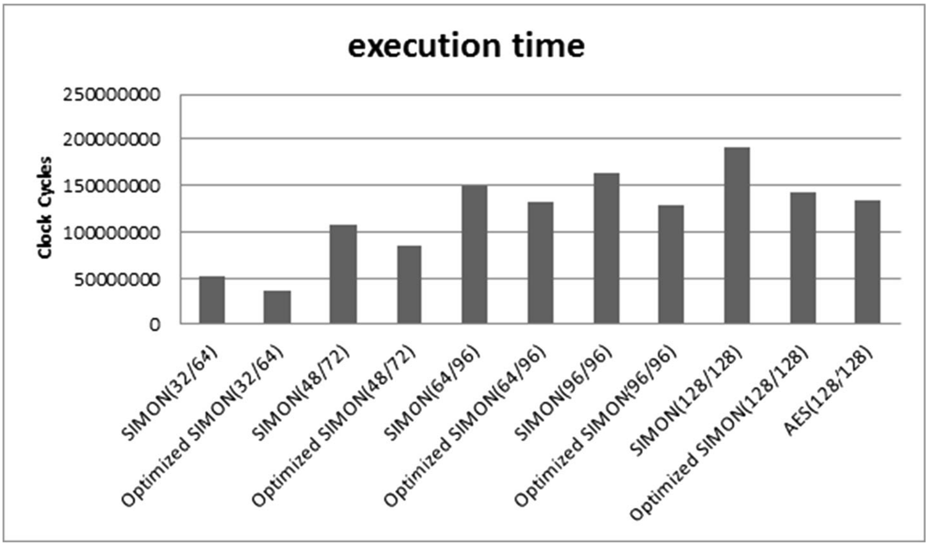


Fig. 12 Comparison of execution time for the original SIMON algorithms, optimized SIMON algorithms and AES

much the optimized SIMON had been enhanced in comparison with the original SIMON; the two versions are presented in Figure 7. The optimized SIMON (32/64) and the optimized SIMON(48/72) consumes clock cycles less than the original SIMON(32/64) and the original SIMON (48/72) by about 20%. Also, the number of execution cycles of optimized SIMON (64/96) is less than SIMON(64/96) by 13%.

Moreover, the execution cycles of the SIMON (96/96) is enhanced by22% where SIMON (128/128) is enhanced by 26%. As a result, the optimized versions of SIMON with 32, 48, 64 and 96 do overcome the speed of the AES algorithm.

The memory consumption for the AES (128/128), original SIMON and the optimized SIMON with respect to the different block sizes has been evaluated by measuring the number of bytes in both RAM and ROM, and the results have been shown in Figure 13 and 14.

SIMON and the optimized SIMON achieved higher performances than AES. Note that some blocks of the optimized SIMON shows different usage of ROM and that because the

Table 3 Execution time and memory comparison results

Algorithm	Execution-time		ROM	
	(Clock Cycles)	Time Gain %	(bytes)	ROM Gain %
SIMON(32/64)	53,467,974	41%	1510	2.7%
Optimized SIMON(32/64)	37,922,749		1470	
SIMON(48/72)	108,901,089	26%	1752	2.3%
Optimized SIMON(48/72)	86,293,887		1712	
SIMON(64/96)	150,539,925	14%	1682	−0.2%
Optimized SIMON(64/96)	132,348,035		1686	
SIMON(96/96)	164,432,890	28%	2086	−6.3%
Optimized SIMON(96/96)	128,641,151		2226	
SIMON(128/128)	191,322,912	34%	1990	−5.6%
Optimized SIMON(128/128)	142,396,108		2108	

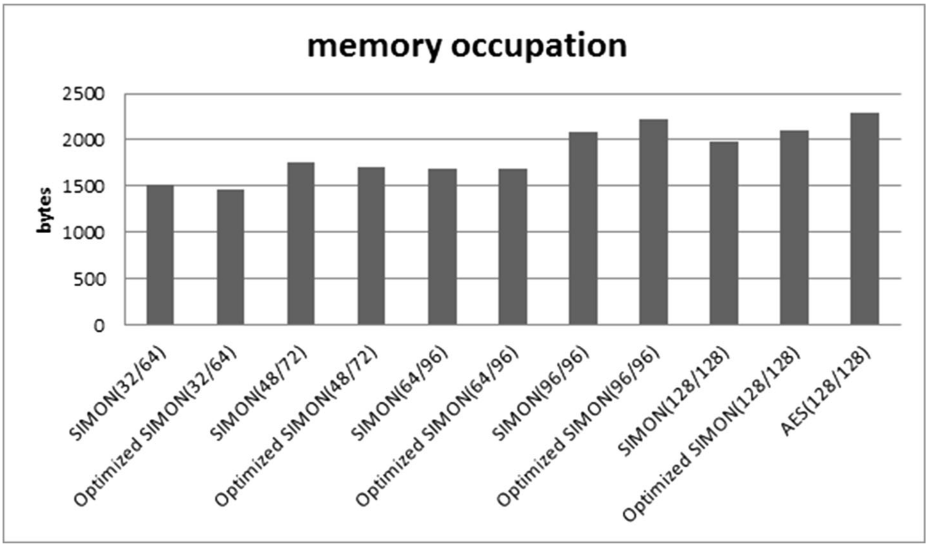


Fig. 13 Comparison of ROM memory occupation for algorithms

manual optimization of each code separately. In fact the improved SIMON, with block 32, 48 and 96, are using slightly less ROM than the original SIMON. This is due to the reuse of the same round function on the key schedule that clearly emerged for the small block size versions as well as the unrolling of the round function is reducing the number of jumps inside the code by reducing the calculation of the indices number inside the loop. However, the optimized SIMON with block 64 appeared to have more four bytes and that because using both the functioning in lining with the unrolling process of the round function leads to a little increase in the ROM. The optimized SIMON 128 appeared to use slightly more ROM. This can be a normal acceptable negligible cost considering the improvement made to the algorithm and specifically due to the memory moves. In other words, copying the data from location to another by using the (memcpy) function.

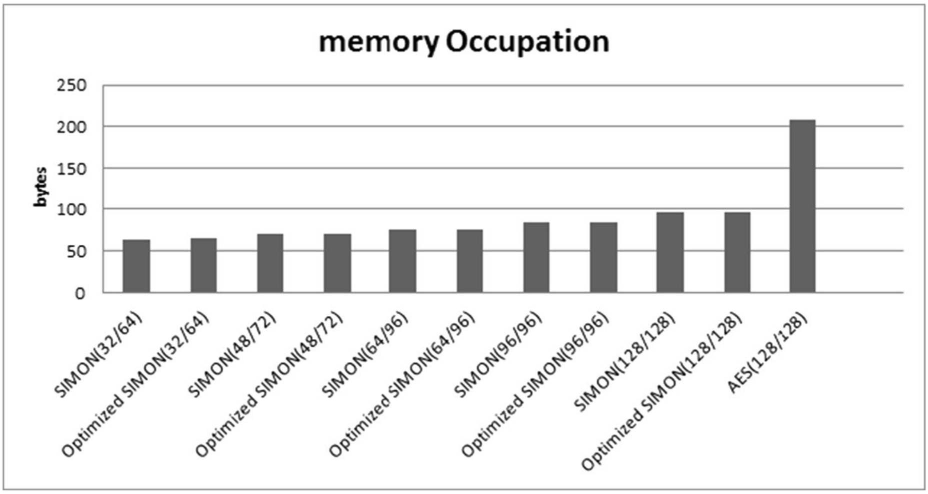


Fig. 14 RAM occupation of SIMON, the optimized version of SIMON and AES

Finally, both original SIMON and the optimized SIMON are at the same level when it comes to RAM use, where the number of measured bytes are ranged from 64 to 96 bytes. In other words, the requirements of RAM consumption between original SIMON and the optimized SIMON are very small as well as very similar and that belongs to the fact that the ARX algorithms tend to avoid the adaption of the tables in their design.

7 Security analysis

In general, the lightweight cryptography may suffer from four type of attacks which involve eavesdropping, saturation, differential attacks and masquerading. In fact, the simplicity of the design of SIMON made it attractive target for lots of cryptanalysis goals. However, to the best of our knowledge, all the published attacks were related to the reduced number of rounds features. Also, the sort of analysis found is to specify the number of rounds that are susceptible to theoretical attack, making security measures by the number of rounds needed to be attacked. In fact, publications emphasise that no attack exceeds 70% of any version of SIMON. Also, the SIMON's best attack related specifically to SIMON 128/128 with 48 out of 69 rounds [6]. As a matter of fact, increasing the number of rounds for both will increase the strength, but unfortunately it affects the performance. In regard to AES, the side channel attacks can be considered as the highest security threat to AES. Actually, this type of attack is not related to specific vulnerabilities, i.e. of the cipher algorithm, instead they depend on exploiting the lack of information related to physical implementation of the algorithm. For instance, the timing attack can reveal the secret key from the number of consumed clock cycles through the encryption process of the targeted device. Additionally, the brute force attack depends on the size of the key. In general, if the key length is equal to *n* bits, then the size of key space is *2* to the power of *n*. In our case of AES 128/128, needs 2^{128} alternative keys to be tried, which is equal to the delay of 10^{38} *3.4. In other words, if the computer calculates 10^{6} decryption/ms, thus the worst case will demand more than 5.4*10^{18} years to extract the correct key to break the cipher, as security of AES is reported in 2014. In general, increasing the key length increases the number of rounds, which increase the complexity of the algorithm to resist attacks.

From the other side, the security analysis of the optimized version of SIMON. The analysis is carried out based on the similarity between the optimized version of SIMON, original SIMON and SPECK. As mentioned earlier, security measures are related to the number of rounds that have been attacked. Thus, due to the sufficient number of rounds, we believe that the encryption strength of the optimized version of SIMON is comparable to the original SIMON. In addition, choosing the shift parameters of the algorithm with regard to the linear and differential characteristics and having at least one byte length rotation are desirable for better performance [31].

Considering differential linear attacks, as has been noted the behavior of the optimized SIMON is related to the original SIMON so it makes sense to use the best known differential and linear attacks of the original SIMON. Thus, optimized SIMON 32/64 can be attacked on 19 rounds in time complexity2^{34}. As well as, the optimized SIMON 128/128 can be attacked by 49 rounds out of 68 and with time complexity near to 2^{128} which is equal to 70%, and denoted as the highest percentage that could happen to that algorithm according to [8]. Also, for more details about the cryptanalytic results please refer to [3].

Additionally with respect to the impossible differential attacks, SIMON by default has a large number of rounds compared to SPECK. This type of attack was more successful with the reduced rounds version of the original SIMON, so optimizing SIMON with reducing number of rounds will result in suspicion for such attack. Moreover, to the best of our knowledge SIMON is not a candidate for the Meet in the Middle Attack (MIMT). We believe our optimized version of SIMON is more protective against this attack. This is because of re-using the round function in the key schedule allowed many key bits from the master key stream to works quickly in the round function, based on a similar idea presented by [47]. Also, the equation solving attacks/ algebraic are considered hard to be classified for the original SIMON and SPECK. Indeed, this type of attack can work successfully to break the small number of rounds according to [48], where authors were able to break around 10 rounds of the ciphers. As a result, we think the same works here, but it may reach to 13 rounds no matter of the block size with respect to a similar scenario shown in [47].

Furthermore, in the case of related key differential attacks, SPECK and SIMON have been criticized from the time of their inception. Until now according to the literature there was no issue about key-related attacks. Thus, the optimized SIMON which follows the scheduling idea of SPECK has good cryptographic properties. As a result, it can contribute to maintain the required balance between security and the healthcare scenarios. For instance, the patient's data can be encrypted in measurable time, then sent to the hospital to be analyzed and evaluated. This will help the hospital to determine if there is an emergency or a need to send feedback.

Using the optimized SIMON (32/64) causes trade-offs between security and normal application needs. Usually, it is not an acceptable solution due to the small size of block and key sizes. However, optimized SIMON (48/72) gives the best balance between memory usage and the responsiveness of health system. Indeed, the speed of execution time for the encryption algorithm puts best priority on patient's life. Where, optimized SIMON (96/96) is considered faster than optimized SIMON (64/96) but there is trade-off between speed and the memory use which is still an acceptable solution. On the other hand, the optimized SIMON 128/128 gives the worst performance making it as unpleasant solution to be used.

8 Conclusion

This paper presented a speedup version of SIMON algorithm with respect to software implementations. The improvement benefitted from interesting features found in a different algorithm named SPECK to modify SIMON original work gaining attractive results. The modification preserved the main structure of SIMON but performed slight reordering within the procedure. The modification has resulted in gain pf interesting performance measures, making SIMON with block sizes 32, 48, 64, 96 bits win among AES 128/128. The modified SIMON with block sizes 32, 48, 64, 96 bits showed interesting speedup compared to the original SIMON, where some of them were originally found slower than AES. The optimized SIMON with all the block versions shows an enhancement percentage range from 20% to 26% except the block size 64 which show an enhancement percentage close to 13%. The implementation study examined both sides of running the algorithm on software, i.e. execution time as well as memory consumption.

The research contribution enhanced the encryption performance of the modified SIMON while preserving an optimal tradeoff between security, performance and memory cost. The paper showed its work results presenting a comparison with the original SIMON with respect to all the block size variants and AES in regard to the speed (execution time measured by the clock cycles) as well as ROM and RAM memory consumption to give the performance overall view. Given the results obtained, the proposed low weight cryptographic algorithm, could find a wide range of applications in IoT based setups. The future research to follow this work is to consider SPECK algorithm, which was used for modification to original SIMON, for comparison and further improvement study hoping interesting attractive outcomes.

Acknowledgements The authors acknowledge all support provided by Umm Al-Qura University (UQU) for encouraging this research work. Special thanks goes to the cooperation between the two UQU departments via Prof. Adnan Gutub from Computer Engineering and Dr Manal Al Ghamdi from Computer Sciences for motivating this research as well as building international collaboration with Dr Shabir Parah from the department of Electronics and IT, University of Kashmir, Srinagar, all working in collaboration on this wonderful research contributions.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

1. AlAssaf N, AlKazemi B, Gutub A (2017) Applicable Light-Weight Cryptography to Secure Medical Data in IoT Systems. Journal of Research in Engineering and Applied Sciences (JREAS) 2(2):50–58
2. Alharthi N, Gutub A (2017) Data Visualization to Explore Improving Decision-Making within Hajj Services. Scientific Modelling and Research 2(1):9–18. https://doi.org/10.20448/808.2.1.9.18
3. Alizadeh J, AlKhzaimi H, Aref MR, Bagheri N, Gauravaram P, Kumar A, Lauridsen M, Sanadhya S (2014) Cryptanalysis of SIMON Variants with Connections. International Workshop on Radio Frequency Identification: Security and Privacy Issues (RFIDSec), Lecture Notes in Computer Science Vol. 8651, pp. 90–107, Springer
4. Almazrooie M, Samsudin A, Gutub A, Salleh MS, Omar MA, Hassan SA (2018) Integrity verification for digital Holy Quran verses using cryptographic hash function and compression. Journal of King Saud University - Computer and Information Sciences, Published by Elsevier
5. Aly S, AlGhamdi T, Salim M, Amin H, Gutub A (2014) Information Gathering Schemes for Collaborative Sensor Devices. Procedia Computer Science, Elsevier 32:1141–1146
6. Beaulieu R, Shors D, Smith J, Clark ST, Weeks B, Wingers L (2015) The SIMON and SPECK lightweight block ciphers. Proceedings of the 52nd Annual Design Automation Conference (DAC '15). ACM, New York
7. Beaulieu R, Shors D, Smith J, Clark ST, Weeks B, Wingers L (2017) Notes on the design and analysis of SIMON and SPECK. IACR Cryptology ePrint Archive 560
8. Beaulieu R, Shors D, Smith J, Treatman-Clark S, Weeks B, Wingers L (2015) S and S: Block Ciphers for the Internet of Things. IACR Cryptography ePrint Archive
9. Ciabattoni L, Freddi A, Longhi S, Monteriù A, Pepa L, Prist M (2016) An open and modular hardware node for wireless sensor and body area networks. Journal of Sensors
10. Daemen J, Rijmen V (1999) AES proposal: Rijndael
11. Dandekar AK, Pradhan S, Ghormade S (2016) Design of AES-512 algorithm for communication network. IRJET-International Research Journal of Engineering and Technology 3(5)
12. Dinu D, Corre YL, Khovratovich D, Perrin L, Großschädl J, Biryukov A (2015) Triathlon of Lightweight Block Ciphers for the Internet of Things. IACR Cryptology ePrint Archive, 209
13. Elhoseny M, Elminir H, Riad A, Yuan X (2016) A secure data routing schema for WSN using elliptic curve cryptography and homomorphic encryption. Journal of King Saud University-Computer and Information Sciences 28(3):262–275
14. Elhoseny M, González GR, Abu-Elnasr OM, Shawkat SA, Arunkumar N, Farouk A (2018) Secure medical data transmission model for IoT-based healthcare systems. IEEE Access 6:20596–20608
15. Elhoseny M, Hosny A, Hassanien AE, Muhammad K, Sangaiah AK (2017) Secure automated forensic investigation for sustainable critical infrastructures compliant with green computing requirements. IEEE Transactions on Sustainable Computing

16. Elhoseny M, Yuan X, El-Minir HK, Riad AM (2016) An energy efficient encryption method for secure dynamic WSN. Security and Communication Networks 9(13):2024–2031
17. Farouk A, Batle J, Elhoseny M, Naseri M, Lone M, Fedorov A, Alkhambashi M, Ahmed SH, Abdel-Aty M (2018) Robust general N user authentication scheme in a centralized quantum communication network via generalized GHZ states. Front Phys 13(2):130306
18. Ghouti L, Ibrahim MK, Gutub AA (2013) Method of Performing Elliptic Polynomial Cryptography with Elliptic Polynomial Hopping. USPTO: United States Patents & Trademark Office, Patent number: US_8385541, Filing date: Feb 18, 2010, Patent Issue date: Feb 26
19. Gulcan E, Aysu A, Schaumont P (2014) A Flexible and Compact Hardware Architecture for The SIMON Block Cipher. Third International Workshop on Lightweight Cryptography for Security and Privacy: LightSec 2014, PP. 34–50. Istanbul
20. Gutub A (2003) High Speed Low Power GF(2^k) Elliptic Curve Cryptography Processor Architecture. IEEE 10th Annual Technical Exchange Meeting, KFUPM, Dhahran
21. Gutub A (2006) 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):44–52
22. Gutub A (2011) Subthreshold SRAM Designs for Cryptography Security Computations. In: ICSECS 2011 - 2nd International Conference on Software Engineering and Computer Systems, Universiti Malaysia Pahang, Kuantan
23. Gutub A (2015) Exploratory Data Visualization for Smart Systems. Smart Cities 2015 - 3rd Annual Digital Grids and Smart Cities Workshop, Burj Rafal Hotel Kempinski, Riyadh
24. Gutub A, Al-Juaid N, Khan E (2017) Counting-based secret sharing technique for multimedia applications. Multimedia Tools and Applications: An International Journal. Springer . https://doi.org/10.1007/s11042-017-5293-6ISSN 1380-7501
25. Gutub A, Khan E (2011) Using Subthreshold SRAM to Design Low-Power Crypto Hardware. International Journal of New Computer Architectures and their Applications (IJNCAA) 1(2):474–483
26. Gutub A, Khan F (2012) Hybrid Crypto Hardware Utilizing Symmetric-Key & Public-Key Cryptosystems. In: International Conference on Advanced Computer Science Applications and Technologies – ACSAT2012, Palace of the Golden Horses, Kuala Lumpur
27. Gutub A, Tabakh AA, Al-Qahtani A, Amin A (2013) Serial vs. Parallel Elliptic Curve Crypto Processor Designs. In: IADIS International Conference: Applied Computing 2013, pp. 67–74, Fort Worth
28. Gutub A, Tahhan H (2008) Efficient Adders to Speedup Modular Multiplication For Cryptography. WoSPA 2008 – 5th IEEE International Workshop on Signal Processing and its Applications, University of Sharjah, UAE
29. Hossain M, Fotouhi M, Hasan R (2015) Towards an analysis of security issues, challenges, and open problems in the internet of things. IEEE World Congress on Services (SERVICES), PP. 21–28
30. Hosseinzadeh J, Hosseinzadeh M (2016) A comprehensive survey on evaluation of lightweight symmetric ciphers: hardware and software implementation. International Journal of Advances in Computer Science 5(4):31–41
31. Kölbl S, Leander G, Tiessen T (2015) Observations on the SIMON block cipher family. Annual Cryptology Conference, pp. 161–185, Springer
32. Kugler P, Nordhus P, Eskofier B (2013) Shimmer, Cooja and Contiki: A new toolset for the simulation of on-node signal processing algorithms. In Body Sensor Networks (BSN), IEEE International Conference on (pp. 1–6). IEEE
33. Loan NA, Hurrah NN, Parah SA, Lee JW, Sheikh JA, Bhat GM (2018) Secure and Robust Digital Image Watermarking Using Coefficient Differencing and Chaotic Encryption. IEEE Access 6:19876–19897
34. Mora H, Gil D, Terol RM, Azorín J, Szymanski J (2017) An IoT-Based Computational Framework for Healthcare Monitoring in Mobile Environments. Sensors 17(10):2302
35. Muhammad K, Ahmad J, Farman H, Jan Z, Sajjad M, Baik SW (2015) A Secure Method for Color Image Steganography using Gray-Level Modification and Multi-level Encryption. TIIS 9(5):1938–1962
36. Nithya R, Kumar DS (2016) Where AES is for Internet, SIMON could be for IoT. Procedia Technology Journal 25:302–309
37. Parah S, Ahad F, Sheikh JA, Bhat GM (2017) Hiding clinical information in medical images: A new high capacity and reversible data hiding technique. J Biomed Inform 66:214–230
38. Parah S, Akhoon J, Sheikh J, Loan NA, Bhat GM (2015) A high capacity data hiding scheme based on edge detection and even-odd plane separation. Annual IEEE India Conference (INDICON), pp. 1–5
39. Parah SA, Loan NA, Shah AA, Sheikh JA, Bhat GM (2018) Nonlinear Dynamics, A new secure and robust watermarking technique based on logistic map and modification of DC coefficient, In press

40. Parah SA, Sheikh JA, Ahad F, Bhat GM (2018) High Capacity and Secure Electronic Patient Record (EPR) Embedding in Color Images for IoT Driven Healthcare Systems, Internet of Things and Big Data Analytics Toward Next-Generation Intelligence, Edited book, pp. 409–437
41. Parah S, Sheikh JA, Ahad F, Loan NA, Bhat GM (2017) Information hiding in medical images: a robust medical image watermarking system for E healthcare. Multimedia Tools and Applications 76(8):10599–10633 Springer
42. Parah SA, Sheikh JA, Bhat GM (2018) Electronic Health Record hiding in Images for smart city applications: A computationally efficient and reversible information hiding technique for secure communication. Futur Gener Comput Syst. https://doi.org/10.1016/j.future.2018.02.023
43. Shehab A, Elhoseny M, Muhammad K, Sangaiah AK, Yang P, Huang H, Hou G (2018) Secure and robust fragile watermarking scheme for medical images. IEEE Access 6:10269–10278
44. Shehab A, Ismail A, Osman L, Elhoseny M, El-Henawy IM (2017) Quantified Self Using IoT Wearable Devices. In International Conference on Advanced Intelligent Systems and Informatics. (pp. 820–831). Springer, Cham
45. Trad A, Bahattab A, Othman S (2014) Performance Trade-offs of Encryption Algorithms for Wireless Sensor Networks. IEEE World Congress on Computer Applications and Information Systems (WCCAIS), PP. 1–6
46. Wu M, Lu TJ, Ling FY, Sun J, Du HY (2010) Research on the Architecture of Internet of Things. IEEE 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE) 5:484–487
47. Yang G, Zhu B, Suder V, Aagaard MD, Gong G (2015) The Simeck Family of Lightweight Block Ciphers. CHES: International Workshop on Cryptographic Hardware and Embedded Systems, PP. 307–329, Springer
48. Zajac P (2017) Upper bounds on the complexity of algebraic crypt- analysis of ciphers with a low multiplicative complexity. Des Codes Crypt 82(1):43–56
49. Zhang W, Thurow K, Stoll R (2014) A knowledge-based telemonitoring platform for application in remote healthcare. International Journal of Computers Communications & Control 9(5):644–654

Norah Alassaf is a graduated student with Master of Sciences (MS) degree in Computer Sciences & Engineering from Umm Al Qura University (UQU), Makkah - Saudi Arabia. She was fully sponsored by UQU with support from Ministry of Education working under the supervision of Prof. Adnan Gutub from Computer Engineering - UQU. Her MS program at UQU was specialized in the computer network track with security focus offered by the College of Computer and Information Systems offered at UQU-Makkah Campus, Saudi Arabia. Her research interest involved Internet of Things programming and implementations where she focused on research of Medical applications. Lately, she is working on improving low weight cryptography and its software enhancements where this research is her current contribution.

Adnan Gutub is currently working as Professor in Computer Engineering Department specialized in Information and Computer Security within Umm Al Qura University (UQU), Makkah -Saudi Arabia. He received his Ph.D. degree (2002) in Electrical & Computer Engineering from Oregon State University, USA. He had his BS in Electrical Engineering and MS in Computer Engineering both from KFUPM, Saudi Arabia. Adnan's research interests involved optimizing, modeling, simulating, and synthesizing VLSI hardware for crypto and security computer arithmetic operations. He worked on designing efficient integrated circuits for the Montgomery inverse computation in different finite fields. He has some work in modeling architectures for RSA and elliptic curve crypto operations. His current interest in computer security also involved steganography such as image based steganography and Arabic text steganography as well as secret sharing, proposing a new published model entitled 'counting-based secret sharing'. He is currently supervising and co-supervising MS Thesis of seven master students working toward their MS degree in computer science & engineering specialized in information security offered at Umm al-Qura University, Makkah, Saudi Arabia.

Shabir A. Parah has completed his M. Sc. and M. Phil and Ph.D. in Electronics from University of Kashmir, Srinagar in the year 2004, 2010 and 2013 respectively in the field of Signal processing and Data hiding. He is working as Assistant Professor in the department of Electronics and I. T, University of Kashmir, Srinagar. His fields of interest are Multimedia Signal Processing, Secure Communication, Digital Watermarking and Steganography. Dr. Shabir A. Parah has guided about twenty five PG projects. He has published more than 100 research papers in International/National journals and conferences.

Manal Al Ghamdi is currently working as an Assistant Professor at Department of Computer Science, Umm Al-Qura University, Saudi Arabia. She received her Ph.D. degree (2015) in Computer Vision from University of Sheffield, UK. She had her MS. degree (2010) in Advanced Computer Science from University of Sheffield, UK. She had her BS in Computer Science from KAU, Saudi Arabia. Manal's research interests involved Machine Learning, Computer Vision and Security. She worked on developing and evaluating video and image processing techniques for various applications. She has efficient research works in features extraction, coding, video sequences alignment, dimensionality reduction and topic modelling from videos. Her current interest in computer vision involved convolution deep learning (CNN) with specific interest to HealthCare applications. Her current interest, lately, involved some work within computer security adopting steganography and cryptography applied on images processing field.