Wireless Networks
https://doi.org/10.1007/s11276-020-02362-8

Lightweight and secure cipher scheme for multi-homed systems

Hassan N. Noura[1,2] · Reem Melki[1] · Mohammad Malli[2] · Ali Chehab[1]

Abstract

Multiple access technologies allow a network to connect to two or more communication networks efficiently, to enhance system performance. They are more robust and reliable than legacy technologies, which allow the connection to one network at a time. In this paper, a new and efficient dynamic and key-dependent cipher scheme is designed to secure multi-homed devices, in real-time, while preserving their main functionality and meeting the stringent Quality of Service requirements. The proposed scheme consists of a dynamic key derivation technique that mixes together a dynamic nonce and a secret key. The dynamic nonce is extracted from multiple channels, after establishing a new session. The proposed approach is based on simple operations such as substitution and permutation, to enhance the network robustness against different types of attacks. The dynamicity is based on the random nature of physical channels and as such, data confidentiality, integrity and source authentication are achieved with the least amount of resources, delay and computational complexity. The implementation of the proposed cipher scheme is appropriate for small and resource-constrained devices including IoT devices and mobile phones (battery-constrained devices). Theoretical and simulation results confirm that the cipher scheme has a better efficiency than other recent schemes, which require multiple rounds. In addition, a cryptanalysis discussion is provided towards proving that the proposed cipher is capable of resisting and preventing the different existing cryptanalysis and security attacks.

Keywords Multi-homing security · Efficient and secure cipher scheme · Cryptanalysis · One round cipher scheme

1 Introduction

Recently, multi-homed devices have been improved to support both cellular and Wi-Fi technologies, which led to a significant increase in traffic demand [1]. In legacy communication and networking systems, only a single access interface is utilized to transmit data. As a result, multi-homed systems have emerged as a promising candidate for modern wireless communications. This became an interesting research topic due to its various applications [2, 3]. Such systems ensure simultaneous data transmission over multiple connections, with the data stream being divided by each node into several data blocks, before transmitting each block using different communication technologies (e.g Wi-Fi, Cellular). As a result, remarkable network enhancements can be achieved, including system performance, throughput, reliability, latency, and Quality of Service (QoS) [2, 3]. To benefit from the existing parallel communication paths, most devices are being equipped with several radio interfaces. From a theoretical point of view, such a device achieves an enhanced communication quality with a higher data rate, along with enhanced reliability and increased robustness. Accordingly, in case of a failure over a communication link, devices can switch to other links to ensure continuous data transmission. Basically, multi-homing can be split into two main classes [4, 5]:

✉ Reem Melki
rmm71@mail.aub.edu

Hassan N. Noura
hn49@aub.edu.lb

Mohammad Malli
mmalli@aou.edu.lb

Ali Chehab
chehab@aub.edu.lb

1 Department of Electrical and Computer Engineering, American University of Beirut, Beirut, Lebanon

2 Department of Computer Sciences, Arab Open University, Beirut, Lebanon

Published online: 06 June 2020

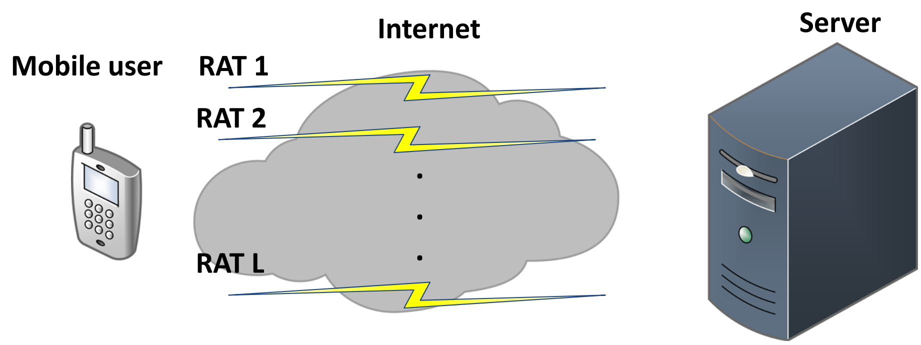


Fig. 1 System model shows the transmitted data over multi-RATs (*L* parallel communication paths)

1. *End-host multi-homing* includes a host with several network interfaces. More precisely, a multi-homed end-host (see Fig. 1) is a node with two or more different communication paths.
2. *Site multi-homing* includes a network with diverse redundant paths.

For example, several sensors may send data to Internet-of-Things (IoT) end devices, which communicate them to application servers. The IoT end devices will then transmit the collected data to an Access Point (AP) using the Wi-Fi technology. Data may also be transmitted directly using a specific communication protocol such as LORA. In fact, the collected data is transmitted based on the best effort transmission mechanism. However, in case of an emergency, monitored data must be sent to the appropriate data center and to the emergency response teams in a real-time manner. In such a case, a multi-homed system plays a significant role and it should exhibit all the desired QoS conditions such as low latency [6, 7].

1.1 Problem formulation

Multi-homed systems are prone to various types of security attacks due to the unreliable broadcast nature of wireless transmission. Hence, the practical benefits of multi-homed systems would be compromised in the absence of proper security, especially when sensitive information is being communicated (such as within a critical infrastructure). In general, there exists two types of security solutions; one is based on cryptographic algorithms and protocols, and the other is based on non-cryptographic techniques.

Typically, cryptographic algorithms ensure multiple security services such as data confidentiality, data integrity and source authentication. Confidentiality is achieved using symmetric key encryption (block or stream cipher), whereas data integrity and source authentication are ensured using a keyed hash function (or block cipher with message authentication operation mode such as CMAC). Recently, security issues of multi-homed systems have been presented in the literature [8, 9], which gives rise to a pressing need for designing appropriate security solutions.

Without encryption, multi-homed systems are vulnerable and weak against passive attacks. Once a passive attacker is able to intercept the different communication technologies used for transmission, the transmitted data is compromised. In order to enhance the protection of such a system, existing security solutions can be used. In general, a multi-homed system can establish a secure channel using IPsec or TLS for each communication path where the hosts support the cryptographic algorithms and protocols of TLS or IPsec to maintain both confidentiality and integrity, as well as source authentication. However, the existing cryptographic algorithms require a high latency and resource overhead, which would degrade the overall system performance.

Moreover, the utilization of small and computationally constrained devices, such as in IoT systems, introduces a hard challenge since achieving the desired security level is typically associated with an increase in latency and the required resources. Hence, in order to ensure the successful and secure deployment of multi-homed systems, appropriate and lightweight cryptographic algorithms should be put in place.

1.2 Related work

In the literature, several traditional cryptographic algorithms proved to be promising for ensuring data confidentiality and message authentication [10, 11]. An encryption algorithm is considered secure and efficient if it exhibits a high security level with low latency and required resources. These two requirements, in addition to low error propagation, should be taken into consideration when designing modern and efficient cipher schemes for wireless communication systems (channel effect, interference and fading).

Traditional block cipher techniques use symmetric-key encryption algorithms such as DES [12], which is based on Feistel Networks (FN), and AES [13], which is based on

Substitution Permutation Networks (SPN). Most existing cipher techniques require iterating a round function that uses static substitution and diffusion primitives. This function is iterated for a relatively large number of rounds, r, since the static primitives are known to attackers [14]. In fact, r is set according to a cryptanalsis study and it should be large enough to prevent statistical attacks such as differential and linear ones. On the other hand, having a small value of rounds, r, while maintaining the required security level, reduces the required computations and the latency [15–17]. This is the type of solutions that is most suitable for limited devices and real-time applications [14].

On the other hand, Chaotic cryptography, which is a growing paradigm, has become the focus of many researchers during the past decade. Several chaotic cryptographic algorithms have been presented, and which employ pseudo-random non-linear functions that ensure a high sensitivity of plaintext [18–21]. However, the main drawback of these algorithms is that they are based on floating-point operations and integer conversion operations. This dramatically increases the required resources and latency overhead, whilst also complicating the software and hardware implementations.

During the last decade, several optimized versions of AES have been developed, by reducing the computations and the associated latency, in order to make it more suitable for limited devices and real-time applications [14]. Moreover, a set of AES optimized assembly instructions was developed and introduced into recent Intel processors [22, 23]. However, still, AES may not be suitable for future applications and small devices. In this context, new efforts were directed towards the design of new and efficient modern cryptographic algorithms [14]. These algorithms [24–29] are designed to achieve the desired security level, but with a low number of iterations r, and/or by using simple substitution/diffusion operations in a given round function. In these algorithms, the minimum number of iterations achieved was 4, in the case of the Hummingbird2 cipher [15, 30].

1.3 Motivation

Effective and robust cryptographic algorithms are required to enable multi-homed systems to achieve the required performance, especially with resource-limited devices. Accordingly, in this paper, we propose a new dynamic cipher solution to improve the security level while maintaining low latency and minimal resource requirements, in addition to simple hardware implementation. Moreover, the proposed authentication-encryption scheme leverages the advantages of Multiple Radio Access Technology (multi-RAT) systems to enhance the randomness of the generated dynamic keys. The main goal is to minimize the required computations during encryption and authentication towards preserving the main functionality of multi-homed systems, improving their throughput and data rates, and increasing their immunity against attacks.

1.4 Contributions

The proposed cipher approach entails several contributions when compared to recent cipher algorithms, as listed below:

1.4.1 System performance

- *Flexibility* the input block size (n bytes), of the proposed encryption algorithm, may be configured depending on the device's memory, and as such, it is suitable for devices having different capabilities.
- *Simplicity* the simple structure of the proposed algorithm lends itself to simple hardware and software implementations, well appropriate for resource-limited devices.
- *Lightweight* The proposed encryption and authentication schemes are based on a single round of a function that uses simple substitution and permutation operations.
- *Low error propagation* The proposed algorithm does not ensure the avalanche effect, however, it overcomes this issue through the use of a dynamic key-dependent approach, which results in an acceptable level of error propagation.

1.4.2 Security performance

- *Variable cipher structure* The secret key is updated for each session, and it can be configured according to the needs of the applications. This makes the cryptanalysis very complicated and challenging to be performed. The proposed cipher achieves high key sensitivity since the dynamic key cannot be predicted. Moreover, short sessions make the cryptanalysis process even more complicated since a dynamic key is used only for a small set of plaintext messages. This results into a high degree of resistance and robustness against known powerful attacks.
- *Random block order* A permutation operation is applied at the block level in order to randomize the order of sequential blocks. This makes the proposed scheme less prone to possible future vulnerabilities, when compared to the existing encryption algorithms, which maintain the sequential order of encrypted blocks. The block permutation operation is a simple one, with low latency, and it does not impact the overall performance.

- *Desirable cryptographic performance* The proposed cipher scheme is very robust as it combines efficiently substitution, byte and block permutation, and the use of a random key-stream to achieve the desired randomness and uniformity properties, in addition to high levels of difference and key sensitivity.

1.5 Organization

This paper is structured as follows: Sect. 2 reviews multi-homed systems. Section 3 describes the proposed multi-homing security solution. In Sect. 4, the dynamic key generation technique is presented. Section 5 describes the different techniques used to construct the required cryptographic primitives (substitution and permutation tables), which are used to generate two different key-streams. Section 6 describes the proposed message authentication and confidentiality schemes. Section 7 analyzes the proposed scheme's security properties in terms of randomness, uniformity and sensitivity. Then, a cryptanalysis discussion is presented in Sect. 8. Section 9 analyzes the proposed cipher performance. Lastly, Sect. 10 concludes the work and highlights its prospects.

2 Multi-homing background

Recently, multi-homing systems managed to attract significant attention from various working domains due to the fact that most recent networking devices (smartphones) support various access technologies including Wi-Fi, LTE, and 3G, which are able to connect and transmit data simultaneously over multiple paths. Moreover, extensive research has demonstrated the advantages of transmitting data over several parallel connections:

1. it overcomes some of the drawbacks facing today's systems such as the limitation of simultaneously transmitting data over multiple paths;
2. it outperforms the performance and security of current network deployments, which use one path/connection at a time.

Moreover, this technology splits transmitted data across several paths, which offers additional benefits in terms of throughput, resource utilization and reaction to failures [1].

In the system model, illustrated in Fig. 1, input messages of D bits need to be processed and then to be sent over L possible communication paths (RATs). In fact, RAT_i has a data rate of R_i bits/second (bps). Therefore, the D bits are divided into L segments, each with a length of D_i bits such that $D = \sum_{i=1}{}^{L} D_i$. Therefore, the required delay to send the ith segment with D_i bits on RAT_i is:

$$T_i = \frac{D_i}{R_i}, \quad i = 1, 2, \ldots, L \tag{1}$$

The main purpose is to profit from the existence of several RATs in order to minimize the delay of transmission compared to when having only one RAT. It should be noted that all L RATs ($R_{\mathrm{T}} = \sum_{i=1}^{L} R_i$) act as a single virtual path with a cumulative data rate. Hence, the required execution time is reduced compared to the traditional case where only one communication link is available. In general, the value of D and the respective data rates R_i are known. Therefore, $T_{\min}$ can be computed according to the following equation:

$$T_{\min} = \frac{D}{R_{\mathrm{T}}} = \frac{\sum_{i=1}^{L} D_i}{\sum_{i=1}^{L} R_i} \tag{2}$$

Determining the optimal size of transmitted data corresponding to each link L_i is as follows:

$$L_i = R_i \times T_{\min} = \frac{R_i}{R_{\mathrm{T}}} \times D \qquad \text{(bits)} \tag{3}$$

A multi-homed system model is illustrated in Fig. 2. In this case, data is transmitted over two RATs ($L = 2$).

3 Proposed solution for secure multi-homed data transmission

The proposed approach is designed for an L-RAT scenario and is divided into three steps:

1. *Data confidentiality* The input data is encrypted using the proposed lightweight cipher scheme, which is described in detail in Sect. 6.
2. *Distribution* Then, the data is divided according to the optimal sizes corresponding to each communication link L_i. As such, the encrypted message $C = \{C_1, C_2, \ldots, C_L\}$ is divided into L parts. For simplicity, we consider that $L = 2$, and messages of size L_1 and L_2 bits are sent over RAT1 and RAT2, respectively.
3. *Message authentication* Each fragment is authenticated using a lightweight keyed hash function such as the one presented in [31]. Finally, the first group of encrypted data, C_1, and its corresponding MAC are transmitted over RAT1 and the remaining group, C_2, and its corresponding MAC are transmitted over RAT2 (Fig. 3).

The proposed approach, for the scenario in Fig. 1, is illustrated in Fig. 3. First, data is encrypted, then divided into L parts (sub-messages). Afterwards a keyed-hash function is applied on each sub-encrypted message C_i for $i = 1, 2, \ldots, L$. The obtained message authentication

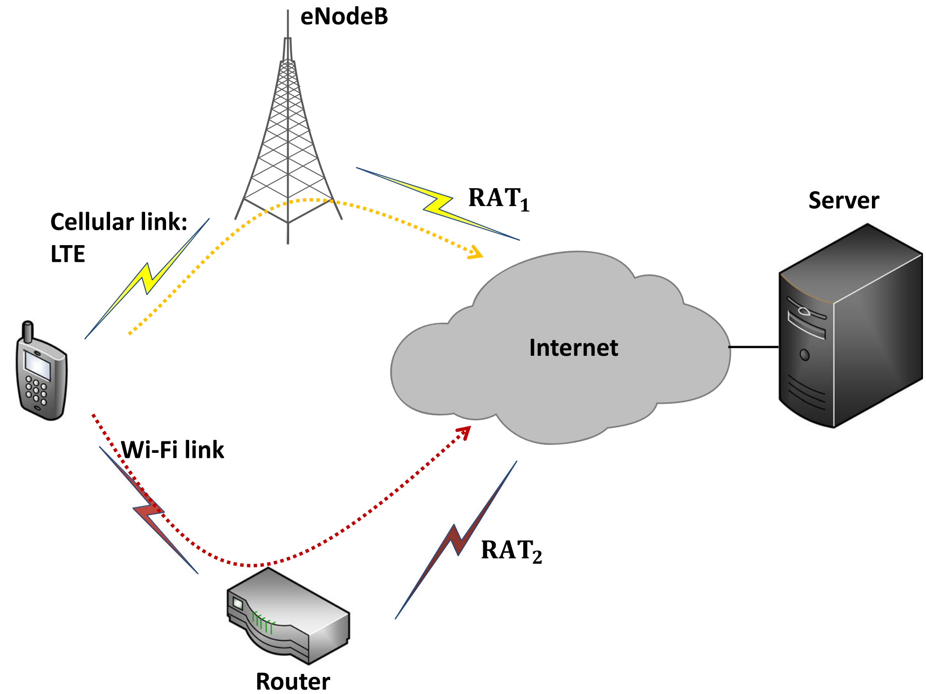


Fig. 2 A special case of a multi-homed system with two RATs ($L = 2$), including LTE cellular communications and Wi-Fi direct

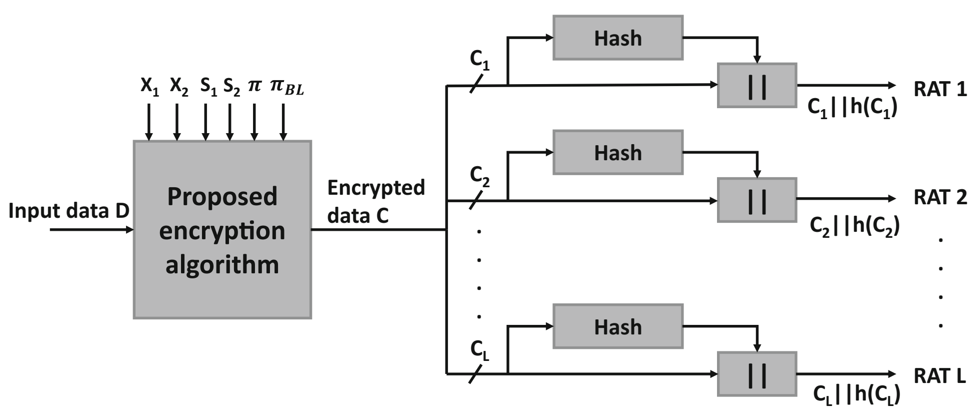


Fig. 3 Block diagram of the proposed data confidentiality and message authentication scheme. The employed hash function is keyed and uses K_A as a secret key

code MAC_i is appended to the corresponding encrypted sub-message C_i and sent on RATi. The message authentication key, K_A, is obtained by mixing the four sub-dynamic keys ($K_A = DK_1 \oplus DK_2 \oplus DK_3 \oplus DK_4$), which will be described in the next section.

As a result, an eavesdropper can listen to one RAT and only intercept a part of the message, which is also encrypted. Hence, this eavesdropper is unable to recover any useful information. Even if an eavesdropper can connect and synchronize to all RATs, he/she will not be able to recover the original data. A receiver will not perform decryption unless the message authentication of each encrypted sub-message is validated. Note that for the receiver to verify the message authentication digest and decrypt the received data, both the sender and receiver should extract a nonce from the available (L) RATs. Another approach can be applied where the nonce is encrypted and sent to the legitimate destination. Then, the receiver can decrypt it, which maintains the security of the proposed key derivation function.

In this paper, all of the utilized notations are listed in Table 1.

4 Proposed key derivation function

The proposed dynamic key generation function is shown in Fig. 4. Each cryptographic primitive depends on a specific dynamic sub-key, and each dynamic key is based on two main inputs, the secret key K and the Nonce N_o, which changes for every session. These inputs are described as follows:

Table 1 Table of notation

Notation	Definition
N	Number of bytes of the input message
Tb	Number of bytes in one block ($Tb = 4, 8, \ldots, 32$). We fix it to 16 in this paper
nb	Number of blocks in corresponding message $\lceil \frac{N}{Tb} \rceil$
K	Secret key
N_o	Nonce extracted from multi-channels
DK	The generate dynamic key, which can be changed for each input message
DK_1	The first 16 most significant bytes of DK It is used to construct the first substitution table
DK_2	The second 16 most significant bytes of DKIt is used to construct the second substitution table
DK_3	The third 16 most significant bytes of DK It is used to construct a permutation table π with length equal to N
DK_4	The first 16 least significant bytes of DKIt is used to construct the block permutation table π_{BL}
π_{BL}	A permutation table is used to permute blocks in order to randomize their transmission over multi-RATs
S_i	The ith dynamic produced substitution table
S_i^{-1}	The inverse corresponding substitution table
P_i	The ith dynamic produced permutation table
P_i^{-1}	The inverse corresponding ith permutation table
D_i	The ith part of data that has a length L_i bits
C_i	The ith corresponding encrypted part and it should be transmitted over the ith RAT
$\mathsf{length}(x)$	The number of elements of vector x
$\mathsf{repmat}(x, 1, l)$	The table x is repeated for l times in the column dimension to form a vector with length equal to $l \times \mathsf{length}(x)$

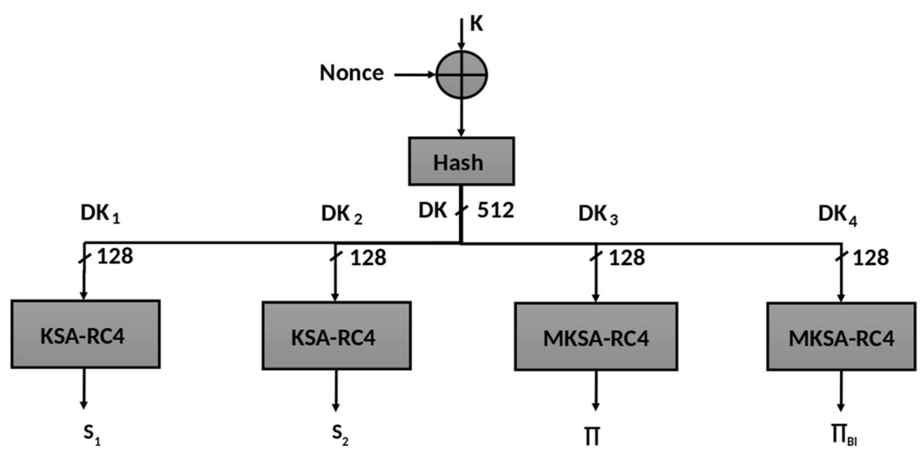


Fig. 4 Dynamic key generation technique. These sub-keys are used as means to form the cipher layer's

- *Secret Key K* It can either be stored or shared. For enhanced robustness, the secret key must constantly be updated after a specific period of time to be configured according to the application requirements.
- *Nonce N_o* Common channel parameters [32] such as distance and Angle of Arrival (AoA), Received Signal Strength (RSS), and Channel State Information (CSI) of each RAT are mixed together and hashed to produce a nonce. This allows the source and destination to be synchronized and to obtain the same nonce independently.

The secret key, K, and the Nonce, N_o, are mixed via the logical exclusive OR operation (XOR), and the corresponding output is hashed to generate a new dynamic key, DK. A secure cryptographic hash function (e.g SHA-512) should be used for the generation of the dynamic keys for every new session. SHA-512 is selected due to its high collision resistance degree. The dynamic key has a length of 512 bits, which ensures that the proposed cipher is robust against several attacks. Next, DK is divided into a set of sub-keys, which are used for the generation of the necessary cryptographic primitives.

Specifically, the dynamic key, DK, is divided into four main sub-keys $\{DK_1, DK_2, DK_3, DK_4\}$, each having a size of 16 bytes (128 bits).

- *Substitution sub-keys,* DK_1 and DK_2, consists of the first and second 16 most significant bytes of DK. They are used to build two different dynamic substitution tables, S_1 and S_2. Note that any key-dependent substitution table generation algorithm may be used. For example, we use the technique presented in [15], which is based on the RC4 Key Setup Algorithm (KSA) [33]. On one hand, KSA is iterated using DK_1 as a seed, to generate the first dynamic substitution table, and in parallel, KSA is iterated using DK_2 as a seed, to generate the second dynamic substitution table.
- *Permutation sub-keys,* DK_3 *and* DK_4, represent the third and fourth least significant 16 bytes of DK. They are employed to build two dynamic permutation tables, π and π_{BL}. In fact, π is used to permute the bytes of a message and to construct two different key-streams, X_1 and X_2, in cooperation with two substitution tables, S_1 and S_2. The block permutation process is done using π_{BL}. This cryptographic permutation can rely on any key-dependent permutation algorithm. In this paper, the KSA algorithm of RSA is modified [15] to generate a flexible permutation table. Also, to ensure that the permutation table satisfies the desired cryptographic properties, the seed should have a length greater than 64 bits [15]; we select a size of 128 bits.

All cipher primitives depend on the dynamic key, which leads to a high key sensitivity since any minor change in DK would result in different cryptographic primitives (Sect. 7.3). The derivation of these dynamic sub-keys is illustrated in Fig. 4, along with the proposed cryptographic primitive construction techniques, which are described next.

5 Construction of dynamic key-dependent cipher primitives

In this section, the proposed cryptographic construction techniques used to construct the required dynamic cryptographic primitives (substitution, permutation and key-stream) are described in details.

5.1 Dynamic substitution primitive

The confusion property is achieved by applying the substitution operation, which is considered as a non-linear operation (transformation). Two substitutions tables, S_1 and S_2, are required in the proposed cipher scheme; they are obtained using the *KSA* algorithm of RC4 as presented in [15].

The generated substitution tables S_1 and S_2 satisfy the desired cryptographic strength according to [15], and they introduce the confusion property, which provides the required immunity level against linear and differential attacks. Moreover, the advantage of the proposed solution is that substitution tables are unknown to the attackers and are updated for each new session.

5.2 Dynamic permutation primitive

In the proposed cipher scheme, two dynamic and flexible permutation tables π and π_{BL} are required: π is not only used to permute the input message bytes (*nb* blocks), but it is also used to produce the required key-streams, X_1 and X_2, which are used during the key-stream-mixing process. The flexible and dynamic permutation table, π, is generated in a similar manner to the substitution tables, based on a modified version of the KSA algorithm of RC4, as presented in [15]. The modified KSA algorithm is iterated, in parallel, with DK_3 and DK_4 as seeds, to construct π and π_{BL} with length N and nb elements, respectively. $\pi(i)$ corresponds to the π value at index i, and $0 \le \pi(i) \le N-1$. The permutation process is done by using a swap function, where $(\pi(i))$ corresponds to the permuted byte positions of the message.

It should be noted that inverse substitution and permutation tables are required for the decryption process. Since the generated S_i, π and π_{BL} are bijective, their corresponding inverse, S_i^{-1}, π^{-1} and π_{BL}^{-1} can be produced through the following equation:

$$Q^{-1}[Q(j)]=j\ ,where\ j=0,1,\ldots \tag{4}$$

5.3 Key-stream generation scheme

A key-stream, X_i, is generated using one substitution table, S_i, and a permutation table, π (Fig. 5). The substitution table is repeated $\lceil \frac{N}{256} \rceil$ times in the column dimension to form a vector T_i with length equal to or greater than N bytes (see Algorithm 1). Then, we remove the last elements of T that have indices greater or equal to N. Next, the permutation table is used to permute T and the output of this operation represents the produced key-stream. Such a technique exhibits a low computation complexity and simple implementation. Also, it ensures high randomness

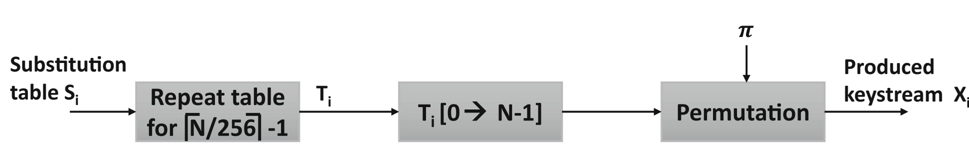


Fig. 5 Block diagram of the proposed key-stream generation scheme

and uniformity since no patterns are found in the produced key-stream, as seen in Fig. 6. Moreover, the substitution tables, S_1 and S_2, can be changed for every session or every input message and hence, new values of X_1 and X_2 are obtained.

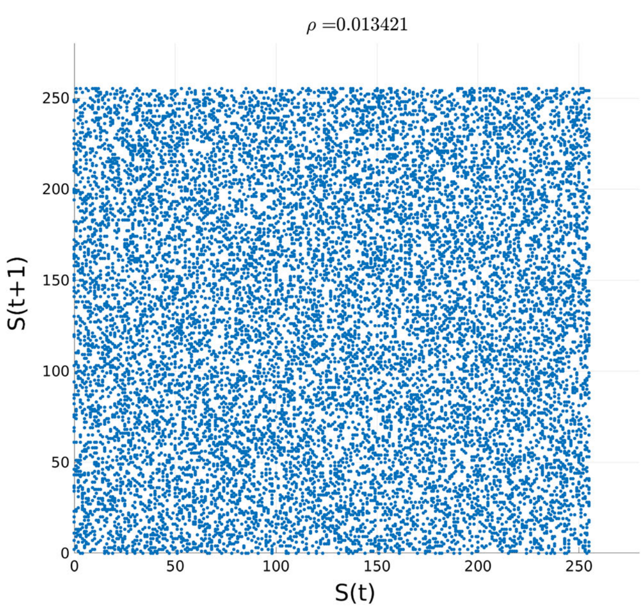
Algorithm 1 The Proposed key-stream Generation Technique

1: **procedure** GENERATION_KEY-STREAM(S_i, π)
2: $\quad T_i \leftarrow \mathsf{repmat}(S_i,\ 1,\ \lceil \frac{N}{256} \rceil)$
3: $\quad T_i \leftarrow T_i[0 \rightarrow N-1]$
4: $\quad X_i \leftarrow T_i[\pi]$
5: $\quad$ **return** X_i
6: **end procedure**

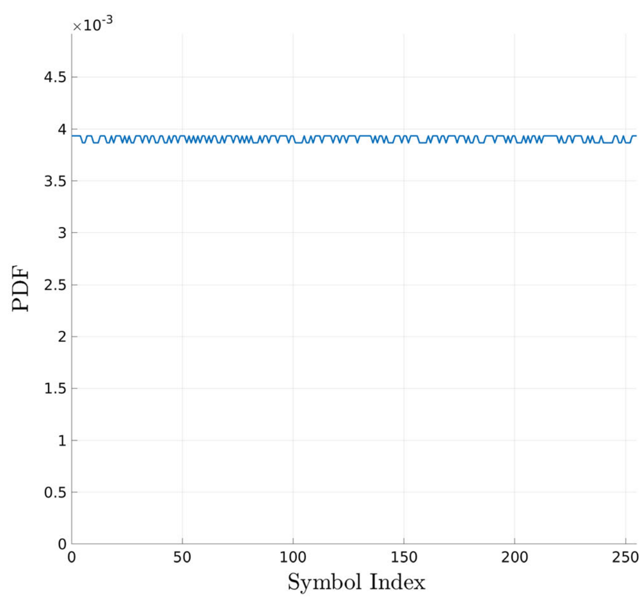
6 The proposed cipher algorithm

The main goal of this work is to design a lightweight and efficient cipher scheme for multi-homed systems, and especially for resource-limited devices. The round function should achieve the proper confusion and diffusion features [34] to prevent statistical attacks. Note that traditional ciphers use static round functions for several iterations to reach the required cryptographic strength. However, static keys are prone to various types of threats [35–37] since the confusion and diffusion primitives are known to attackers. The resistance against existing and powerful attacks can be enhanced by using the dynamic key-dependent cipher approach [38], since cipher primitives are frequently updated in a dynamic manner. This prevents attackers from recovering any valuable information from the intercepted ciphertext. This permits the reduction of r, which reduces the required resources and delay [39], while preserving the main functionality of multi-homed systems.

This section describes the different steps of our one round encryption/decryption algorithm, as shown in Figs. 7 and 8, respectively. The proposed cipher algorithm uses a symmetric secret key K, which is shared among communicating (legitimate) entities. As stated earlier, a dynamic key will be generated for each input session or message, from this key and a *Nonce*. *Dk* is split into four main sub-keys, which are used to construct the cipher primitives. This proposed cipher relies on a large-size dynamic key, which results into a high level of security using only *one round*.



(a) Recurrence



(b) PDF

Fig. 6 **a** Recurrence, and **b** PDF of the produced key-stream for a random secret key

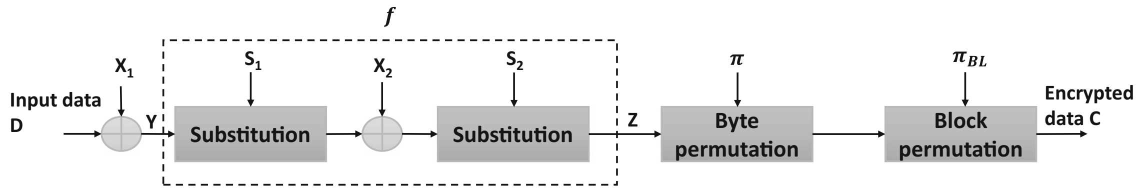


Fig. 7 Block diagram illustrating the key steps of the proposed lightweight one-round encryption scheme

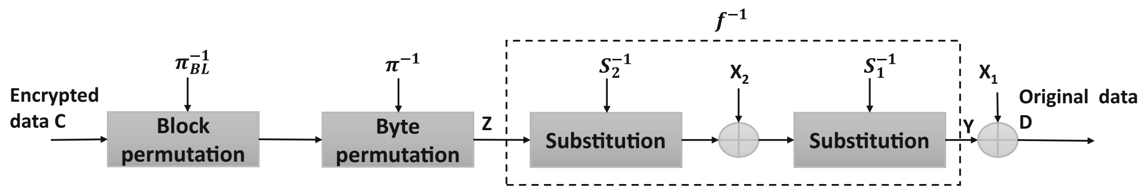


Fig. 8 The corresponding one round decryption algorithm

The proposed cipher takes as input a message D of size N bytes. The message is padded if necessary, so that it can be divided into nb blocks, each with a size of Tb bytes. Tb can be adjusted depending on the used application and its available memory space; however, throughout this paper this value will be fixed to 16. The total number of blocks is nb, where $nb = \lceil \frac{N}{Tb} \rceil$.

6.1 Encryption algorithm

In this part, the steps of the encryption algorithm, which are illustrated in Fig. 7, are described briefly.

The proposed encryption scheme is summarized in Algorithm 2. It is mainly divided into four sub-functions (operations), 1) Mixing with a Key-stream, 2) Substitution, 3) Byte-Permutation and 4) Block-Permutation.

6.1.2 Function *f*

Both dynamic S-boxes, S_1 and S_2, are used in this step. The output of the first step, Y, is first substituted using the first substitution table, S_1. Then, the output is XORed with the second key-stream, X_2. Next, the output is again substituted using the second substitution table, S_2. This function (f) can be expressed as follows:

$$Z = S_2(S_1(Y) \oplus X_2) \tag{6}$$

6.1.3 Byte permutation

Next, Z is permuted using the permutation table, π, which we express as:

Algorithm 2 The proposed One Round Encryption Algorithm.

procedure ONE_ROUND_ENCRYPTION($D, X_1, X_2, S_1, S_2, \pi$)
  $Y \leftarrow X1 \oplus D$
  $Z \leftarrow S2(S_1(Y) \oplus X_2)$
  $C \leftarrow Z(\pi)$
  return C
end procedure

6.1.1 Initial addition key-stream operation

The first key-stream, X_1, is XORed with the input message (Eq. 5). The main goal of this operation is to benefit from the uniformity and randomness of the produced key-stream, X_1.

$$Y = D \oplus X_1 \tag{5}$$

$$C = Z(\pi) \tag{7}$$

6.1.4 Block selection

The following step is to select the encryption blocks for the different RATs. This step introduces additional randomness and eliminates any sequential relation among neighbouring block elements in the encrypted messages, which makes it more difficult for attackers to decipher the transmitted messages. For the selection of the blocks of the encryption/decryption processes, the dynamic permutation table π_{BL} is used.

Finally, encrypted data is divided into L parts, which are sent to the intended receiver using the available communication links. The receiver executes the decryption algorithm, as described next.

6.2 Decryption algorithm

Figure 8 illustrates the decryption scheme and Algorithm 3 represents the corresponding pseudo-code. The decryption process is only performed after successfully validating the authenticity of the message and its integrity. After receiving and validating the L encrypted sub-messages, the receiver concatenates them to form C, which is reshaped into nb blocks. Then, decryption takes place to recover the original message, D. The decryption algorithm is very similar to the encryption algorithm except for some minor modifications; the operations are performed in reverse order, while using the inverse of function f, (f^{-1}), and the inverse substitution and permutation (S_1^{-1}, S_2^{-1}, π^{-1} and π_{Bl}^{-1}) tables.

Table 2 Statistical results of the listed tests by using original chosen message with the proposed cipher scheme with $L = 2$ and for 10,000 random keys

	Min	Mean	Max	Std
First encrypted sub-message $RAT-1$				
Dif	49.7991	50.0013	50.2159	0.0562
KS	49.7794	49.9977	50.1882	0.0560
ρ	−0.0102	−0.0001	0.0120	0.0032
$H-E$	7.9969	7.9978	7.9984	0.0002
Second encrypted sub-message $RAT-2$				
Dif	49.7566	50.0077	50.2541	0.0688
KS	49.7702	50.0015	50.2365	0.0693
ρ	−0.0101	−0.0002	0.0127	0.0039
$H-E$	7.9948	7.9964	7.9975	0.0005

Algorithm 3 The proposed One Round Decryption Algorithm.

```
procedure ONE_ROUND_DECRYPTION(C, X1, X2, S1, S2, π)
    Z ← C(π^{-1})
    Y ← S2^{-1}(S_2^- 1(Y) ⊕ X2)
    D ← X1 ⊕ Y
    return D
end procedure
```

6.2.1 Inverse byte permutation

After applying inverse block permutation, C is permuted using the permutation table π^{-1}. This step is expressed by the following equation:

$$Z = C(\pi^{-1}) \tag{8}$$

6.2.2 Inverse function f^{-1}

The inverse S-boxes, S_1^{-1} and S_2^{-1}, are used in this step. First, Z is substituted using the second inverse substitution table S_2^{-1}. Then, the output is XORed with the second key-stream X_2. Finally, Y will be recovered using the first inverse substitution table S_1^{-1}, as shown below:

$$Y = S_1^{-1}(S_2^{-1}(Z) \oplus X_2) \tag{9}$$

6.2.3 Inverse initial addition key-stream operation

Finally, the first key-stream, X_1, is generated as described at the source side. Then, it is XORed with the output of the previous step to recover the original data D, as expressed below:

$$D = Y \oplus X_1 \tag{10}$$

Using the proposed cipher scheme, confusion and diffusion properties are maintained through the use of a single, simple and dynamic round function. The robustness and efficiency of the scheme are evaluated and proved in the following section.

7 Security analysis

In this section, different analytic attacks that compromise data confidentiality, including chosen/known plain-text ciphert-text, and brute-force attacks [39], are presented to evaluate the levels of security and robustness of the proposed cipher scheme. For this purpose, we have conducted several experiments using normally distributed messages (data having a mean of 128 and standard deviation of 16). These experiments target important security measures such as statistical analysis and sensitivity. Table 2 shows the statistical results of the security tests that were performed.

7.1 Randomness of the key-stream generation algorithm

The produced key-stream should be highly random and uniform. In Fig. 6a, the plotted results validate that the produced key-stream, X, is completely non-linear, whereas Fig. 6b prove that the PDF of the produced key-stream is very close to a uniform one.

7.2 Statistical analysis

A cipher scheme should satisfy the randomness and uniformity properties to efficiently resist statistical attacks [39, 40]. Consequently, several statistical security tests were conducted to validate that the proposed scheme exhibits the desired uniformity and randomness degrees.

7.2.1 Uniformity analysis

The generated ciphertext should possess high levels of randomness and uniformity to guard against statistical attacks. The most commonly used method to evaluate uniformity is the PDF of the ciphertext, which should be close to a uniform distribution. In principle, the probability of each symbol should be very close to $\frac{1}{n}$, with n being the number of symbols. Figure 9 shows the PDF of the original message and that of the corresponding encrypted version ($L = 2$). The obtained results clearly show that the encrypted sub-messages have an almost uniform PDF with a value close to 0.039 ($\frac{1}{256}$). The entropy of the original and encrypted messages is presented and described to confirm the previous results. A specific entropy sequence $M = \{m_1, m_2, \ldots, m_n\}$, expressed in bits, quantifies the level of uncertainty [41]. The entropy value can be calculated according to the following equation:

$$H(M) = -\sum_{i=1}^{n} p(m_i) \log_2 \frac{1}{p(m_i)}, \tag{11}$$

with $p(m_i)$ and n representing the probability of occurrence of the symbol m_i and the number of symbols, respectively. The entropy test is performed at the sub-message level, where the ith encrypted sub-message is of length L_i bytes. When the entropy is close to $\log_2(n)$, then the sub-message is considered uniform.

The entropy test results are shown in Fig. 10. The test was applied on the original and encrypted (using random dynamic keys) messages having a length of $N = 1600$ bytes (the first ciphertext has a length $L_1 = 1000$ bytes, while the second ciphertext has a length $L_2 = 600$ bytes). The results show that each encrypted sub-message has an entropy value close to 8, which proves that the encrypted sub-messages have a uniform distribution, and redundancy is eliminated.

7.2.2 Correlation test between original and encrypted sub-messages

Eliminating spatial redundancy is a must for an efficient solution [42, 43]. As a result, ensuring a low correlation close to zero between the original and encrypted sub-messages is crucial to validate the high randomness degree of the proposed cipher. The correlation test is conducted by evaluating the coefficient correlation between the original sub-messages and the corresponding encrypted sub-messages. The correlation coefficient, r_{xy}, is expressed as follows:

$$r_{xy} = \frac{\mathrm{cov}(x, y)}{\sqrt{D(x) \times D(y)}} \tag{12}$$

where

$$E_x = \frac{1}{N} \times \sum_{i=1}^{N} x_i$$

$$D_x = \frac{1}{N} \times \sum_{i=1}^{N} (x_i - E(x))^2$$

$$\mathrm{cov}(x, y) = \frac{1}{N} \times \sum_{i=1}^{N} (x_i - E(x))(y_i - E(y))$$

Figure 11 plots the correlation coefficient for 1000 original and encrypted sub-messages ($L = 2$). In this test, a different secret key is used for each encrypted message. The results show that the ciphered message has a correlation coefficient close to 0, which proves that the presented cipher scheme clearly overcomes spatial redundancy.

7.2.3 Difference between plain-text and encrypted messages

The difference between the original and encrypted messages (both divided into L sub-messages), should be close to the desired value of 50%, at the bit level. The percentage of the bit difference between the original and the corresponding ciphered messages, for 1000 random dynamic keys, is shown in Fig. 12. The obtained values are very close to 50%, which proves that the independence criteria is successfully satisfied.

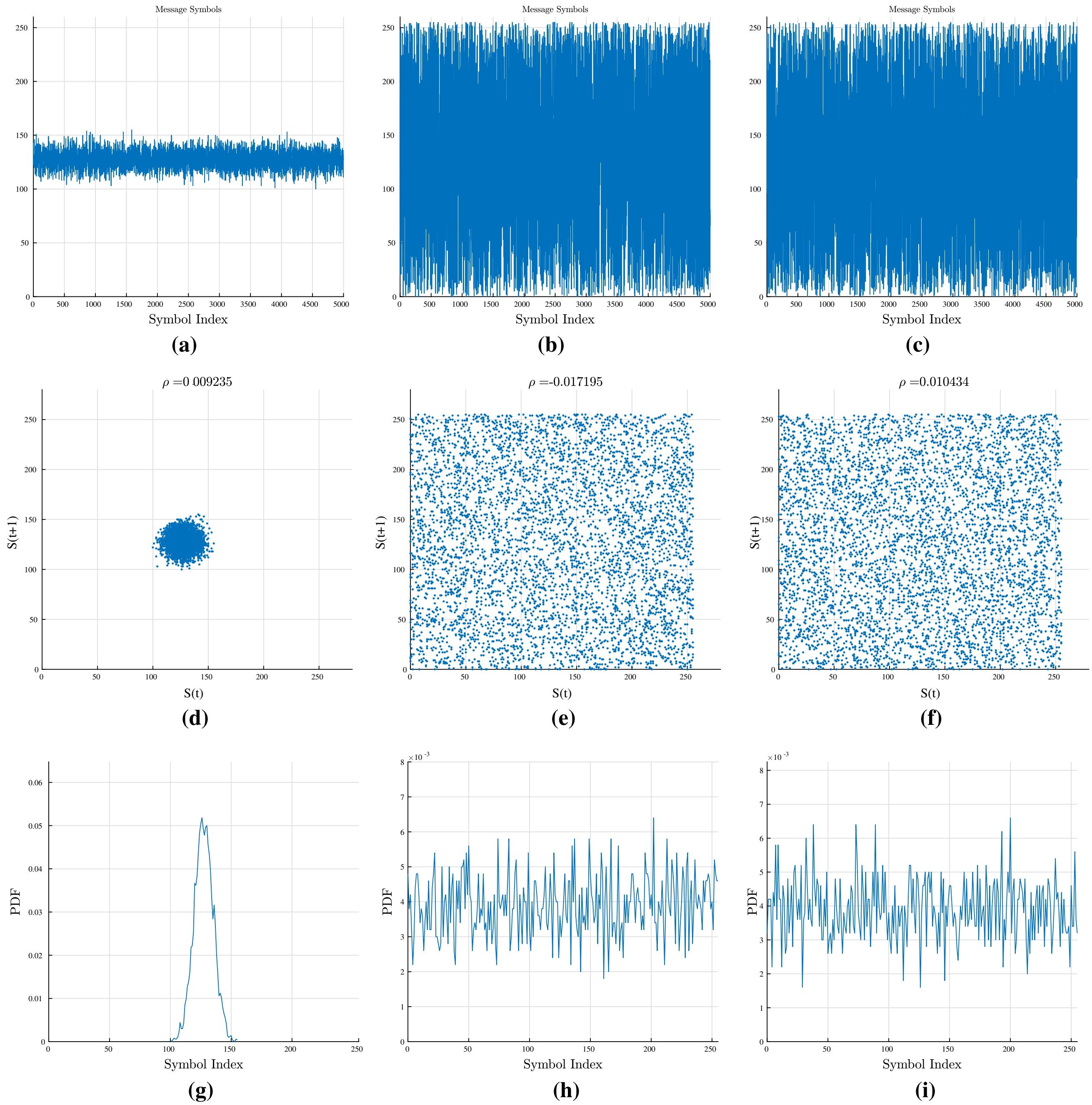


Fig. 9 **a** Variation of the original message (10,240 bytes), **b** the first encrypted sub-message (5120 bytes), **c** the second encrypted sub-messages (5120 bytes), **d** the original message recurrence, **e**, **f** recurrence of the corresponding encrypted sub-messages, **g** PDF of the original message, **h** and **i** PDF of the first and second encrypted sub-messages

7.3 Sensitivity test

In order to prevent sensitivity-related attacks, such as differential and linear ones, the sensitivity of the plaintext and key must be confirmed. Any one-bit difference in the key or in the plain-text should severely impact the resulting encrypted message, and result in a totally different encrypted block. Next, we assess the *Plain-text Sensitivity (PS)* and *Key Sensitivity (KS)*.

7.3.1 Plain-text sensitivity

Since the proposed scheme utilizes different dynamic keys for different messages, this sensitivity type becomes irrelevant to the proposed scheme. When a new message slightly differs from the previous message, a new dynamic key is generated and then, new cipher primitives are produced, which results into completely different encrypted messages.

7.3.2 Key sensitivity

This is an important test that quantifies the sensitivity of encrypted messages against any modification that may affect either the Nonce or the secret key. The key derivation function relies on a channel-extracted Nonce along with the secret key. In this test, two secret keys are used, K_1 and K_2, which have a one-bit difference. The produced dynamic keys are different and consequently different cryptographic primitives are generated. Also, the encrypted messages are obtained separately, where the Hamming distance of these encrypted sub-messages, C_1 and C_2, is calculated and presented in Fig. 13, against 1000 random dynamic keys. As it can be seen, the majority of these key sensitivity values is close to the ideal value of 50 %. This proves that the proposed encryption model is highly sensitive against any slight change in the dynamic key.

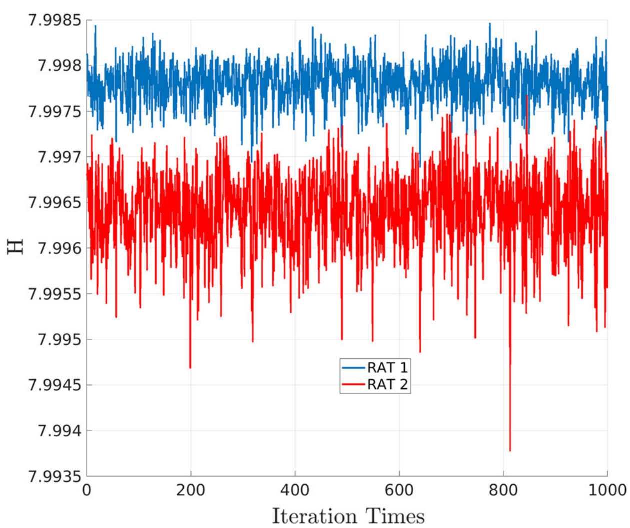


Fig. 10 Variation of the entropy of both encrypted sub-messages versus 1000 random dynamic keys

8 Cryptanalysis

Several types of attacks should be analyzed towards validating the robustness of any new proposed cipher scheme. Examples of these attacks include statistical, differential and linear attacks, as well as brute-force, and chosen/known plain-text/cipher-text attacks. Therefore, in this section, the proposed cipher scheme is analyzed against these attacks. The attacker has a complete knowledge of the cipher structure, the techniques which are employed to generate the different cipher primitives, the transmission protocol, the multi-RAT channels and their characteristics, but none regarding the secret key. Moreover, an attacker is able to collect the complete ciphertext by intercepting the transmitted message over the L RATs, which is not an easy task.

8.1 Immunity against statistical attacks

In order to resist statistical attacks, the encrypted data must have a high randomness level and a uniform distribution. According to Figs. 9 and 11, the encrypted data has a uniform distribution and a random non linear recurrence plot. Additionally, Fig. 12 indicates that the difference

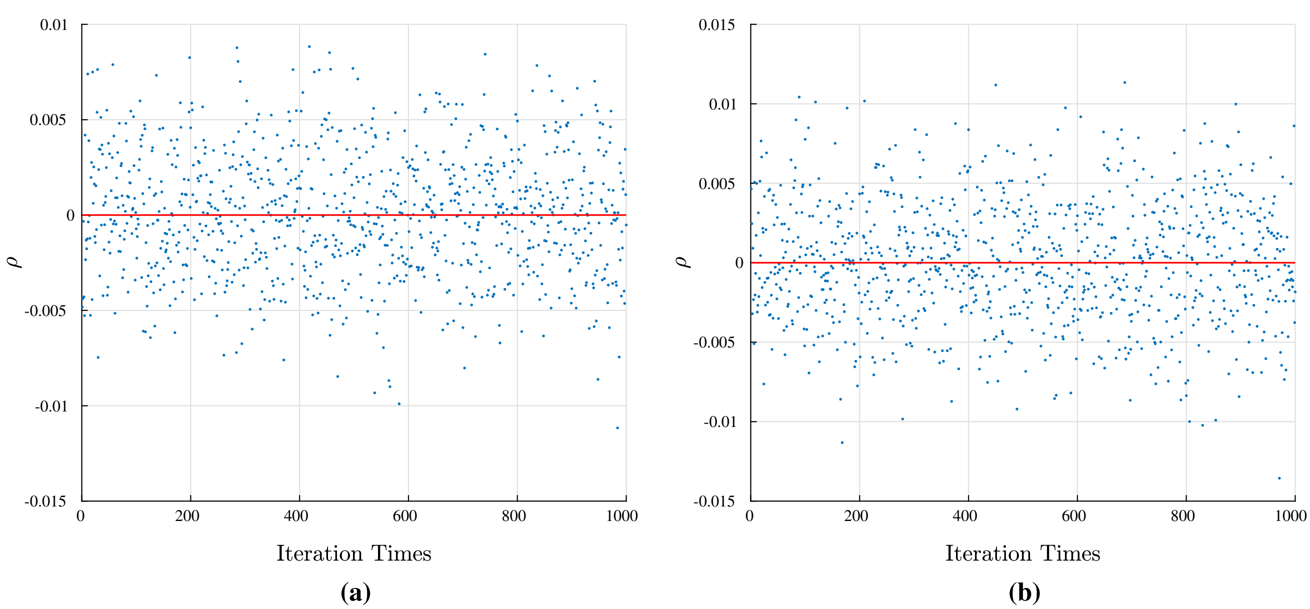


Fig. 11 Correlation Coefficient between: **a** the first original and encrypted sub-messages and **b** the second original and encrypted sub-messages for 1000 random dynamic keys

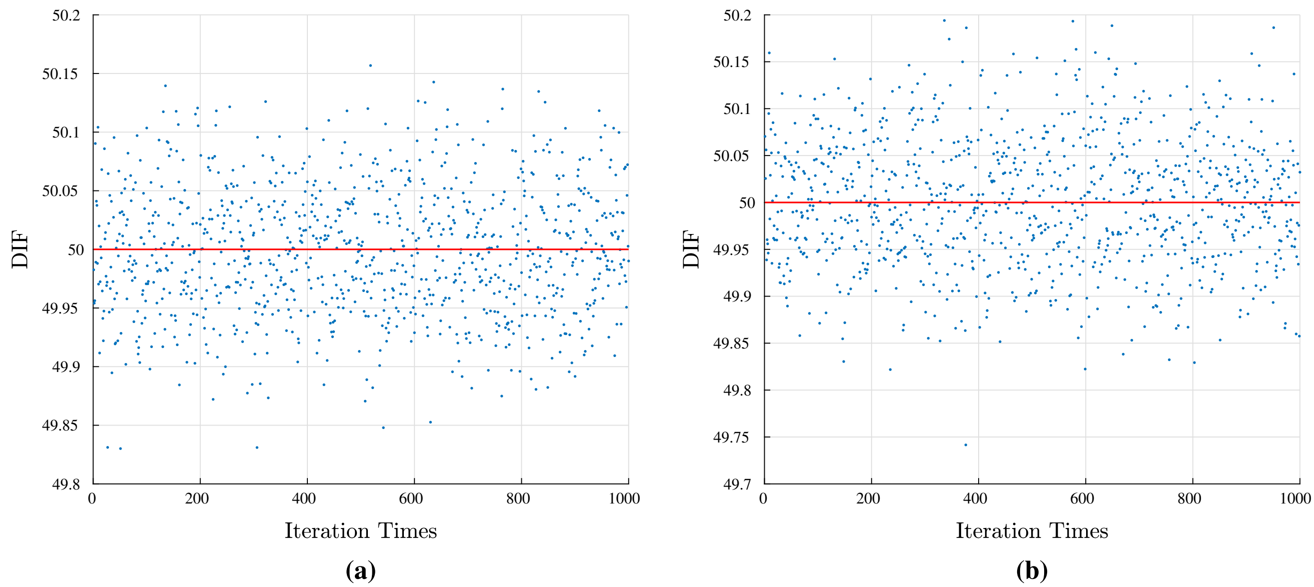


Fig. 12 Percentage difference between the first original and encrypted sub-messages **a** and between the second original and encrypted sub-messages for 1000 random dynamic keys

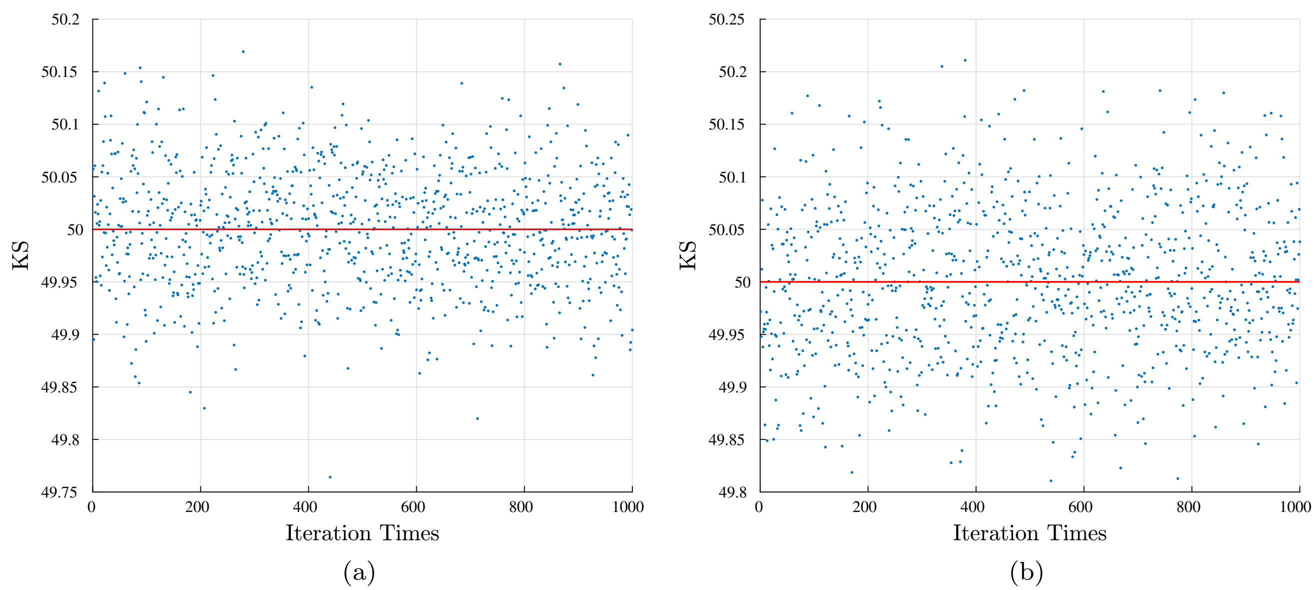


Fig. 13 Key sensitivity against 1000 random dynamic keys

between original and encrypted messages is always near 50%, which confirms the independence property between original and encrypted data. Therefore, the proposed cipher scheme exhibits the main statistical properties required to prevent statistical attacks.

8.2 Immunity against chosen/known plain-text/cipher-text attacks

The proposed technique includes a dynamic key structure, where the substitution, permutation, and key-stream primitives vary for each input data message. Therefore, no critical information can be deduced from the collected encrypted messages since they are encrypted with different dynamic keys. In addition, the proposed cipher mandates data interception over all available RATs (L) in contrast to the traditional case, where only one communication channel is used. Moreover, the key sensitivity to any bit of the secret key or Nonce is achieved according to Fig. 13. Consequently, the proposed scheme can be considered secure against chosen and known plain-text/cipher-text attacks.

8.3 Immunity against related-key attacks

Using the proposed cipher scheme, related-key attacks are mitigated since:

1. The proposed solution uses a dynamic key structure;
2. The key derivation function is a non invertible function (one-way function);
3. A high key sensitivity level is attained.

8.4 Weak keys

This key derivation function utilizes a secure hash function to produce a set of dynamic keys and it maintains a high randomness level according to statistical NIST tests. Therefore, the divided sub-keys inherently have a high randomness level. Moreover, all cipher primitives including substitution, permutation, and key-streams, are derived from a dynamic key. This, in turn, achieves the required cryptographic properties. In case any dynamic key possesses a certain weakness, only the symbols that are encrypted using the corresponding key are affected. Hence, the proposed approach can successfully resist any weak-key attack.

The nonce is directly related to the different L wireless channels that possess a high randomness level. Hence, the Nonce is changed in a random non-invertible manner towards generating a new dynamic key and consequently, new cipher primitives for each new input message. Therefore, the produced dynamic key can be varied frequently. This will complicate the task of attackers since recovering useful information about any sub-dynamic key or the dynamic key, will be an extremely hard task.

8.5 Resistance against Brute-force attacks

The size of the secret key is flexible and it can be set to 128 bits, 196 bits, or even 256 bits. This is very similar to the case of AES encryption. On the other hand, the size of the dynamic key is 512 bits. This renders any brute force attack infeasible. Moreover, the used Nonce must have a size of 512 bits to reduce the probability of any possible collision.

8.6 Resistance against future powerful attacks

This approach can resist future powerful attacks since different channel parameters are employed towards providing different dynamic and random Nonce, which will result in a totally different dynamic key and cipher primitives for each input data message. In addition, an attacker should be able to intercept and synchronize with all of the L communication channels.

In this section, we discuss and validate the security of the presented cipher scheme, in addition to the importance of the dynamic key approach. The previous tests results show that well-known attacks including brute force, statistical, linear and differential attacks, as well as chosen/known plaintext/ciphertext attacks are unable to break the proposed cipher algorithm.

Also note that attackers are not able to intercept data transmitted over multiple different RATs. In addition, the dynamicity of the dynamic key depends on L communication channels, and it increases randomness, periodicity and collision when compared to our previous scheme [16], where only one communication channel is used. Hence, it is extremely difficult to recover the dynamic key used for each input message or session, which also makes it very challenging to generate the corresponding cipher primitives.

9 Performance analysis

In this section, we discuss the criteria and tests needed to prove the efficiency of the proposed cipher in multi-homed systems. Generally, for a cipher scheme to be efficient, it should have (1) a low delay (low execution time) and (2) a low error propagation rate. The proposed technique is efficient and has low complexity since it employs a one-round structure and uses simple substitution and permutation operations. In addition, the proposed method to produce the required key-streams, X_1 and X_2, is based only on a permutation process, which is associated with a low delay and required resources. Simulation tests are performed using MATLAB to assess the performance of the proposed solution. Moreover, the latency is computed with different message sizes. Equally important, the effect of error propagation is also evaluated towards quantifying the scheme's immunity against channel errors.

9.1 Execution time

Low execution time, which results in low delay and energy consumption due to fewer computational operations, is an important condition for wireless devices that have limited computation resources or battery life. The average time to encrypt a message for 10, 000 iterations, having flexible sizes from 1 Kbyte to 100 kbytes, is calculated using *Matlab R2017b simulator, Intel Core i7, 3 GHz CPU, 2 GB RAM Intel and the Debian Linux OS (Operating System).* Figure 14 illustrates the obtained results. The proposed cipher scheme requires few computational operations and

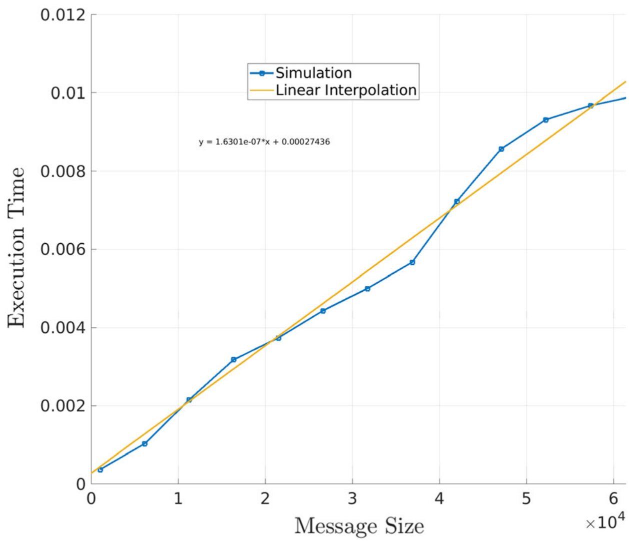


Fig. 14 Execution time (sec) of the proposed cipher scheme for different message sizes

low overhead. In fact, it is approximately $\frac{1}{3}$ of that of optimized "AES" with CTR mode, as assessed using Openssl. Therefore, the proposed scheme is simple and efficient when compared to traditional block ciphers; it exhibits high QoS in terms of delay and resource requirements when used in multi-homed systems.

9.2 Error propagation

Error propagation is an important criterion for assessing the performance of a cipher scheme against channel errors. In wireless transmission, error propagation is mainly caused by fading, interference and noise. However, the effect of this error on the transmitted ciphertext should be minimized as much as possible. A bit error refers to substituting a '1' bit with a '0' bit and vice-versa. An efficient cipher scheme should prevent the propagation of errors to avoid data corruption. This is extremely challenging due to the critical trade-off between error propagation and the avalanche effect. Using the proposed scheme, any bit error in any encrypted byte does not affect other bytes; it only affects the corresponding byte at a specific position in the decrypted message, which clearly minimizes error propagation.

10 Conclusion

This paper proposes a message authentication-encryption scheme for multi-homed systems. The proposed scheme has a single-round structure, and depends on a dynamic key that can be frequently changed. It has been shown that the proposed solution is efficient (low latency and error propagation), secure, and has low computational complexity. Unlike existing cipher techniques, the proposed scheme has a dynamic structure. Its corresponding structure includes different substitution/permutation primitives, along with two dynamic key-streams that change in an efficient and dynamic manner. This provides better robustness against different attack types. The extensive security analysis confirm that the cipher approach is secure against different attack types. Moreover, the test results clearly indicated that the proposed scheme outperforms the optimized AES implementation in TLS. This indicates the efficiency and applicability of this approach in delay-sensitive, and resource-limited multi-homed systems. In addition, one can benefit from multiple Radio Access Technologies to transmit encrypted data since an attacker is not able to intercept different radio technologies, simultaneously. Future enhancements aim at designing a physical layer encryption algorithm for multi-homed systems, which further reduces the associated latency and required resources.

Acknowledgements This paper is supported with funds from the Maroun Semaan Faculty of Engineering and Architecture at the American University of Beirut.

References

1. Melki, R., Mansour, M. M., & Chehab, A. (2018). A fairness-based congestion control algorithm for multipath TCP. In *IEEE Proceedings of wireless communications network conference (WCNC)*. IEEE
2. Wen, P., Dan, L., Gan, R., Xiao, Y., & Fu, B. (2018). Power allocation for multi-homing heterogeneous network aided by energy harvesting. In *IEEE Proceedings of the computing and communication workshop and conference (CCWC)* (pp. 515–519). IEEE.
3. Zhang, X., & Yang, F. (2017). Joint bandwidth and power allocation for energy efficiency optimization over heterogeneous lte/wifi multi-homing networks. In *IEEE Proceedings of the wireless communications network conference (WCNC)* (pp. 1–6). IEEE.
4. Ismail, M., & Zhuang, W. (2014). Mobile terminal energy management for sustainable multi-homing video transmission. *IEEE Transactions on Wireless Communications*, *13*(8), 4616–4627.
5. Chen, Y., Yuan, A. S., Lu, K., & Wu, Q. (2013). The study of 6LoWPAN with SCTP multi-homing in smart grid. In *IEEE Proceedings of the international conference on sensing technology (ICST)* (pp. 735–740). IEEE.
6. Chatterjee, A. (2017). Internet of things (IoT). In *Building apps for the universal windows platform* (pp. 157–165). Berlin: Springer.
7. Lee, Y., Jeong, J., & Son, Y. (2017). Design and implementation of the secure compiler and virtual machine for developing secure iot services. *Future Generation Computer Systems*, *76*, 350–357.
8. Shuminoski, T., Janevski, T., Risteski, A., & Bogdanoski, M. (2017). Security and QoS framework for 5G and next generation mobile broadband networks. In *IEEE Proceedings of the international conference on smart technologies (EUROCON)* (pp. 104–109). IEEE.

9. Venkadesh, P., Dhas, J., & Divya, S. (2015). Techniques to enhance security in SCTP for multi-homed networks. In *IEEE Proceedings of the global conference communication technologies (GCCT)* (pp. 468–473). IEEE.
10. Schneier, B. (2007). *Applied cryptography: Protocols, algorithms, and source code in C*. London: Wiley.
11. Ferguson, N., Schneier, B., & Kohno, T. (2011). *Cryptography engineering: Design principles and practical applications*. London: Wiley.
12. Biham, E., & Shamir, A. (1993). *Differential cryptanalysis of the data encryption standard* (Vol. 28). New York: Springer.
13. Daemen, J., & Rijmen, V. (2002). *The design of Rijndael: AES-the advanced encryption standard*. Berlin: Springer.
14. Beaulieu, R., Shors, D., Smith, J., Treatman-Clark, S., Weeks, B., & Wingers, L. (2015). Simon and speck: Block ciphers for the internet of things. *IACR Cryptology ePrint Archive*, *2015*, 585.
15. Noura, H., Chehab, A., Sleem, L., Noura, M., Couturier, R., & Mansour, M. One round cipher algorithm for multimedia iot devices. In *Multimedia tools and applications* (pp. 1–31).
16. Noura, H., Melki, R., Chehab, A., Mansour, M., & Martin, S. (2018). *Efficient and secure physical encryption scheme for low-power wireless M2M devices, in IWCMC 2018 Security Symposium (IWCMC-Security 2018)*. Cyprus: Limassol.
17. Noura, H. N., Chehab, A., & Couturier, R. (2019). Efficient & secure cipher scheme with dynamic key-dependent mode of operation. In *Signal processing: Image communication* (vol. 78, pp. 448–464).
18. Kocarev, L. (2001). Chaos-based cryptography: A brief overview. *IEEE Circuits and Systems Magazine*, *1*(3), 6–21.
19. Noura, H., Henaff, S., Taralova, I., & El Assad, S. (2009). Efficient cascaded 1-d and 2-d chaotic generators. *IFAC Proceedings Volumes*, *42*(7), 122–127.
20. Noura, H., El Assad, S., Vladeanu, C., & Caragata, D. (2011). An efficient and secure SPN cryptosystem based on chaotic control parameters. In *International conference for internet technology and secured transactions (ICITST)* (2011). IEEE, *2011*, 226–231.
21. Noura, H., Guyeux, C., Chehab, A., Mansour, M., & Couturier, R. (2019). Efficient chaotic encryption scheme with ofb mode. *International Journal of Bifurcation and Chaos*, *29*(05), 1950059.
22. Gueron, S. (2009). Intel's new aes instructions for enhanced performance and security. In *FSE* (vol. 5665, pp. 51–66). Berlin: Springer.
23. O'Melia, S., & Elbirt, A. (2010). Enhancing the performance of symmetric-key cryptography via instruction set extensions. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, *18*(11), 1505–1518.
24. Hong, D., Lee, J., Kim, D., Kwon, D., Ryu, K., & Lee, D. (2014) LEA: A 128-bit block cipher for fast encryption on common processors. In *Information security applications* (pp. 3–27). Berlin: Springer.
25. Guo, J., Peyrin, T., Poschmann, A., & Robshaw, M. (2011). The LED block cipher. In *Cryptographic hardware and embedded systems (CHES)* (pp. 326–341). Berlin: Springer.
26. Kumar, M., Pal, S., & Panigrahi, A. (2014). FeW: A lightweight block cipher. *IACR Cryptology ePrint Archive*, *2014*, 326.
27. Borghoff, J., Canteaut, A., Güneysu, T., & Kavun, E., et al. (2012). Prince—A low-latency block cipher for pervasive computing applications. In *Advances in cryptology (ASIACRYPT)* (pp. 208–225). Berlin: Springer.
28. Shibutani, K., Isobe, T., Hiwatari, H., Mitsuda, A., Akishita, T., & Shirai, T. (2011). Piccolo: An ultra-lightweight blockcipher. In *Cryptographic hardware and embedded systems-CHES* (Vol. 2011, pp. 342–357). Berlin: Springer.
29. Suzaki, T., Minematsu, K., Morioka, S., & Kobayashi, E. (2013). TWINE: A lightweight block cipher for multiple platforms. In L. Knudsen & H. Wu, (Eds.), *Selected areas in cryptography, ser. Lecture notes in computer science* (Vol. 7707, pp. 339–354). Berlin: Springer.
30. Noura, H., Chehab, A., Noura, M., Couturier, R., & Mansour, M. M. (2019). Lightweight, dynamic and efficient image encryption scheme. *Multimedia Tools and Applications*, *78*(12), 16 527–16 561.
31. Noura, H., Hussein, S., Martin, S., Boukhatem, L., & AlAgha, K. (2015). ERDIA: An efficient and robust data integrity algorithm for mobile and wireless networks. In *IEEE Proceedings of the wireless communications network conference (WCNC)* (pp. 2103–2108). IEEE.
32. Badawy, A., et al. (2016). Unleashing the secure potential of the wireless physical layer: Secret key generation methods. *Physical Communication*, *19*, 1–10.
33. Rivest, R. (1992). *The RC4 encryption algorithm*. rsa data sec, Inc. (1998)
34. Shannon, C. E. (1949). Communication theory of secrecy systems. *MD Computing*, *15*(1), 57–64.
35. Zhang, P., Jiang, Y., Lin, C., Fan, Y., & Shen, X. (2010). P-coding: Secure network coding against eavesdropping attacks. In *Proceedings of the IEEE international conference on computer communications (INFOCOM)* (pp. 1–9). IEEE.
36. Pradeep, L., & Bhattacharjya, A. (2013). Random key and key dependent s-box generation for aes cipher to overcome known attacks. In *International symposium on security in computing and communication* (pp. 63–69). Berlin: Springer.
37. Paar, C., & Pelzl, J. (2009). *Understanding cryptography: A textbook for students and practitioners*. Berlin: Springer.
38. Noura, H., Sleem, L., Noura, M., Mansour, M. M., Chehab, A., & Couturier, R. (2017). A new efficient lightweight and secure image cipher scheme. *Multimedia Tools and Applications*,. https://doi.org/10.1007/s11042-017-5124-9.
39. Noura, H., Chehab, A., Sleem, L., Noura, M., Couturier, R., & Mansour, M. M. (2018). One round cipher algorithm for multimedia iot devices. *Multimedia Tools and Applications*,. https://doi.org/10.1007/s11042-018-5660-y.
40. Xu, S., Wang, Y., Wang, J., & Tian, M. (2008). Cryptanalysis of two chaotic image encryption schemes based on permutation and xor operations. In *IEEE Proceedings of the computational intelligence and security (CIS)* (vol. 2, pp. 433–437). IEEE.
41. Zhang, G., & Liu, Q. (2011). A novel image encryption method based on total shuffling scheme. *Optics Communications*, *284*(12), 2775–2780.
42. Norouzi, B., Seyedzadeh, S. M., Mirzakuchaki, S., & Mosavi, M. (2014). A novel image encryption based on hash function with only two-round diffusion process. *Multimedia Systems*, *20*(1), 45–64.
43. Rhouma, R., & Belghith, S. (2008). Cryptanalysis of a new image encryption algorithm based on hyper-chaos. *Physics Letters A*, *372*(38), 5973–5978.

Hassan N. Noura received his degree in Computer and Communication Engineering from the IUL University in 2008, Lebanon, and his Ph.D. degree from Polytech'Nantes in 2012, France. In 2013, Noura joined Paris-Sud XI University, as a postdoctoral researcher, after that as research engineering at CEA, Grenoble. After that, he joined QMIC in 2015 and Telecom ParisTech in 2016. Recently, he gets his HDR in 2016 and joins AUB in 2017. His research interests include cryptography, network security, secure network coding, secure multimedia and secure distributed system.

Reem Melki received her BS degree in electrical and computer engineering from the Rafik Hariri University in 2013, and her MS degree in 2015. She is currently a Ph.D. student at the American University of Beirut (AUB). Her main areas of research interests include security and privacy in wireless and mobile communication.

Mohammad Malli received his degree in electrical and computer engineering from the Lebanese university in 2000, his master's degree in computer networks in 2001 and his Ph.D. in computer communication in 2006. His main areas of research interests cover the wireless and mobile communication.

Ali Chehab received his Bachelor degree in EE from AUB in 1987, the Master's degree in EE from Syracuse University in 1989, and the PhD degree in ECE from the University of North Carolina at Charlotte, in 2002. From 1989 to 1998, he was a lecturer in the ECE Department at AUB. He rejoined the ECE Department at AUB as an Assistant Professor in 2002, became Full Professor in 2014. He received the AUB Teaching Excellence Award in 2007. He teaches courses in Programming, Electronics, Digital Systems Design, Computer Organization, Cryptography, and Digital Systems Testing. His research interests include: Wireless Communications Security, Cloud Computing Security, Multimedia Security, Trust in Distributed Computing, Low Energy VLSI Design, and VLSI Testing. He has about 210 publications. He is a senior member of IEEE and a senior member of ACM.