Chaotic NHCP: Building an Efficient Secure Framework for Cloud Computing Environment Based on Chaos Theory

Diaa Salama Abdul Minaam, Mostafa Abdullah Ibrahim, and Elsayed Badr
(Corresponding author: Diaa Salama Abdul Minaam)

Information Systems Department, Faculty of Computers and Informatics, Benha University
Benha City, Egypt
(diaa.salama@fci.bu.edu.eg)

(Received Dec. 4, 2018; Revised and Accepted June 5, 2019; First Online Jan. 21, 2020)

Abstract

Cloud computing is an advanced trend, which provides access to applications and resources over the internet. In a cloud computing environment, the data is stored on remote servers accessed through the internet. The increasing volume of necessary data brings up more focus on securely storing data. Encryption plays a vital role in security for different types of data. The existing methods encrypt all data using the same key without taking into account the confidentiality level of data, which in turn will increase the encryption time. In this research, a novel encryption algorithm based on chaos theory in the cloud computing environment is developed. The new hybrid cryptography algorithm based on chaotic mapped called (Chaotic NHCP). Chaotic NHCP uses a classification method. The new framework of data encryption operates as follows, Firstly, KNN method is used to classify the data credibility level, and then Fast RSA algorithm and blowfish algorithm are used to encrypt the data to achieve the effect of Fast data encryption. The objects are classified by a maximum value of its neighbours, with the object being assigned to the class with most common among its K-nearest neighbours. Then, the 32-bit plaintext data was split into two 16-bit plaintext data, and the 32-bit ciphertext data was synthesised after encryption by Fast RSA and Blowfish hybrid algorithm, respectively. The proposed method was tested with different encryption algorithms and evaluated according to the encryption time, throughput and power consumption. The experimental results show that the Chaotic NHCP method minimises the encryption time needed to secure data that leads to a suitable confidentiality level required for the data. In addition, it has high throughput and low power consumption along with time-saving. The proposed method has proven the superior in the performance of processing time when compared with other encryption algorithms.

Keywords: Chaotic Map; Classification; Cloud Computing; Fast RSA; Hybrid Cryptography Algorithms

1 Introduction

Today, cloud computing has become an incoming trend for many organisations and people as it provides a wide range of services such as, Platform as a Service (PaaS), Infrastructure as a Service (IaaS), and Software as a Service (SaaS) [52]. Cloud computing has many issues, and the most important ones are security and confidentiality. Confidentiality level of data is not taken into consideration in some cloud systems, which leads to encrypt additional or unrelated data [20, 52]. Figure 1 shows cloud service models [8, 12, 50].

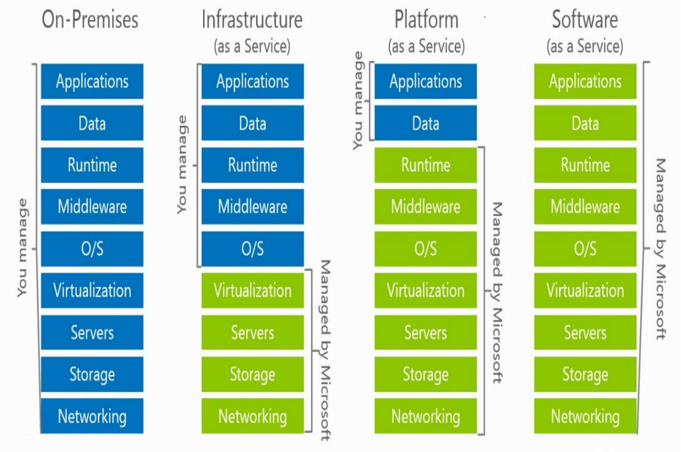


Figure 1: Cloud service models

In cloud computing, there are a lot of security challenges [31, 44, 45, 53] such Confidentiality, Privacy [1, 48], Data location [2, 3].

Encryption algorithms have two types: Symmetric and asymmetric key algorithms. Symmetric key algorithms uses the same key for encryption and decryption [16, 33]. DES, AES and Triple-DES, Blowfish [37, 49] are examples of symmetric key algorithms. Asymmetric algorithms have two keys; public key and private key for both encryption and decryption. RSA, Diffie-Hellman and homomorphic encryption are examples of asymmetric key algorithms. Symmetric algorithms are faster in performance than asymmetric algorithms because its key size is small. On the other hand, Symmetric algorithms have some drawbacks such as key transportation, as the key is transmitted to the received system before the original message is transmitted. Figure 2 shows the structure of the Blowfish [19, 32].

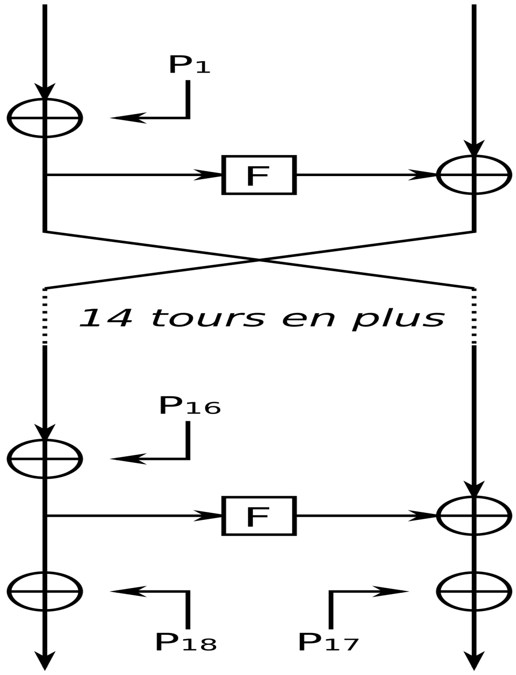


Figure 2: Structure of blowfish

In asymmetric algorithms, there is no need to exchange keys, thus solving the key distribution problem [44] of symmetric encryption algorithms. The primary advantage of public-key algorithms is increased security [48]. On the other hand, a disadvantage of using public-key cryptography for encryption is speed; There are secret-key encryption methods which are faster than currently available public-key encryption algorithm. RSA algorithm is illustrated in Figure 3. Disadvantages of symmetric and asymmetric encryption algorithms have motivated us to apply hybrid encryption algorithm.

Fast RSA [14, 23, 47] uses a modulus in form N = pr qs such that p, q are two distinct primes and r, s ¿= 2. It consists of the main three steps key generation, encryption, and decryption [18, 30]. So the main objective of this paper is to study the problem of data encryption algorithm based on chaos theory in the cloud computing environment and proposes a new framework of data encryption. Firstly, the KNN method is used to classify the data credibility level, and then Fast RSA algorithm and the Blowfish algorithm are used to encrypt the data to achieve the effect of Fast data encryption. An object is classified by a maximum value of its neighbours, with the object being assigned to the class with most common among its K nearest neighbours, which is named KNN. Then, the 32-bit plaintext data is splitted into two 16-bit plaintext data, and the 32-bit ciphertext data is synthesized after encryption by Fast RSA and Blowfish hybrid algorithm respectively.

1.1 Chaos Theory

Chaos theory [9] is a branch of mathematics that focuses on the behaviour of dynamic systems that are sensitive to initial conditions. It aims to predict the unexpected [36], and it concerns deterministic systems whose behaviour can be predicted [26]. Chaotic systems are predictable for a while and then ’appear’ to become random. The amount of time that the behaviour of a chaotic system can be predicted depends on three factors: How much uncertainty can be tolerated in the forecast, how accurately its current state can be measured, and a time scale depending on the dynamics of the system. Chaos theory is based on the observation that simple rules when iterated can give rise to complex behaviour according to the following equation.

$$X_{N+1} = X_N (\bmod\ 1) where 0 \leq X_N \leq 1$$

Chaotic systems are sensitive to the control parameters and initial conditions; Therefore, it can be connected with some cryptographic features of good cyphers, such as diffusion and balance property. When comparing chaos with other traditional methods, the ones based on chaos theory are suitable for extensive data such as images and videos. Also, the chaos-based method has achieved excellent performance, and it is recommended for many cryptosystems. A chaotic system is considered as a symmetric block cipher. There are two methods of chaotic systems: analogue and digital. A chaotic digital system has a significant concern in the digital world [28, 29]. In this paper, a matrix element M1Xi is encrypted in every round as follows:

$$C_{1Xi} = M_{1Xi} XOR (X\tilde{f} \bmod\ 256).$$

One of the commonly used maps in chaos theory is the logistic map as described below.

$$X_{n+1} = rX_n \ (1 - X_n).$$

Where the parameter r belongs to the interval [0, 4] and determines the mapping behaviour, while n is the iteration number that determines the time.

The significant advantage of a chaotic system over a noisy one is that the chaotic system is deterministic; Therefore, the knowledge of system parameters and initial conditions enables one to recover a message [21].

Confusion and diffusion are related to the fundamental characteristics of chaos theory, and any strong cryptosystem should consider features of chaos or pseudorandomness. Chaotic synchronisation is a type of chaotic systems.

Analogue implementation is an excellent advantage of chaotic synchronisation schemes. Chaotic communication offers the advantage of message waveform encryption without a need to digitalise it [7].

The following equation expresses confusion and diffusion processes

$$R = D^{\alpha}(C^{\beta}(P, K_C), K_D).$$

Where P and R are respectively plain text and cypher text, C and D are the confusion and diffusion functions, K_C and K_D are the confusion and diffusion keys, and α and β are numbers of rounds for total encryption and confusion, respectively. The chaotic map uses parameters as keys to providing high security.

1.2 Classification

Classification is the process of categorising data based on different classes [17]. One of the main classification techniques is a K-Nearest Neighbour (KNN). In this paper, we applied classification by KNN as it has high accuracy at K = 3, as mentioned in Section 5. Classification techniques can be parametric, semiparametric and non-parametric. For classification, a useful technique can be used to assign a weight to the contributions of the neighbours, so that the nearer neighbours contribute more to the average than the more distant ones [24].

K-nearest neighbour algorithm (k-NN) is a non-parametric method used for classification. The input consists of the k closest training examples in the feature space. KNN uses Euclidean distance to calculate the distance between two points of test data and training data [13]. The training examples are vectors in a multidimensional feature space, each with a class label. The training phase of KNN consists of storing the feature vectors and class labels of the training samples. K-nearest neighbour is considered as a type of instance-based learning, where the function is only approximated locally, and all computation is deferred until classification. It is easy to implement and apply for training data. KNN is good against noisy training data and is efficient if the training data is astronomical.

The rest of this paper is organised as follows. In the next section, we give a brief review of some related work. In Section 3, we introduce our proposed method. In Section 4, we give evaluation matrices. In Section 5, we give results. In Section 6, we discuss our results. Finally, we present our conclusions.

2 Related Work

A lot of different approaches proposed recently focusing on the challenges of security issues on cloud computing by using different encryption techniques. Some of these methods only use a single encryption techniques methods and other used hybrid encryption. In [6] uses FHE algorithm as the encryption is performed on the ciphertext. The system solves the security problem for stored data in the cloud.

Encrypted the data by a key is proposed in [46] that is not available for the provider. It based on the idea of manual classification and addressed data confidentiality problem. It compared with AES 128 and AES 256 with SHA 2. The results show that it achieved less processing time when compared with AES 128 and AES 256. While in [39], a model depends on simple key generation by an arbitrary matrix is proposed.

In [15] proposed a framework using fast RSA to provide security to the data in the cloud. This algorithm increases the speed up time for encryption and decryption when compared with RSA.

A hybrid cryptography algorithm is proposed in [25] that uses AES for file uploads and file download. AES key is encrypted using the RSA algorithm. In [41], the authors combine the DES algorithm, followed by a CAST encryption algorithm to achieve data protection.

In [5] applies Blowfish with a different number of rounds to achieve better security and reduce hacking while in [51] applied the ElGamal algorithm to enhance cloud security and allows encrypting ciphertext in two levels. [42] presents a new security framework for achieving data security. Data is split into blocks of bits. Genetic algorithm is applied to every two blocks of bits. The final output of every genetic algorithm is a cypher text, which is also two blocks of bits. Each cypher text is stored on the cloud at a distinct location. In [22] applies setup, keygen, encrypt and decrypt algorithms to perform encryption operations on ciphertext using the private key and public key. It applies two-party computation 2PC protocols between Key Generation Center and data storing centre to ensure security.

All mentioned methods used a single algorithm and manual classification to deal with security issues. However, we applied a hybrid encryption algorithm and classifier such as the K-Nearest Neighbor. Table 1 represents a summary of related work.

3 Proposed Method

Our proposed method is based on chaotic map and classification. Chaotic map depends on chaos theory. The chaotic map can generate values of low cost with simple iterations, which makes it suitable for the construction of stream ciphers. Therefore, cryptosystem can provide a fast and secure means for data encryption.

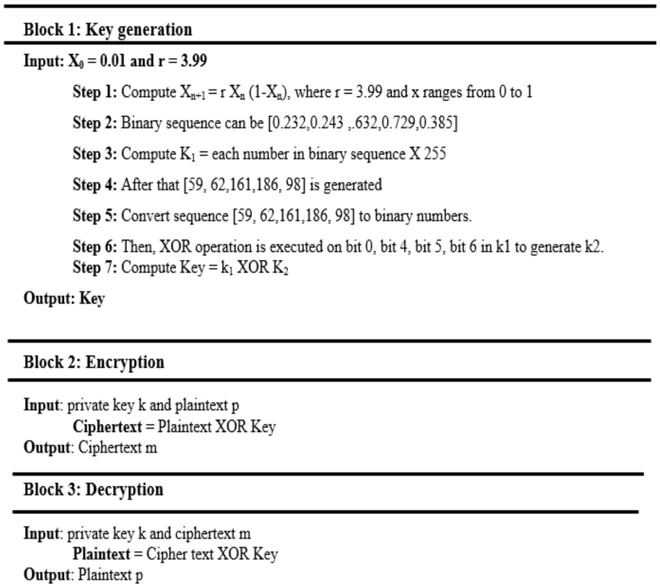
Table 1: Comparison between different security frameworks

	[6]	[46]	[10]	[39]	[15]	[38]	[25]
Parameter	FHE	Multi-cloud	Secure cloud model	probabilistic encryption	Fast RSA	Proposed algorithm symmetric	Proposed model
Algorithm used	Homomorphic Encryption	RSA	AES and SHA	probabilistic encryption	Fast RSA	Symmetric algorithm	AES
Applied security on cloud	Yes	Yes	Yes	No	No	Yes	Yes
Used chaos theory	No	No	No	No	No	No	No
Used hybrid algorithm	No	No	Yes	No	No	No	No
Performance	Complexity less than CAST-128	More secure when compared to regular system	Less processing time	Less encryption time when compared to AES and DES	Less encryption time when compared to cloud RSA	Less encryption time when compared to AES	file upload has less time than a file download

	[41]	[5]	[11]	[51]	[42]	[4]	[27]	[40]
Parameter	Hybrid DES&CAST	Recursive blowfish	Homomorphic Encryption	ElGamal	New security framework	Data splitting mechanism	homomorphic token and error correcting codes	Protection model
Algorithm used	DES&CAST	Enhanced blowfish	Homomorphic Encryption	ElGamal	Genetic algorithm	AES	homomorphic token and error correcting codes	AES
Applied security on cloud	No	No	Yes	Yes	No	Yes	Yes	Yes
Used chaos theory	No	No	No	No	No	No	No	No
Used hybrid algorithm	Yes	No	No	No	No	No	No	No
Performance	High encryption time when compared to DES	More secure than standard blowfish	More secure	More secure	More secure and efficient	Safer than similar methods	Safer	More secure

3.1 The Proposed Chaotic Encryption Algorithm (Chaotic NHCP)

The following sub-section illustrated the basic steps for key generation, encryption and decryption methods as essential building blocks for the proposed algorithm.



```
Block 1: Key generation
Input: $X_0 = 0.01$ and $r = 3.99$
        Step 1: Compute $X_{n+1} = r X_n (1-X_n)$, where $r = 3.99$ and x ranges from 0 to 1
        Step 2: Binary sequence can be [0.232,0.243 ,.632,0.729,0.385]
        Step 3: Compute $K_1$ = each number in binary sequence X 255
        Step 4: After that [59, 62,161,186, 98] is generated
        Step 5: Convert sequence [59, 62,161,186, 98] to binary numbers.
        Step 6: Then, XOR operation is executed on bit 0, bit 4, bit 5, bit 6 in k1 to generate k2.
        Step 7: Compute Key = $k_1$ XOR $K_2$
Output: Key

Block 2: Encryption
Input: private key k and plaintext p
        Ciphertext = Plaintext XOR Key
Output: Ciphertext m

Block 3: Decryption
Input: private key k and ciphertext m
        Plaintext = Cipher text XOR Key
Output: Plaintext p
```

3.2 Classification

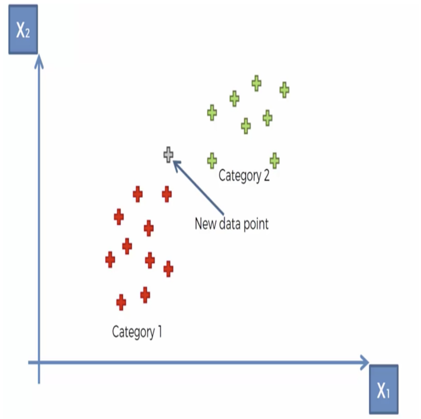


Figure 3: K-nearest neighbor

In this paper, we deal with the impossibility of encrypting all data without taking into account its confidentiality degree. So, we encrypt data based on the degree of confidentiality. We can take into consideration the degree of confidentiality in classifying data for saving the processing time. We applied classification by K-Nearest Neighbor (KNN). We classified data as highly sensitive or less sensitive. The output KNN is a class membership. An object is classified by a maximum value of its neighbours, with the object being assigned to the class with most common among its k nearest neighbours. If $k =\sim 1$, then the object is assigned to the class of the single nearest neighbour. KNN is illustrated in Figure 3.

3.3 Building Hybrid Cryptography Algorithms (NHCP)

After the classification process, we applied a hybrid algorithm which combines both Fast RSA and Blowfish cipher algorithm. The goal of the hybrid algorithm is to encrypt data efficiently, and this can reduce encryption time. Two encryption algorithms were implemented in the hybrid cryptography algorithm. These algorithms are implemented to improve the efficiency of encryption algorithm security and processing time. Hybrid encryption algorithm provides security since it encrypts data by two algorithms. It offers the advantage of reducing encryption time as FastRSA is an asymmetric algorithm and Blowfish is a symmetric one. By this way, data size is reduced to half. Figure 4 shows the encryption process for the hybrid algorithm as below.

1) 32-bit plaintext is divided into plaintext1 and plaintext 2;

2) FastRSA is used to encrypt plaintext1 generating ciphertext1;

3) Blowfish is used to encrypt plaintext2 generating ciphertext2;

4) Ciphertext1 and ciphertext2 are combined into 32-bit ciphertext.

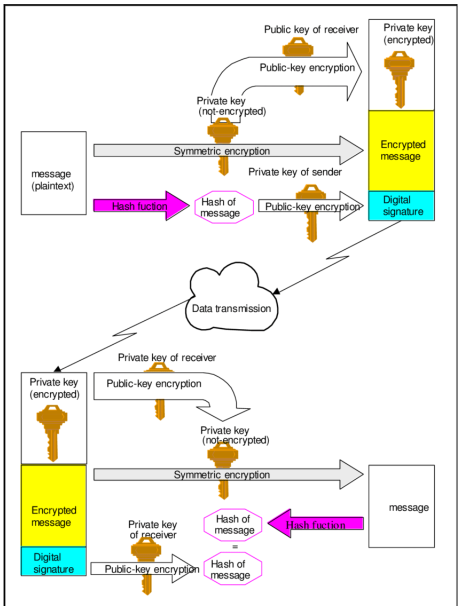


Figure 4: Hybrid algorithm using Fast RSA and blowfish

3.3.1 Proposed Encryption Algorithm

Input: M (Plain text), k(secret key of FastRSA encryption), s(32 bit size of block).

Output: C (Cipher text), ci (encrypted text using FastRSA), Ci (encrypted text using Blowfish).

```
$n = M/s$;
let $i = 0$;
do{
$m = \sum_{i=0}^{i=\frac{n}{2}-1}(Bi)$ the first part of plain text;
for $(j = 0; j <= n - I; j++)$
$c_i = E_{\text{FastRSA}}\ (K_j, B_i)$
$i++$;
}
while $(i < n/2)$;
$i = (n/2)$
let $K$ be a private key of Blowfish
do {
$M = \sum_{i=n/2}^{i=n}(Bi)$ the second part of plain text which
    encrypted simultaneously with the first part ;
$C_i = E_{Blowfish}\,(K_j, B_i)$
$i++$;
}
while $(i < n)$
$C = c_i + C_i$
```

Where n is a number of blocks, i is a counting number, ($\boldsymbol{K}$) is Private key of Blowfish for the encryption process.

4 Evaluation Metrics

In order to evaluate the proposed algorithm, some performance metrics are used such as encryption time, throughput, battery power and Accuracy.

- The encryption time is considered the time that an encryption algorithm takes to produce a ciphertext from a plaintext. Encryption time is used to calculate the throughput of an encryption scheme. It indicates the speed of encryption.

- The throughput of the encryption scheme is calculated as in Equation (1).

$$\text{Throughput} = \frac{T_p}{E_t} \qquad (1)$$

Where T_p : total plain text bytes) and E_t : encryption time (second).

The CPU process time is the time that a CPU is committed only to the particular process of calculations. It reflects the load of the CPU.

The CPU clock cycles are a metric, reflecting the energy consumption of the CPU while operating on encryption operations. Each cycle of CPU will consume a small amount of energy.

Measurement of Energy Consumption.

Energy consumption of security primitives can be measured in many ways. The used method can be measured by counting the number of computing cycles which are used in computations related to cryptographic operations. For the computation of the energy cost of encryption, we use the same techniques as described in the following equations.

$$\mathbf{B}_{\text{cost encryption}}\ \text{(ampere-cycle)} = \boldsymbol{\tau} * \mathbf{I}$$

$$\text{T}_{\text{energy cost}(ampere-seconds)} = \frac{B_{costencryption}(ampere-cycle)}{F(cycles/sec)}$$

$$\text{E}_{\text{cost}}\ \text{(Joule)} = \text{T}_{\text{energy cost}}(ampere-seconds) * V$$

Where

- B_cost_encryption: A basic cost of encryption (ampere-cycle).
- τ: The total number of clock cycles.
- I: The average current drawn by each CPU clock cycle.
- Tenergy_cost: The total energy cost (ampere-seconds).
- F: Clock frequency (cycles/sec).
- E_cost (Joule): The energy cost (consumed).

By using the cycles, the operating voltage of the CPU, and the average current drawn for each cycle, we can calculate the energy consumption of cryptographic functions. For example, on average, each cycle consumes approximately 270 mA on an Intel 486DX2 processor [34] or 180 mA on Intel StrongARM [43]. For a sample calculation, with a 700 MHz CPU operating at 1.35 Volt, encryption with 20,000 cycles would consume about 5.71 x 10-3 mA-second or 7.7 μ Joule. So, the amount of energy consumed by program P to achieve its goal (encryption or decryption) is given by

$$\mathbf{E} = \mathbf{V_{cc}} \times \boldsymbol{I} \times \boldsymbol{N} \times \boldsymbol{\tau}.$$

Where N: The number of clock cycles, τ: the clock period. $\boldsymbol{V}_{CC}$: The supply voltage of the system, I: The average current in amperes drawn from the power source for T seconds.

Since for a given hardware, both $\boldsymbol{V}_{CC}$ and τ are fixed $E \propto I \times N$. However, at the application level, it is more meaningful to talk about T than N, and therefore, we express energy as $E \propto I \times T$. Since for a given hardware V_{cc} are fixed [35].

Accuracy is one of the measures for evaluating classification models. Accuracy is the fraction of predictions our model got right. Accuracy=Number of correct predictions / Total number of predictions (2). Accuracy: It measures the correctness according to the following

$$\text{Accuracy} = (\mathbf{TP} + \mathbf{TN})/(\mathbf{TP} + \mathbf{N} + \mathbf{FP} + \mathbf{FN}). \qquad (2)$$

Where TP = True Positives, TN = True Negatives, FP = False Positives, and FN = False Negatives.

4.1 Experiments and Performance Analysis

Algorithms are implemented using Python programming language in Windows-10, the 64-bit operating system on a 2.20 GHz processor using 8GB RAM to analyse their performance. A twenty-two text different file size ranges from 8 KB to 15 MB.The twenty-two text files of different sizes are used to carry out the experiment, where we evaluate the performance of different algorithms AES, DES, chaotic and hybrid algorithm. The experiments are conducted on the test system. These implementations are thoroughly tested and are optimised to give the maximum performance for each algorithm. The performance of these algorithms is evaluated based on parameters like encryption time, throughput and power consumption.

The size of the ciphertext. Table 2 describes the output of the encryption process. It shows the size of the ciphertext in bytes.

Table 2: Size of cipher text (bytes)

Hybrid(Fast RSA+ Blowfish)	Chaotic	DES	AES	File size in KB
12	8	6	8	8
20	16	14	16	16
36	32	30	32	32
52	48	46	48	48
68	64	62	64	64
84	80	78	80	80
104	100	98	100	100
204	200	198	200	200
304	300	298	300	300
404	400	398	400	400
504	500	498	500	500
604	600	598	600	600
804	800	798	800	800
1.2 MB	1 MB	0.8 MB	1 MB	1 MB
2.2	2	1.8	2	2
3.2	3	2.8	3	3
5.2	5	4.8	5	5
7.2	7	6.8	7	7
9.2	9	8.8	9	9
11.2	11	10.8	11	11
13.2	13	12.8	13	13
15.2	15	14.8	15	15

Time of encryption and decryption processes. The encryption time is the time that an encryption algorithm takes to produce a ciphertext from a plaintext. The decryption time is the time that a decryption algorithm takes to produce a plaintext from a ciphertext.

Table 3 and Figure 5 show the time of the encryption process for different sizes of plain text. It is shown that proposed hybrid cryptography protocol based on chaotic map (Chaotic NHCP) achieve the least time for encryption followed by Hybrid between Fast RSA and Blowfish (NHCP). Table 4 and Figure 6 show the time of decryption process for different sizes of plain text. As in the encryption, it is clear that Chaotic NHCP achieve the least time for decryption followed by (NHCP).

Table 3: Encryption time (seconds) of cryptographic algorithms

Hybrid(Fast RSA+ Blowfish)	Chaotic	DES	AES	File size in KB
0.01	0.001	.04	.1	8
0.02	0.002	.055	.2	16
0.03	0.003	.07	.23	32
0.04	0.004	0.11	.26	48
0.05	0.005	0.13	0.33	64
0.06	0.006	0.15	0.41	80
0.07	0.008	0.17	0.5	100
0.13	0.01	0.27	0.6	200
0.19	0.012	0.37	0.7	300
0.25	0.014	0.47	0.8	400
0.31	0.016	0.57	0.9	500
0.37	0.018	0.66	1	600
0.5	0.025	0.8	1.15	800
0.6	0.043	0.9	1.3	1 MB
1.2	0.08	1.8	2.1	2
1.8	0.13	2.6	3	3
3	0.23	4.2	4.8	5
4.2	0.33	5.8	6.6	7
5.4	0.43	7.4	8.4	9
6.6	0.53	9	10.2	11
7.8	0.63	10.6	12	13
9	0.73	12.2	13.8	15

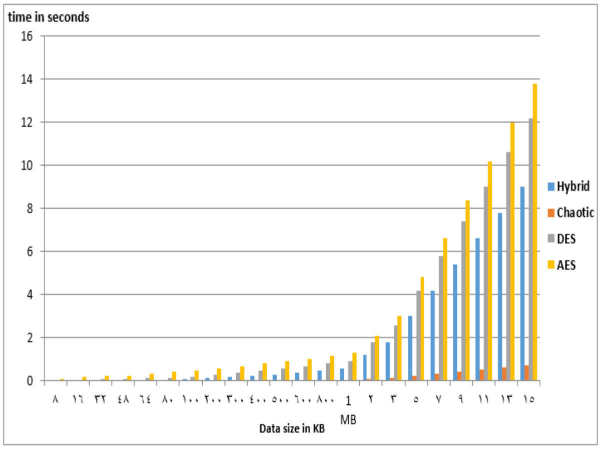


Figure 5: Encryption time of cryptographic algorithms

So it can be concluded from Table 3, Table 4, Figure 5, and Figure 6 that Chaotic NHCP and NHCP has encryption time and decryption time less than AES and DES. Chaotic NHCP has the least encryption time and decryption time.

Throughput. Encryption time is used to calculate the

Table 4: Decryption time (seconds) of cryptographic algorithms

Hybrid(Fast RSA+ Blowfish)	Chaotic	DES	AES	File size in KB
0.009	0.0009	.02	.07	8
0.01	0.001	.035	.17	16
0.02	0.002	.05	.2	32
0.03	0.003	0.09	.23	48
0.04	0.004	0.11	0.3	64
0.05	0.005	0.13	0.38	80
0.06	0.007	0.15	0.47	100
0.12	0.009	0.25	0.57	200
0.18	0.011	0.35	0.67	300
0.24	0.013	0.45	0.77	400
0.3	0.015	0.55	0.87	500
0.36	0.017	0.64	.89	600
0.49	0.024	0.78	.92	800
0.59	0.042	0.88	1	1 MB
1.19	0.07	1.78	1.8	2
1.79	0.12	2.58	2.7	3
2.99	0.22	4.18	4.5	5
4.19	0.32	5.78	6.3	7
5.39	0.42	7.38	8.1	9
6.58	0.52	8.98	9.9	11
7.78	0.62	10.58	11.7	13
8.98	0.72	12.18	13.5	15

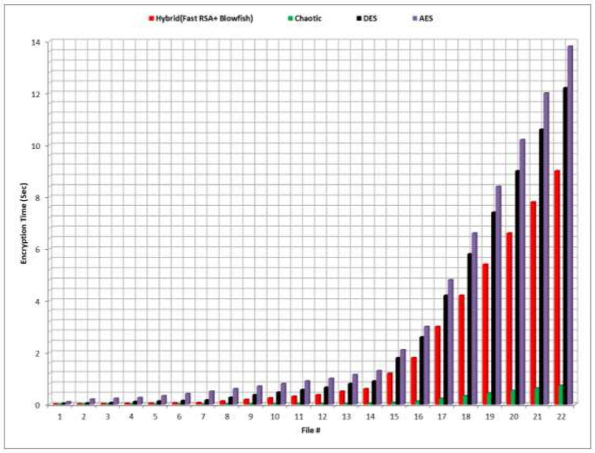


Figure 6: Decryption time of cryptographic algorithms

throughput of an encryption scheme. It indicates the speed of encryption. Table 5, and Figure 7 show that the encryption throughput of the proposed hybrid cryptography algorithm based on chaotic map (Chaotic NHCP) is more significant than other algorithms for different sizes of plain text. It is shown that both (Chaotic NHCP) and NHCP achieve the most significant values.

Table 5: Encryption throughput (KB/second) of cryptographic algorithms

File size in KB	AES	DES	Chaotic	Hybrid(Fast RSA+ Blowfish)
8	80	200	8000	800
16	80	290.91	8000	800
32	139.13	457.14	10666.67	1066.67
48	184.62	436.36	12000	1200
64	193.94	492.31	12800	1280
80	195.12	533.33	13333.33	1333.33
100	200	588.24	12500	1428.57
200	333.33	740.74	20000	1538.46
300	428.57	810.81	25000	1578.95
400	500	851.06	28571.43	1600
500	555.56	877.19	31250	1612.9
600	600	909.09	33333.33	1621.62
800	695.65	1000	32000	1600
1 MB	787.69	1137.78	23813.95	1706.67
2	975.24	1137.78	25600	1706.67
3	1024	1181.54	23630.77	1706.67
5	1066.67	1219.05	22260.87	1706.67
7	1086.06	1235.86	21721.21	1706.67
9	1097.14	1245.41	21432.56	1706.67
11	1104.31	1251.56	21252.83	1706.67
13	1109.33	1255.85	21130.16	1706.67
15	1113.04	1259.02	21041.1	1706.67

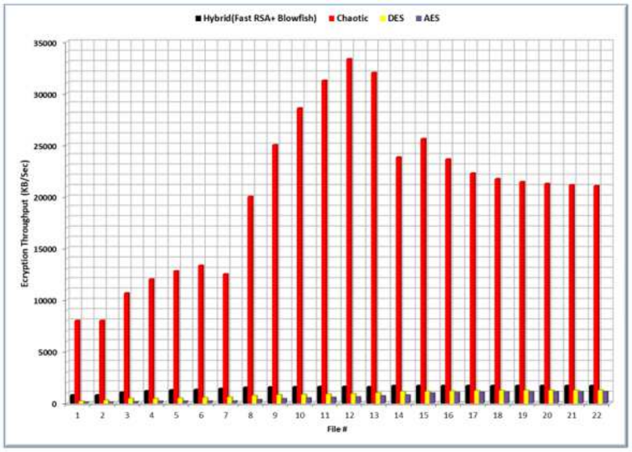


Figure 7: Encryption throughput of cryptographic algorithms(KB/Sec)

Table 6, and Figure 8 also show that the decryption throughput of the proposed hybrid cryptography algorithm based on chaotic map (Chaotic NHCP) is more significant than other algorithms for different sizes of plain text. It is shown that both (Chaotic NHCP) and NHCP achieve the most significant values.

Table 6: Decryption throughput (KB/Second) of cryptographic algorithms

File size in KB	AES	DES	Chaotic	Hybrid(Fast RSA+ Blowfish)
8	114.29	400	8888.89	888.89
16	94.12	457.14	16000	1600
32	160	640	16000	1600
48	208.7	533.33	16000	1600
64	213.33	581.82	16000	1600
80	210.53	615.38	16000	1600
100	212.77	666.67	14285.71	1666.67
200	350.88	800	22222.22	1666.67
300	447.76	857.14	27272.73	1666.67
400	519.48	888.89	30769.23	1666.67
500	574.71	909.09	33333.33	1666.67
600	674.16	937.5	35294.12	1666.67
800	869.57	1025.64	33333.33	1632.65
1 MB	1024	1163.64	24380.95	1735.59
2	1137.78	1150.56	29257.14	1721.01
3	1137.78	1190.7	25600	1716.2
5	1137.78	1224.88	23272.73	1712.37
7	1137.78	1240.14	22400	1710.74
9	1137.78	1248.78	21942.86	1709.83
11	1137.78	1254.34	21661.54	1711.85
13	1137.78	1258.22	21470.97	1711.05
15	1137.78	1261.08	21333.33	1710.47

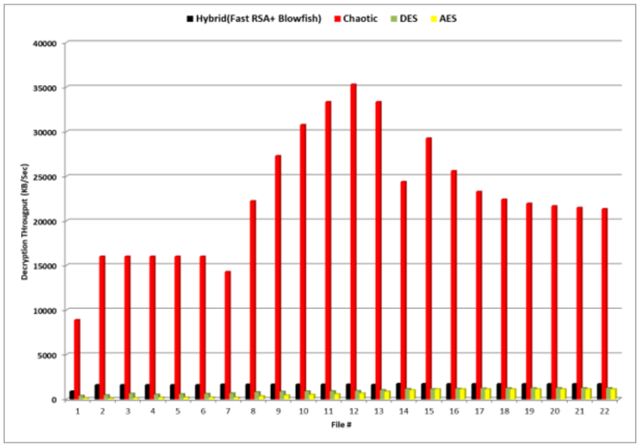


Figure 8: Decryption throughput (KB/Second) of cryptographic algorithms

Power consumption. It is noticed from Table 7, and Figure 9 chaotic NHCP, and NHCP has the least power consumption.

Accuracy of KNN depends on the value of k; in our case, $K = 3$.KNN with $K = 1$ gives better results and

Table 7: Power consumption (watt) for encryption of different cryptographic algorithms

File size in KB	AES	DES	Chaotic	Hybrid(Fast RSA+ Blowfish)
8	0.66	0.264	0.0066	0.066
16	1.32	0.363	0.0132	0.132
32	1.5	.462	0.02	0.2
48	1.7	0.7	0.03	0.3
64	2.18	0.86	0.033	0.33
80	2.7	1	0.04	0.4
100	3.3	1.12	0.053	0.46
200	3.96	1.8	0.066	0.86
300	4.6	2.4	0.08	1.25
400	5.28	3.1	0.1	1.65
500	6	3.76	0.106	2
600	6.6	4.36	0.12	2.44
800	7.6	5.28	0.17	3.3
1 MB	8.6	6	0.28	4
2	12.86	11.88	0.53	8
3	20	17.16	0.86	11.88
5	31.68	27.72	1.518	19.8
7	43.56	38.28	2.178	27.72
9	55.44	48.84	2.838	35.64
11	67.32	59.4	3.498	43.56
13	79.2	69.96	4.158	51.48
15	91.08	80.52	4.818	59.4

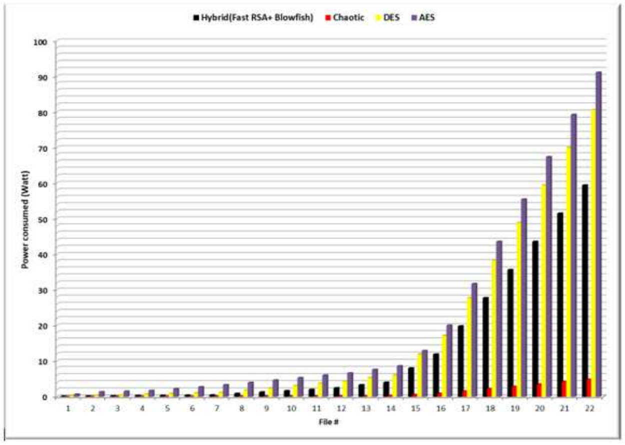


Figure 9: Power consumption (watt) of cryptographic algorithms

accuracy. KNN requires that classes can be separable to provide excellent results.

Results analysis. The results show the superiority of (Chaotic NHCP) algorithm over other algorithms in terms of the power consumption, processing time, and throughput followed by NHCP in case of encryption and decryption -(when the same data is encrypted by using DES and AES. it is found that NHCP requires approximately 60% of the time used for encryption which is consumed for AES and 71% in case of compared by DES). Another point can be noticed that Chaotic NHCP requires approximately 5% of the time used for encryption, which is consumed for AES and 6% in the case of comparing by DES).

In the case of decryption, the results also show the superiority of (Chaotic NHCP) algorithm over other algorithms in terms of decryption time. It is found that NHCP requires approximately 62.7% of the time used for encryption, which is consumed for AES and 71.5% in case of compared by DES). Another point can be noticed that Chaotic NHCP requires approximately 4.8% of the time used for encryption, which is consumed for AES and 5.4% in the case of comparing by DES).

In the case of power consumption for encryption, the results also show the superiority of (Chaotic NHCP) algorithm over other algorithms in terms of power consumption. It is found that NHCP requires approximately 60.12% of the time used for encryption, which is consumed for AES and 71.35% in case of compared by DES). Another point can be noticed that Chaotic NHCP requires approximately 4.7% of the time used for encryption, which is consumed for AES and 5.58% in case of compared by DES).

In case of power consumption for decryption, the results also show the superiority of (Chaotic NHCP) algorithm over other algorithms in terms of decryption time. It is found that NHCP requires approximately 61.4% of the time used for encryption, which is consumed for AES and 70.92% in case of compared by DES). Another point can be noticed that Chaotic NHCP requires approximately 4.6% of the time used for encryption, which is consumed for AES and 5.48% in the case of comparing by DES). Finally, It is shown from experimental results that the chaotic encryption algorithm is the fastest algorithm among other cryptographic algorithms.

The chaotic map has the least encryption time as it depends on simple operations like XOR, multiplication and logistic function. It uses a logistic function to generate random values that are used to produce key k. The encryption algorithm that has the least encryption time is the best algorithm. It can have the most value of throughput, and the least value of power consumption Nearest Neighbor (KNN) classifier has high accuracy as it has 83% when $K = 3$, as shown in Figure 11. Both AES and DES use 16 rounds with XOR operation, and this leads to high encryption time. The hybrid algorithm has encryption time less than AES and DES. It merges both FastRSA and Blowfish. FastRSA uses a modulus of the form N=prqs, so it has less encryption time. On the other hand, blowfish uses F function with 16 rounds.

5 Conclusion

In this paper, a novel secured, the optimised framework is proposed to improve the efficiency of security of the data to the cloud. This framework design an encryption method based on chaotic theory (chaotic NHCP) that reduces the encryption time and ensures confidentiality through data classification and a hybrid cryptographic algorithm (NHCP) that merges fast RSA and Blowfish cryptographic algorithms. This study presents a performance evaluation of selected encryption algorithms on power consumption to be used to provide security for the cloud environment. The selected algorithms are AES, DES, NHCP, and chaotic NHCP. Several points can be concluded from the experimental results. The experiment with these parameters, such as encryption time, throughput, and power consumption, is done, and those results show that chaotic NHCP has better performance to other cryptographic algorithms. Performance evaluation of selected this study presents a performance evaluation of selected encryption algorithms on power consumption to be used to provide security for the cloud environment. The selected algorithms are AES, DES, NHCP, and chaotic NHCP. Several points can be concluded from the experimental results. The experiment with these parameters, such as encryption time, throughput, and power consumption, is done, and those results show that chaotic NHCP has better performance to other cryptographic algorithms. Performance evaluation of selected encryption algorithms. Encryption algorithms. As shown in results, the chaotic map has the least encryption time; the hybrid algorithm has encryption time less than AES and DES. The data classification helps in decreasing the time of encrypting stored data. It is noticed from experimental results that K Nearest Neighbor (KNN) has high accuracy in the classification process. By comparing the result of this method with other cryptographic methods, we can recommend the implemented chaotic method to be used in securing data through cloud computing. We found that chaotic NHCP has better performance than other encryption algorithms, followed by NHCP in case of encryption time, throughput, and power consumption for encryption and decryption. Chaotic NHCP and NHCP are faster than DES, and AES. NHCP encrypts and decrypts data faster than DES and AES. Chaotic NHCP is faster than NHCP. These results are the same in encryption and decryption process with different packet size. So the chaotic NHCP and NHCP is sufficient to provide security on cloud computing.

References

[1] D. S. AbdElminaam, "Improving the security of cloud computing by building new hybrid cryptography algorithms," *International Journal of Electronics and Information Engineering*, vol. 8, no. 1, pp. 40–48, 2018.

[2] M. Aledhari, A. Marhoon, A. Hamad, and F. Saeed, "A new cryptography algorithm to protect cloud-based healthcare services," in *IEEE/ACM International Conference on Connected Health: Applications, Systems and Engineering Technologies (CHASE'17)*, pp. 37–43, 2017.

[3] R. Arora, A. Parashar, "Secure user data in cloud computing using encryption algorithms," *International Journal of Engineering Research and Applications*, vol. 3, no. 4, pp. 1922–1926, 2013.

[4] V. R. Balasaraswathi and S. Manikandan, "Enhanced security for multi-cloud storage using cryptographic data splitting with dynamic approach," in *IEEE International Conference on Advanced Communications, Control and Computing Technologies*, pp. 1190–1194, 2014.

[5] N. Balkish, A. M. Prasad, and V. Suma, "An efficient approach to enhance data security in cloud using recursive blowfish algorithm," in *ICT and Critical Infrastructure: Proceedings of the 48th Annual Convention of Computer Society of India-Vol I*, pp. 575–582, 2014.

[6] P. V. Bharati and T. S. Mahalakshmi, "Data storage security in cloud using a functional encryption algorithm," in *Emerging Research in Computing, Information, Communication and Applications*, pp. 201–212, 2016.

[7] M. Boumaraf and F. Merazka, "Speech encryption based on hybrid chaotic key generator for amr-wb g. 722.2 codec," in *The 12th International Conference for Internet Technology and Secured Transactions (ICITST'17)*, pp. 87–91, 2017.

[8] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. F. D. Rose, and R. Buyya, "Cloudsim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," *Software: Practice and Experience*, vol. 41, no. 1, pp. 23–50, 2011.

[9] F. Dachselt and W. Schwarz, "Chaos and cryptography," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 48, no. 12, pp. 1498–1509, 2001.

[10] N. S. Darwazeh, R. S. Al-Qassas, F. AlDosari, *et al.*, "A secure cloud computing model based on data classification," *Procedia Computer Science*, vol. 52, pp. 1153–1158, 2015.

[11] C. A. Dhote, "Homomorphic encryption for security of cloud data," *Procedia Computer Science*, vol. 79, pp. 175–181, 2016.

[12] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, "A survey of mobile cloud computing: Architecture, applications, and approaches," *Wireless Communications and Mobile Computing*, vol. 13, no. 18, pp. 1587–1611, 2013.

[13] S. E. El-Khamy, H. A. Elsayed, and M. M. Rizk, "Classification of multi-user chirp modulation signals using higher order cumulant features and four types of classifiers," in *The 28th National Radio Science Conference (NRSC'11)*, pp. 1–10, 2011.

[14] K. El-Makkaoui, A. Beni-Hssane, A. Ezzati, and A. El-Ansari, "Fast cloud-rsa scheme for promoting data confidentiality in the cloud computing," *Procedia Computer Science*, vol. 113, pp. 33–40, 2017.

[15] K. El-Makkaoui, A. Beni-Hssane, A. Ezzati, and A. El-Ansari, "Fast cloud-rsa scheme for promoting data confidentiality in the cloud computing," *Procedia Computer Science*, vol. 113, pp. 33–40, 2017.

[16] D. S. A. Elminaam, H. M. Abdual-Kader, and M. M. Hadhoud, "Evaluating the performance of symmetric encryption algorithms," *International Journal Network Security*, vol. 10, no. 3, pp. 216–222, 2010.

[17] A. Freier, P. Karlton, and P. Kocher, *The secure sockets layer (ssl) protocol version 3.0*, RFC 6101, 2011.

[18] K. Hansen, T. Larsen, and K. Olsen, "On the efficiency of fast RSA variants in modern mobile phones," *International Journal of Computer Science and Information Security*, vol. 6, no. 3, 2009.

[19] X. He, A. Machanavajjhala, and B. Ding, "Blowfish privacy: Tuning privacy-utility trade-offs using policies," in *Proceedings of the ACM SIGMOD international conference on Management of data*, pp. 1447–1458, 2014.

[20] W. F. Hsien, C. C. Yang and M. S. Hwang, "A survey of public auditing for secure data storage in cloud computing," *International Journal of Network Security*, vol. 18, no. 1, pp. 133-142, 2016.

[21] M. Hu, H. Gao, and T. Gao, "Secure and efficient ranked keyword search over outsourced cloud data by chaos based arithmetic coding and confusion," *International Journal Network Security*, vol. 21, no. 1, pp. 105–114, 2019.

[22] J. Hur, "Improving security and efficiency in attribute-based data sharing," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 10, pp. 2271–2282, 2011.

[23] M. S. Hwang, C. C. Lee, Y. C. Lai, "Traceability on RSA-based partially signature with low computation", *Applied Mathematics and Computation*, vol. 145, no. 2-3, pp. 465–468, Dec. 2003.

[24] J. P. Kandhasamy and S. Balamurali, "Performance analysis of classifier models to predict diabetes mellitus," *Procedia Computer Science*, vol. 47, pp. 45–51, 2015.

[25] Z. Kartit, A. Azougaghe, H. K. Idrissi, M. El-Marraki, M. Hedabou, M. Belkasmi, and A. Kartit, "Applying encryption algorithm for data security in cloud storage," in *International Symposium on Ubiquitous Networking*, pp. 141–154, 2015.

[26] A. Kumar and M. K. Ghose, "Overview of information security using genetic algorithm and chaos," *Information Security Journal: A Global Perspective*, vol. 18, no. 6, pp. 306–315, 2009.

[27] S. P. Kumar and R. Subramanian, "An efficient and secure protocol for ensuring data storage security in cloud computing," *International Journal of Computer Science Issues (IJCSI'11)*, vol. 8, no. 6, p. 261, 2011.

[28] C. Li, G. Luo, and C. Li, "A novel scheme for the preview of the image encryption based on chaotic ikeda map," *International Journal Network Security*, vol. 20, no. 6, pp. 1105–1114, 2018.

[29] C. Li, G. Luo, and C. Li, "An image encryption scheme based on the three-dimensional chaotic logistic map," *International Journal Network Security*, vol. 21, no. 1, pp. 22–29, 2019.

[30] C. Lu, A. L. M. dos Santos, and F. R. Pimentel, "Implementation of fast rsa key generation on smart cards," in *Proceedings of the ACM symposium on Applied computing*, pp. 214–220, 2002.

[31] H. Ma, X. Han, T. Peng, and L. Zhang, "Secure and efficient cloud data deduplication supporting dynamic data public auditing," *International Journal Network Security*, vol. 20, no. 6, pp. 1074–1084, 2018.

[32] P. C. Mandal, "Superiority of blowfish algorithm," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 2, no. 9, 2012.

[33] D. S. A. Minaam, H. M. Abdual-Kader, and M. M. Hadhoud, "Evaluating the effects of symmetric cryptography algorithms on power consumption for different data types.," *International Journal Network Security*, vol. 11, no. 2, pp. 78–87, 2010.

[34] K. Naik and D. S. L. Wei, "Software implementation strategies for power-conscious systems," *Mobile Networks and Applications*, vol. 6, no. 3, pp. 291–305, 2001.

[35] C. Panait and D. Dragomir, "Measuring the performance and energy consumption of aes in wireless sensor networks," in *Federated Conference on Computer Science and Information Systems (FedCSIS'15)*, pp. 1261–1266, 2015.

[36] T. S. Parker and L. O. Chua, "Chaos: A tutorial for engineers," *Proceedings of the IEEE*, vol. 75, no. 8, pp. 982–1008, 1987.

[37] P. Patil, P. Narayankar, D. G. Narayan, and S. M. Meena, "A comprehensive evaluation of cryptographic algorithms: DES, 3DES, AES, RSA and blowfish," *Procedia Computer Science*, vol. 78, pp. 617–624, 2016.

[38] V. Ponnuramu and L. Tamilselvan, "Encryption for massive data storage in cloud," in *Computational Intelligence in Data Mining-Volume 2*, pp. 27–37, 2015.

[39] P. Ratha, D. Swain, B. Paikaray, and S. Sahoo, "An optimized encryption technique using an arbitrary matrix with probabilistic encryption," *Procedia Computer Science*, vol. 57, pp. 1235–1241, 2015.

[40] A. Sachdev and Mohit Bhansali, "Enhancing cloud computing security using aes algorithm," *International Journal of Computer Applications*, vol. 67, no. 9, 2013.

[41] N. Sengupta and R. Chinnasamy, "Contriving hybrid descast algorithm for cloud security," *Procedia Computer Science*, vol. 54, pp. 47–56, 2015.

[42] S. K. S. ShaluMalla, "A new security framework for cloud data," *Procedia Computer Science*, vol. 143, pp. 765–775, 2018.

[43] A. Sinha and A. P. Chandrakasan, "Jouletrack-a web based tool for software energy profiling," in *Proceedings of the 38th Design Automation Conference*, pp. 220–225, 2001.

[44] W. Stallings, *Cryptography and Network Security: Principles and Practice*, 2017. ISBN 13: 978-0134444284.

[45] M. Sulochana and O. Dubey, "Preserving data confidentiality using multi-cloud architecture," *Procedia Computer Science*, vol. 50, pp. 357–362, 2015.

[46] M. Sulochana and O. Dubey, "Preserving data confidentiality using multi-cloud architecture," *Procedia Computer Science*, vol. 50, pp. 357–362, 2015.

[47] G. D. Sutter, J. P. Deschamps, and J. L. Imaña, "Modular multiplication and exponentiation architectures for fast rsa cryptosystem based on digit serial computation," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 7, pp. 3101–3109, 2010.

[48] A. A. Taha, D. S. A. Elminaam, and K. M. Hosny, "An improved security schema for mobile cloud computing using hybrid cryptographic algorithms," *Far East Journal of Electronics and Communications*, vol. 18, no. 4, 2018.

[49] J. Thakur and N. Kumar, "DES, AES and blowfish: Symmetric key cryptography algorithms simulation based performance analysis," *International Journal of Emerging Technology and Advanced Engineering*, vol. 1, no. 2, pp. 6–12, 2011.

[50] W. Voorsluys, J. Broberg, and R. Buyya, "Introduction to cloud computing," *Cloud Computing: Principles and Paradigms*, pp. 1–41, 2011.

[51] L. Xiong, Z. Xu, and Y. Xu, "A secure re-encryption scheme for data services in a cloud computing environment," *Concurrency and Computation: Practice and Experience*, vol. 27, no. 17, pp. 4573–4585, 2015.

[52] D. Zissis and D. Lekkas, "Addressing cloud computing security issues," *Future Generation Computer Systems*, vol. 28, no. 3, pp. 583–592, 2012.

[53] D. Zissis and D. Lekkas, "Addressing cloud computing security issues," *Future Generation Computer Systems*, vol. 28, no. 3, pp. 583–592, 2012.

Biography

Diaa Salama Abdul-Minaam was born on November 23, 1982, in KafrSakr, Sharkia, Egypt. He received the B.S from Faculty of Computers and Informatics, Zagazig University, Egypt in 2004 with grade very good with honor, and obtains the master degree in information system from the faculty of computers and information, menufia university, Egypt in 2009 specializing in Cryptography and network security. He obtained his Ph.D. degree in information system from the faculty of computers and information, menufia university, Egypt in 2015. He is currently a Assistance Professor in Information systems department, Faculty of Computers and Information, Benha University, Egypt since 2011. He has worked on a number of research topics..Diaa has contributed more than 40+ technical papers in the areas of wireless networks, wireless network security, Information security and Internet applications, Cloud Computing, Mobile Cloud Computing, Internet of Things, and Machine learning in international journals, international conferences, local journals and local conferences. He majors in Cryptography, Network Security, IoT, Big Data, Cloud Computing, deep learning. (Mobile: +201019511000 ; E-mail: $ds_desert@yahoo.com$)

Mostafa Abdullah Ibrahim received the B.S from Faculty of Computers and Informatics, Benha University, Egypt grade very good with honor, and register the master degree in information system from the faculty of computers and information, Benha university, Egypt.he is specializing in Cryptography and network security.

Elsayed Badr is an Associate professor of computer science at Benha Faculty of Computers & Informatics; Benha University in Egypt. He received his Ph.D. degree in Parallel Algorithms (mainly in parallel graph algorithms) in 2006 from the University of Macedonia; Greece. Dr. Badr holds a Certificate of Quality Assurance from the university of Benha, Egypt and M.Sc. in graph theory and graph algorithms applications, B.Sc. in Mathematics from Benha faculty of science in Egypt. In addition to over 8 years of teaching and academic experiences In Egypt and Greece, Dr. Badr has accumulated broad practical experiences and developed a solid set of skills in algorithms, Fuzzy theory, graph labeling, Wireless Networks, Distributed System, Parallel Programming and linear programming.