International Journal of Advances in Engineering and Management (IJAEM)
Volume 3, Issue 2 Feb 2021, pp: 435-444 www.ijaem.net ISSN: 2395-5252

New Lightweight Hybrid Cloud Computing Encryption Algorithm based on Modified SALSA20

Muhned Hussam[1] Ghassan H. Abdul-majeed[2] Haider K. Hooomod[3]
[1]Computer Science Department, College of Education, Mustansiryah University, Baghdad, Iraq
[2]Ministry of Higher Education & Science Research Baghdad, Iraq
[3]Computer Science Department, College of Education, Mustansiryah University, Baghdad, Iraq

Submitted: 02-02-2021 Revised: 18-02-2021 Accepted: 22-02-2021

ABSTRACT: Cloud computing define as massive collection of virtualized and Updateable computer resources. That have the ability to consists various application and provide multiple services that are needed by customers. It has a "pay-for-use only”, where customers pay for the cost of using the Services. But there are some companies, for example Google Inc., that give specific free storage space to users with limited services specifically in the public cloud. Cloud computing services are usually provided by an external provider that owns the infrastructure. But as more and more information’s and files on individuals and organization that putted in the cloud, concerns are beginning to grow about just how secure an environment it is. These include security of virtualization software, distributed computing, app security, management of identity, access control and authentication. Therefore an algorithm is required by which our data can be transferred speedily and securely. The main aim of this particular research is to protect the transmitted data with the help of encryption and decryption techniques. This research paper presents a model for encrypting the data transmitted through cloud. The algorithms used in this model are: new lightweight encryption algorithm and modify salsa20 algorithms. This hybrid algorithm will help users and cloud service provider to transmit their data without being stolen or affected and at high speed due to the use of lightweight algorithms. The experimental results show that the hybrid encryption algorithm has the advantages of fast encryption and decryption speed, high security, good processing ability for longer data, and can solve the data security problem in cloud.

Keywords: Cloud Security, Cloud computing, salsa20 algorithm, user authentication

I. INTRODUCTION

Cloud computing is an Internet based technique where A large number of resources (e.g., networks, storage, applications and services) are shared to the customer. Data transmitted through internet in cloud is getting larger every day Because of the multiple benefits of Cloud that helps the user to keep his data with the least effort, reasonable cost and rapidly provisioned.

In spite of all the hype surrounding the cloud, customers are still hesitant to publish their business in the cloud. Security problems in cloud computing played a major role in slowing their acceptance, Data must be secured during rest, transportation and use, and access to data must be controlled. The customer can use encryption to protect the data during the transfer, although this holds primary Cloud Providers (CP) management responsibilities. The customer can enforce access control technologies [1]. In fact, security was first classified as the biggest challenge to cloud computing. Fig.1 shows Results of International Data Corporation (IDC) survey ranking security challenges [2].

Confidentiality does not ensure protection. In addition, we must take into account the authentication function. Authenticated users can read and write cloud-based data. For authentication, user ID and password are required. There are also security vulnerabilities in cloud computing [3]. This includes a solid user authentication system for cloud computing.

In this respect, the above challenges motivate us to build a secure and powerful cloud-based user authentication system, where legitimate users demonstrate their legitimacy prior to entering the cloud. The proposed scheme verifies user authenticity using two-step. The first step is encryption, which is based on password, smartcard by using lightweight salsa20 algorithm and authentication (hash function) by using proposed system design (SASH) that provides mutual authentication, identity management, session key establishment, user privacy and secure against many popular attacks.

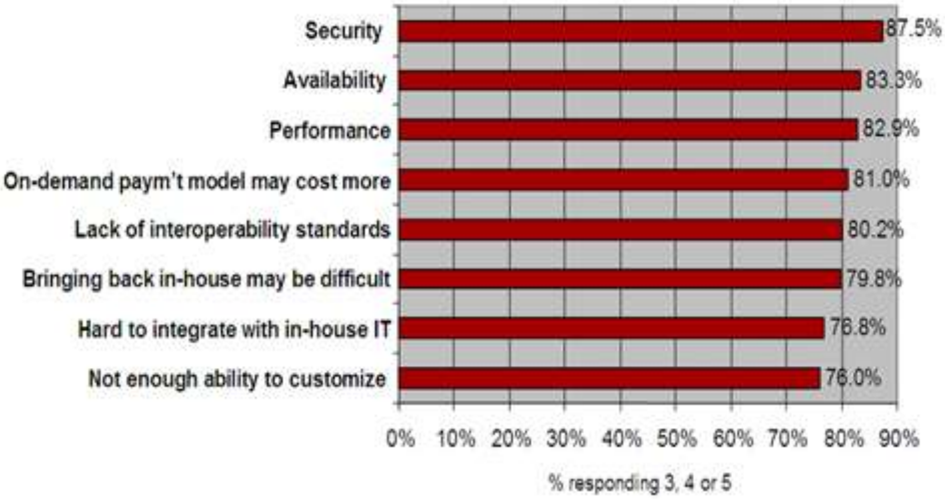


Fig.1 : IDC Survey Ranking Challenges [2]

II. CLOUD COMPUTING SERVICE MODELS

Basically, there are different models or services in cloud computing as shown in fig.2, and explained in the following [4]:

- Software as a Service (Saas): This is a overhaul model which provide accessing ability to the user‘s and allow them to use an application and services that are hosted in the cloud.
- Platform as a Service (Paas): In this service model, users deploy their applications and software in the cloud by purchasing the access to the platforms that are hosted as a service in the cloud. Hardware equipment's and communication channels are not managed by the users. Cloud platform has certain constraints within which users should implement their applications.
- Infrastructure as a Service (Iaas): The characteristics of this service are to providing, controlling and managing hardware equipment's and systems such as OS, software applications, storage device and internet connections. By using the facilities of this service, user can save his money by taking the hardware devices and other cloud services on rent on the bases of pay as per use instead of purchasing.

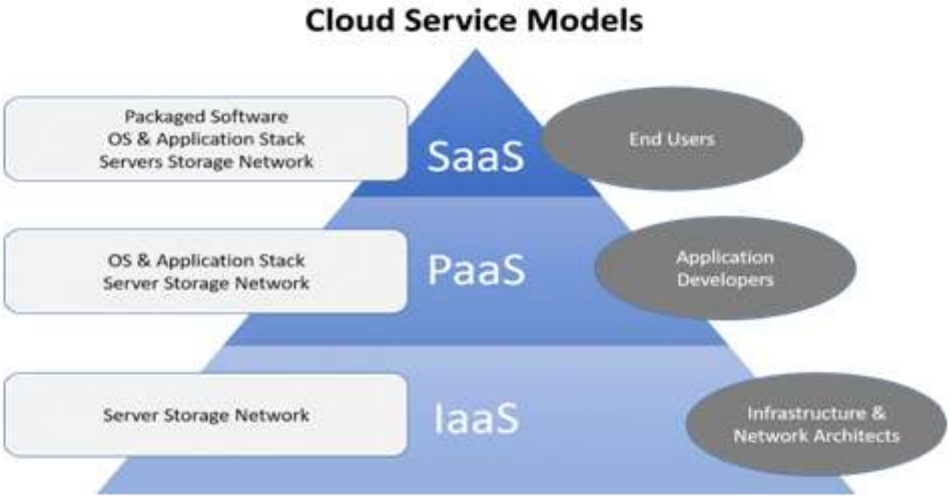


Fig.2 : Cloud computing service models [5]

III. COMPUTER CLOUD DEPLOYMENTS

Basically, there are four types of clouds [6,7], they are:

- Public cloud: It is a type of the cloud, where in cloud services are made available to the users through an intermediate called as cloud service providers via an internet. The cloud itself controls and maintains the mechanisms of the services provided to the users. Generally, these services are provided free during its free trial period and later it will be charged on the basis of pay as per use.
- Private cloud: This cloud provides many of the public cloud benefits, but only difference is the information is stored and maintained only within the organization.
- Community cloud: It is also a one kind of cloud where the information is maintained and administered by multiple organizations that have same goal. The authorization of access to the data that is resided in cloud is shared among members of organizations.
- Hybrid cloud: It is the combination of both public and private cloud. It is also known as systems of multiple clouds. These systems are interconnected in such a way that it allows users to move data and programs easily from one system to another system.

IV. RELATED WORKS

S. Hakim, M. Fouad (2017), a new secure hash algorithm is proposed for the MD5 and SHA-256 algorithms (with the final hash code length of 256) that can be used to sign applications or any message integrity since the hash code length is 512 bits [8].

Gope, Prosanta, (2019), proposed a lightweight and privacy-preserving Anonymous mutual user authentication protocol in which only the user with a trusted device has the right to access the Industrial Wireless Sensor Networks (IWSN). They considered the security of the physical layer to hold the sensor to guarantees safety even if the sensor node is captured by the opponent. The proposed protocol uses lightweight encoding alternatives, such as one-way cryptographic hash function, Physically Unclonable Function (PUF) and bitwise exclusive (XOR) operations [9].

Garima and Naveen (2014), according to this paper, introduce a method to secure cloud data using a combination of two cryptographic and steganographic algorithms. They proposed combining two cryptographic algorithms like DSA and AES (Advanced Standard Encryption) and Steganography. DSA is used for authentication and AES is used for data encryption, and Steganography is used for additional encryption. The first step is the signing of the data. The signature is first created by using a hash function on the data, providing compact data type called message digest. The message digest is then signed with the private key of the sender. After the message has been signed, the data and the signature are encrypted using AES. After the encryption with the AES algorithm has been done, the data is also encrypted with steganography. Steganography hides message along with other media which attract the intruder's attention and therefore protects the data. This complete mechanism is introduced on the ASP.NET platform and guarantees authenticity, data integrity and cloud protection. This paper concludes that the overall mechanism has a high time complexity, one by one. [10].

Jolan Rokan Naif et al (2019), a new authorization technique has been proposed based on a lightweight Bcrypt with 4D chaos system. The proposed method includes three stages (128bit-SHA1, bit-256 HMAC, modified bit-128 with Blowfish-chaos) as these three stages have been modified to be suitable with IOT devices at high speed [11].

Sanjoli and Jasmeet (2013), proposed the blend of two encryption algorithms, the Extensible Authentication Protocol (EAP-CHAP) and the Rijndael Encryption Algorithm. EAP is used to provide authenticated cloud access. For authentication purposes, CHAP, an EAP process, is implemented. The encryption is then followed by Rijndael Encryption Algorithm. The entire technique needs a few steps. In the first step, the user will send an authentication request to the Cloud Service Provider (CSP). In the second stage, after verifying the user identity by EAP-CHAP, CSP sends acknowledgement. The third step is to encrypt the user using Rijndael Encryption Algorithm and upload the encrypted data to the CSP server. The data is stored on the server in encrypted form. [12].

Dilli Ravilla et al ,(2015) , a hybrid MANET protocol is being applied in Network Simulator 2 (NS2) and hashing algorithm It. it focuses on using the HMAC-SHA 256 algorithm for message authentication and data integrity. This algorithm is a trust-based system to make the network more secure by preventing Denial-of-Services (DoS) and brute force attacks [13].

Chowdhury and DasBit (2015), proposed a lightweight hash based symmetric key message authentication code. to achieving secured communication and high-speed authentication. Detailed security analysis shows that lightweight message authentication code (LMAC) also thwarts passive attack as well as active attack [14].

L. Archana et al (2018), suggested a hybrid encryption algorithm by integrating AES and Fully Holomorphic algorithm to encrypt the data in the cloud, thereby file size get is compressed thereby increasing Data security and stack pile [15].

Fenghua Zhang et al (2019), discussed design a hybrid algorithm to solve the data security problem in the hospital cloud database. First, the AES algorithm is improved. The improved algorithm is called P-AES algorithm. The P-AES algorithm is then combined with the RSA algorithm, called a hybrid algorithm. The experimental results show that the hybrid encryption algorithm has the advantages of fast encryption and decryption speed, high security, good processing ability for longer data, and can solve the data security problem in cloud database to a certain extent [16].

V. SALSA20 ALGORITHM

Bernstein developed the Salsa20 Stream Cipher in 2005 as a candidate for the Salsa 20/12 eSTREAM and was accepted into the 2008 eSTREAM software portfolio. Salsa20 provides a very simple, smooth and scalable interface and quite naturally supports 128- and 256-bit keys. From a safety point of view, Bernstein recommends a hidden 256-bit key. Initial Salsa has 20 rounds But Salsa later with 8 (Salsa20/8) and 12 (Salsa20/12) numbers. The Salsa 20/12 and Salsa20/8 reduced-round ciphers are among the fastest available 256-bit stream ciphers and are ideal for applications that need faster speed than trust. [17].

Reduced rounds ciphers are attractive options in the Salsa20 family for users who value speed higher than safety. Through of these rounds comprises three steps [18]:

32-bit addition, which generates a total of a + b mod 2^{32} of a, b.

Exclusive-32-bit xor a + b ○produced, with two 32-bit word a, b.

32-bit rotation with a constant distance generating a < < b of a 32-bit word a by b bit to the left, where b is constant. Fig.3 shows the overall block diagram of the Salsa20 encryption algorithm

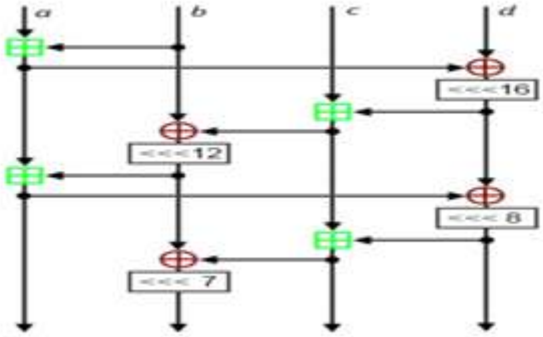


Fig.3 : A block diagram showing in Salsa20.[18]

VI. THE PROPOSED SYSTEM DESIGN:

The proposed system includes a number of advanced lightweight encryption and authentication techniques consisting of numerous security algorithms that work together to improve safety performance. Therefore, the proposed system consists of more than one algorithm, combined into a single security system. These systems are:

- 3D Chaos Keys Generation (using Lorenz chaos system).
- New Lightweight Encryption Algorithm with Modified Salsa20 (LHKHA-MSA).
-

The proposed lightweight security mechanism for Cloud (PLSMC) combines safety algorithms, provides encryption to the data by applying a new hybrid encryption algorithms. The cloud computing system work with numerical data. Data blocks were encrypted using hybrid encryption consisting of more than one encryption algorithm. The hybrid encryption is merge of new lightweight encryption algorithm with modify salsa20, we called it (LHKHA-MSA). Also we use different chaos keys (Lorenz chaos system) in both encryption and decryption. For making cloud more secure and to Confidence the data.

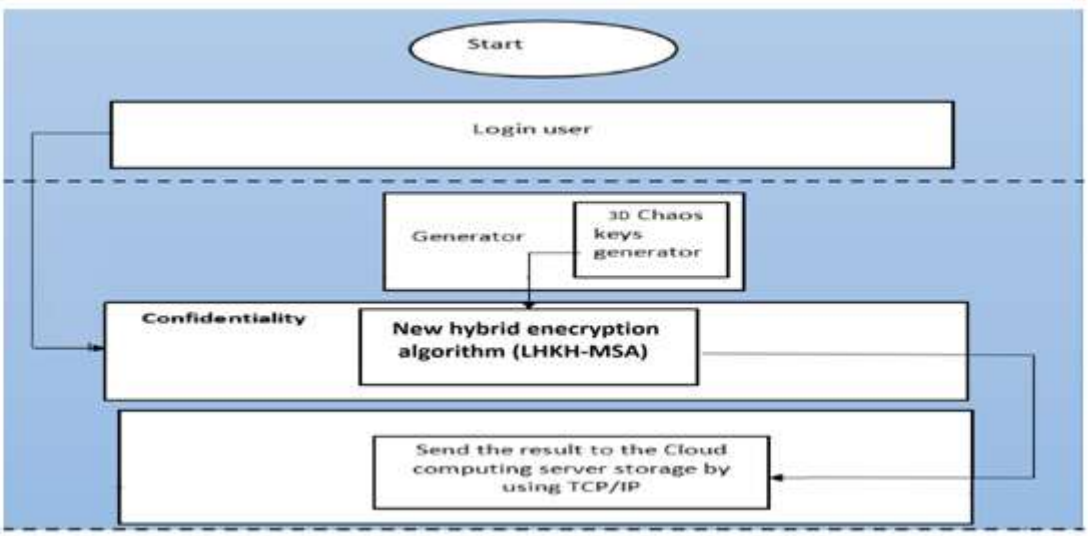

Start
Login user
Generator
3D Chaos keys generator
Confidentiality
New hybrid enecryption algorithm (LHKH-MSA)
Send the result to the Cloud computing server storage by using TCP/IP

Fig.4 : Block Diagram Of The Proposed System Structure And Internal

VII. PROPOSED MODIFIED SALSA20 ALGORITHM

The first stage of the proposed system is modified the Salsa20 Algorithm (SA). We proposed a modification to (SA) in order to obtain high efficiency in encryption, resistance to various types of attacks with a suitable execution time, and to save memory. The modification adapted the SA was combines with 3D chaotic system (Lorenz map) used to generate three chaos keys (SK1, SK2, and SK3). These chaos keys were used for making cloud more secure and to confidence the data or privacy against attacks.

All the chaos keys well be generated and stored before any operations. In addition, the SA core builds an array of 16 words to produce a 64-byte blocks containing of 8 (32-bits sub chaos keys) as (P1, P2, P3… P8), 4*(32 bits) of constant, 2*(32 bits) of plaintext, and 2*(32 bits) of block counter. The SA is a long chain of three simple operations on 32-bit words (32-bit addition, 32-bit exclusive-or, constant-distance 32-bit rotation). The 16-word array of SA was illustrated in table (1).

Table 1 SA Word Array for Encryption

Constant word 16 byte	P1 16 byte	P2 16 byte	P3 16 byte
P4 16 byte	Constant word 16 byte	Input 1 32 byte	Input 2 16 byte
Counter word 16 byte	Counter word 16 byte	Constant word 16 byte	P5 16 byte
P6 16 byte	P7 16 byte	P8 16 byte	Constant word 16 byte

The first step on SA is xor the diagonal and above-diagonal words, rotate left by 7 bits, and store resulte into the below below-diagonal words. The result is explaine on the follwing equations.

T (2.1) = ((T(1.1) T$\oplus$1)) <<< sk1
T (3.2) = ((T(2.2) T$\oplus$2)) <<< sk2
T (4.3) = ((T(3.3) T$\oplus$3)) <<< sk1
T (1.4) = ((T(4.4) T$\oplus$4)) <<< sk3 .... (1)

The second step on SA is xor the diagonal and below-diagonal words, rotate left by 9 bits for maximum, and store resulte into the below-diagonal words. The result is explaine on the follwing (2) equations.

T (3.1) = ((T(1.1) T$\oplus$1)) <<< sk3
T (4.2) = ((T(2.2) T$\oplus$2)) <<< sk1
T (1.3) = ((T(3.3) T$\oplus$3)) <<< sk1
T (2.4) = ((T(4.4) T$\oplus$4)) <<< sk2 .... (2)

SA continues down each column, rotating left by 13 bits for maximum, then modifies the diagonal words, this time rotating left by 18 bits for maximum, finally transposes the array. This steps will repeated 8 round until accessed to final cipher text result.

The chaotic system used in the Lorenz chaos map (equation 3).It is a continuous nonlinear system with unstable paths for different system parameter values. This method was proposed by Lorenz as a series of three ordinary differential equations to model an atmospheric fluid convection thermally [20].

$$\left.\begin{aligned}\bar{X} &= \sigma(y-x)\\ \bar{Y} &= x(\rho-z)-y\\ \bar{Z} &= xy-\beta z\end{aligned}\right\} \quad \text{.... (3)}$$

X is proportional to the velocity of circulatory fluid
Y explains the difference in temperature between rising and falling fluid regions,
Z characterizes the distortion of the height variance linear vertical temperature profile.

The σ parameter is related to the number of Prandtl, R to the number of Rayleigh and β to a geometry. The $\bar{X}$, $\bar{Y}$, and $\bar{Z}$ used to get the sk1, sk2, and sk3.

VIII. THE PROPOSED NEW LIGHTWEIGHT ENCRYPTION ALGORITHM WITH MODIFY SALSA20 (LHKHA-MSA))

In this proposed system, the researcher enhanced to the operations of MSA by embedded a new lightweight 64-bit block cipher we called it the “HKH” algorithm and its working on only 20 rounds. When combined LHMH with MSA, it will be avoiding the broken by impossible differential attack (or other attacks).

The proposed system (LHKH-MSA) deal with 128 bits of data. The 128 bits of data that entered to the system will be divided into two equal parts size 64-bits. The both section was combines with chaos keys (K1, K2) and (K3, K4) respectively generated by (proposed hyper chaotic system) and then the first section processed using the S-box, and the second section will be encrypted by MSA. And then they will be collected together to generate 128-bits of encrypted data.

The all keys may be generated by hyper chaos are stored in file before any operations. In addition, we proposed to used (8*8) of S-BOX that works on parallel to decrease time that taken by 8 S-BOX.

The series of the operations started from the inputs 128-bit block; the input block will be divided into two-part blocks (IL0, IR0) of size 64-bits. The first part (IL0) are XORed with Chaos key (K1, K2). The result stored on (IL1). Then the result divided in to 8*8 bits to produce as (L0, L1, L2 … L7).

The eight parallel S-boxes were used in parallel case on (L0, L1, L2 … L7), to produce the outputs (S0, S1, S2,… S7). The output of S-box operation was collected one more time to produce 64-bits and the result stored in (IL2). IL2 rotating (by K2) XORing Operations (with K7, K8) are done on them.

Second part of input block (IR0) was encryption by stream cipher MSA. 64-bits of input block (IR0) are Xoring with 64-bit chaos keys (k3, k4). So, we get 64 bits of the input data that passed to the SA. The MSA builds an array of 16 words holds the input data (64-bits), the constant word (128-bits), the chaos keys word (256 bits), and the counter word (128 bits). The Salsa20 encryption function consists of the 64-bit word operations that repeated 3 iteration (addition, exclusive-or, rotation).

The final result from MSA is 64-bits of cipher text will be stored in (IR1). IR1 is rotates (by K6) and XORing operations (with K4, K5) and stored in IR2. Then the 64-bits that stored in (IL2) with 64-bits output of (IR2) will be swapped and the operations repeated until 20 rounds to produce final result 128-bits cipher block.

The general block diagram of the proposed modified hybrid algorithm as illustrated in Figure (5).

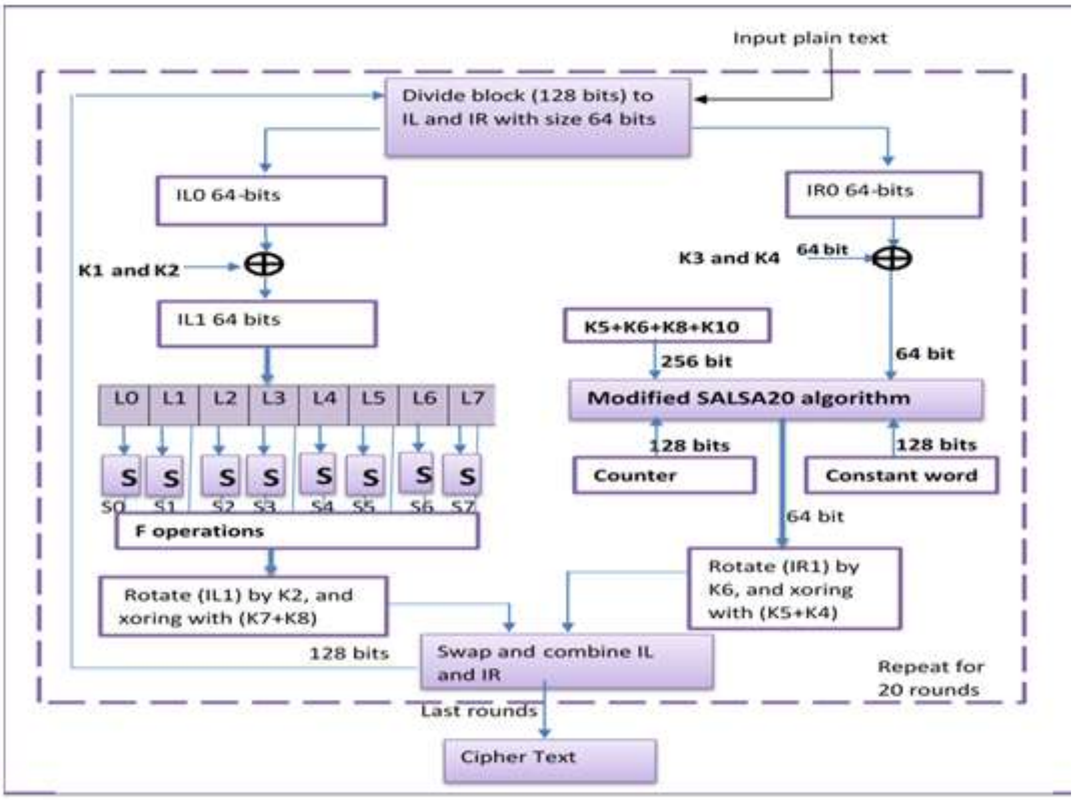


Fig 5: the Diagram of the Proposed Lightweight Hybrid LHKHA-MSA Algorithm

IX. EXPERIMENTAL RESULTS

Many researches in hashing algorithms were designed and implemented to prove the confidentiality of data (cloud computing) integrity transfer through the networks. The proposed (LHKHA-MSA) algorithm was implemented in the manner of cloud data confidentiality and integrity using chaotic keys generating system to randomly increase of the primitive modified stages of the (LHKHA-MSA). The proposed system have been tested by using several tests. The time processing comprising for different data size are shown in Table (2). The statistical tests of NIST comprising for (LHKHA-MSA) in different rounds (12 and 24 rounds) are shown in Table (3).

File Size	(LHKHA-MSA)(32 round) (m-sec) 64-bit	(LHKHA-MSA)(32 round) (m-sec) 128-bit
1KB	0.415	0.220
10KB	2.1551	1.6240
100KB	4.8256	3.5001
1MB	31.9788	26.2230
10MB	300.7765	250.0933

Table 2. Benchmarking Performance of the LHKHA-MSA (32 rounds) average time (in msec) using the Random Input Data

As shown in Table 2, shows the MSA and LHKH-MSA a shorter period of encryption/decryption than other forms of LHKHA (with 64 or 128 bits), input data generated by random size (from 1KB to 1MB) with a different input block for (LHKH-MSA, LHKHA-128, LHKHA-64, and MSA) as shown in figure (6). The benchmarking average LHKH-MSA encryption (0.15 sec to 226.452 msec) and MSA (0.11 sec to 199.789 msec).

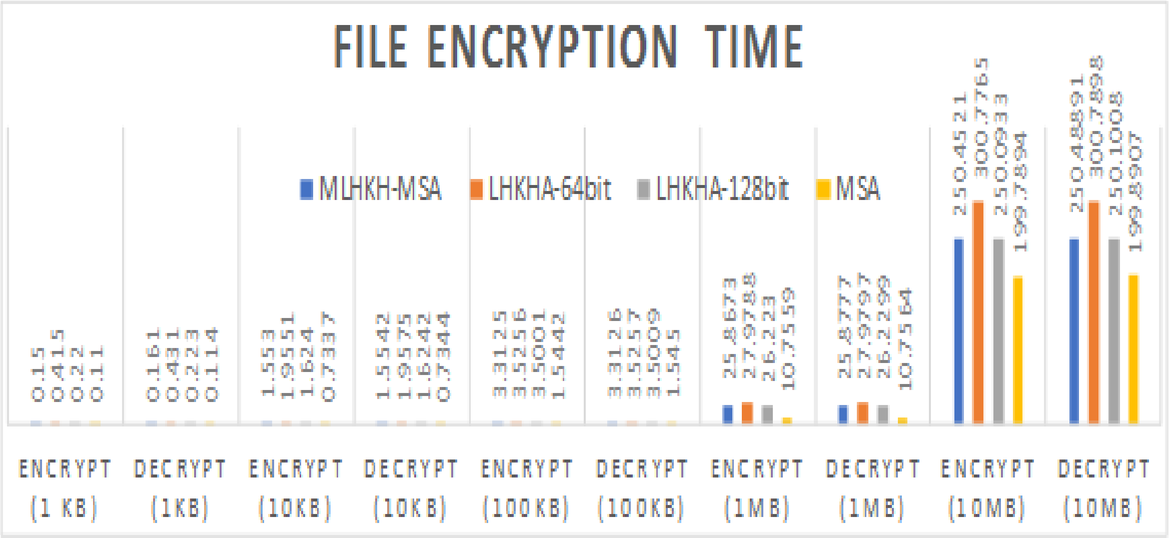
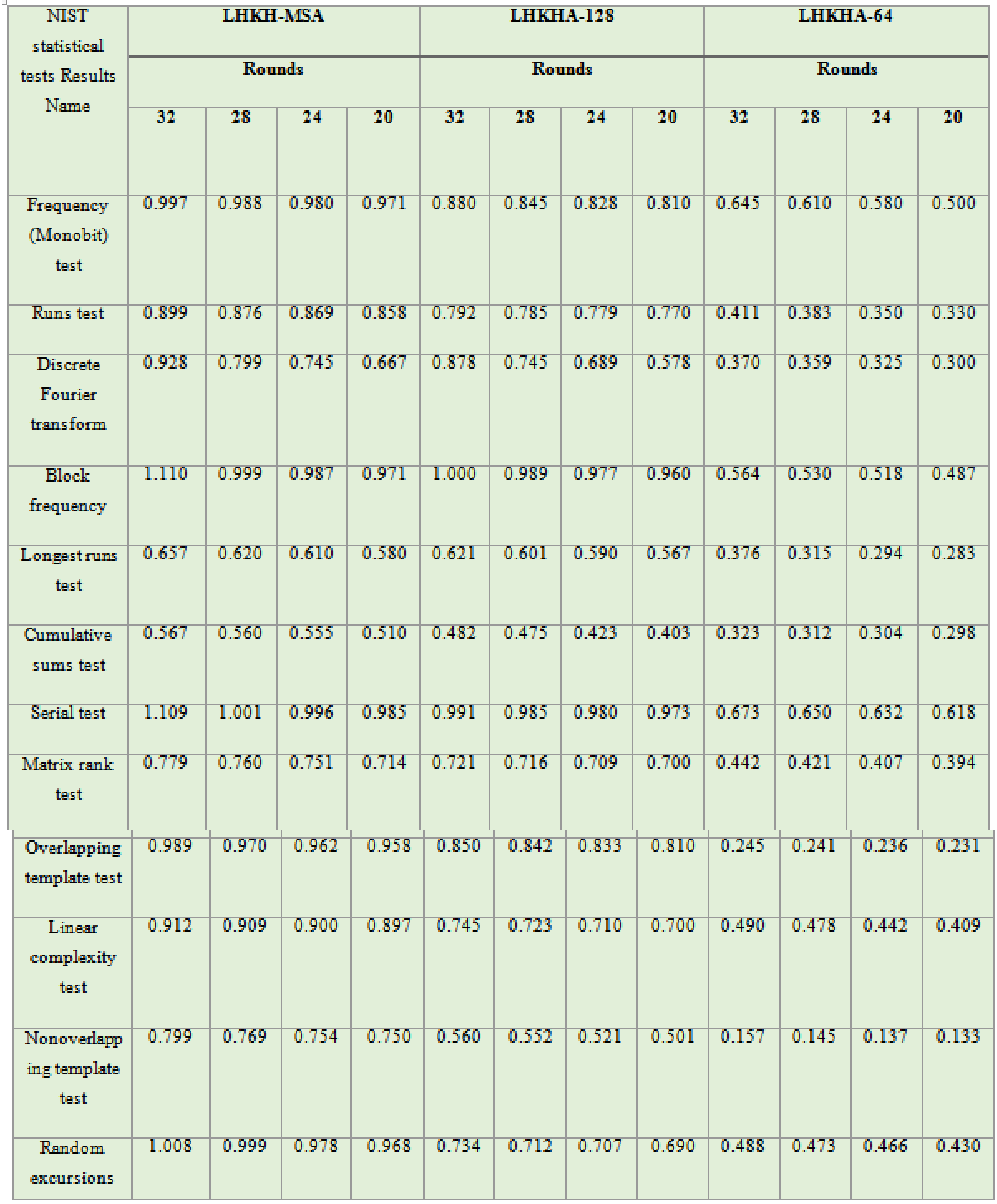


Fig 6: File Encryption Time comparison between LHKH-MSA, LHKHA-64bit, LHKHA-128bit, and MSA.

Table 3, shows NIST test results when applying (LHKHA-MSA) with different rounds (20, 24, 28 and 32) to various text sizes. The (LHKHA-MSA) passed all NIST tests and indicates the (LHKHA-MSA) has security features and can avoids many attacks types in both (12 and 24) rounds cases.

Table 3: the NIST Tests Results Comparison



NIST statistical tests Results Name	LHKH-MSA				LHKHA-128				LHKHA-64			
	Rounds				Rounds				Rounds			
	32	28	24	20	32	28	24	20	32	28	24	20
Frequency (Monobit) test	0.997	0.988	0.980	0.971	0.880	0.845	0.828	0.810	0.645	0.610	0.580	0.500
Runs test	0.899	0.876	0.869	0.858	0.792	0.785	0.779	0.770	0.411	0.383	0.350	0.330
Discrete Fourier transform	0.928	0.799	0.745	0.667	0.878	0.745	0.689	0.578	0.370	0.359	0.325	0.300
Block frequency	1.110	0.999	0.987	0.971	1.000	0.989	0.977	0.960	0.564	0.530	0.518	0.487
Longest runs test	0.657	0.620	0.610	0.580	0.621	0.601	0.590	0.567	0.376	0.315	0.294	0.283
Cumulative sums test	0.567	0.560	0.555	0.510	0.482	0.475	0.423	0.403	0.323	0.312	0.304	0.298
Serial test	1.109	1.001	0.996	0.985	0.991	0.985	0.980	0.973	0.673	0.650	0.632	0.618
Matrix rank test	0.779	0.760	0.751	0.714	0.721	0.716	0.709	0.700	0.442	0.421	0.407	0.394
Overlapping template test	0.989	0.970	0.962	0.958	0.850	0.842	0.833	0.810	0.245	0.241	0.236	0.231
Linear complexity test	0.912	0.909	0.900	0.897	0.745	0.723	0.710	0.700	0.490	0.478	0.442	0.409
Nonoverlapping template test	0.799	0.769	0.754	0.750	0.560	0.552	0.521	0.501	0.157	0.145	0.137	0.133
Random excursions	1.008	0.999	0.978	0.968	0.734	0.712	0.707	0.690	0.488	0.473	0.466	0.430

X. CONCLUSION

The tests obtained shows, that the proposed system security has reached to the good results and all the objectives that have been set previously were achieved, through the availability of a high degree of security and reliability to the personal privacy. The confidentiality and data integrity operation be focus fields in many applications and systems used the sensors/ devices, cloud computing and networking in data collection and transferring. We can conclude that, the modifications added to the (LHKHA) and Salsa20 algorithm and combination of them are support the (LHKHA-MSA) to avoid many attacks and work with high-speed hashing in various file sizes.

REFERENCES

[1]. Omotunde, A. A., Awodele, O., Kuyoro, S. O., & Ajaegbu, C. (2013). Survey of cloud computing issues at implementation level. Journal of Emerging Trends in Computing and Information Sciences, 4(1), 91-96..

[2]. Prasadreddy, P. V. G. D., Rao, T. S., & Venkat, S. P. (2011, July). A threat free architecture for privacy assurance in cloud computing. In 2011 IEEE World Congress on Services (pp. 564-568). IEEE.

[3]. Karuppanan, K., AparnaMeenaa, K., Radhika, K., & Suchitra, R. (2012, August). Privacy adaptation for secured associations in a social cloud. In 2012 International Conference on Advances in Computing and Communications (pp. 194-198). IEEE.

[4]. Bokhari, M. U., Shallal, Q. M., & Tamandani, Y. K. (2016, March). Cloud computing service models: A comparative study. In 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom) (pp. 890-895). IEEE.

[5]. Azam, Md Gulnawaz. (2019). Application of cloud computing in library management: innovation, opportunities and challenges. RESEARCH REVIEW International Journal of Multidisciplinary, Volume-04 Issue-01 January -2019

[6]. Mandeep Kaur and Manish Mahajan, 2013, ―Using encryption Algorithms to enhance the Data Security in Cloud Computing‖, International Journal of Communication and Computer Technologies Volume 01 – No.12, Issue: 03 January 2013.

[7]. Gajendra, B. P., & Singh, V. K. (2016, April). Achieving cloud security using third party auditor, MD5 and identity-based encryption. In 2016 International Conference on Computing, Communication and Automation (ICCCA) (pp. 1304-1309). IEEE.

[8]. Hakim, S., & Fouad, M. (2017). Improving data integrity in communication systems by designing a new security hash algorithm. Journal of Information Sciences and Computing Technologies (JISCT), 6(2), 638-647.

[9]. Gope, P., Das, A. K., Kumar, N., & Cheng, Y. (2019). Lightweight and physically secure anonymous mutual authentication protocol for real-time data access in industrial wireless sensor networks. IEEE transactions on industrial informatics, 15(9), 4957-4968.

[10]. Saini, G., & Sharma, N. (2014). Triple security of data in cloud computing. International Journal of Computer Science and Information Technologies, 5(4), 5825-5827.

[11]. Naif, J. R., Abdul-majeed, G. H., & Farhan, A. K. (2019). Internet of Things Authentication Based on Chaos-Lightweight Bcrypt. Baghdad College of Economic sciences University, 2019(8), 370-385.

[12]. Singla, S., & Singh, J. (2013). Cloud data security using authentication and encryption technique. Global Journal of Computer Science and Technology.

[13]. Ravilla, D., & Putta, C. S. R. (2015). Enhancing the security of MANETs using hash algorithms. Procedia Computer Science, 54, 196-206.

[14]. Chowdhury, A. R., & DasBit, S. (2015, December). Lmac: A lightweight message authentication code for wireless sensor network. In 2015 IEEE Global Communications Conference (GLOBECOM) (pp. 1-6). IEEE.

[15]. Sajay, K. R., Babu, S. S., & Vijayalakshmi, Y. (2019). Enhancing the security of cloud data using hybrid encryption algorithm. Journal of Ambient Intelligence and Humanized Computing, 1-10.

[16]. Zhang, F., Chen, Y., Meng, W., & Wu, Q. (2019). Hybrid encryption algorithms for medical data storage security in cloud database. International Journal of Database Management Systems (IJDMS) Vol, 11.

[17]. Ding, L. (2019). Improved related-cipher attack on Salsa20 Stream Cipher. IEEE Access, 7, 30197-30202.

[18]. Bernstein, D. J. (2008). The Salsa20 family of stream ciphers. In New stream cipher designs (pp. 84-97). Springer, Berlin, Heidelberg.

[19]. Lawande, Q. V., Ivan, B. R., & Dhodapkar, S. D. (2005). Chaos based cryptography: a new approach to secure communications. BARC newsletter, 258(258).

IJAEM

International Journal of Advances in Engineering and Management

ISSN: 2395-5252

IJAEM

Volume: 03 Issue: 02 DOI: 10.35629/5252

www.ijaem.net

Email id: ijaem.paper@gmail.com