Diyala Journal for Pure Science



# Multi-Resolution Like Data Transmission Security Using Diffie-Hellman Protocol

**Hanaa Mohsin Ahmed[**] and Rozeen Waheed Jassim[*]**

Computer Sciences Department – University of technology – Baghdad – Iraq

[*]rozeeenalqaisy@gmail.com
[**]110113@uptechnology.edu.iq

**Received: 28 August 2019** **Accepted: 2 December 2019**

**DOI: https://dx.doi.org/10.24237/djps.16.01.516B**

## Abstract

In this paper, special schema has been proposed for long message transmission which has been encrypted in efficient manner that is called (parallel multi-resolution like) encryption approach, that means multiple levels or sub-image represented in a matrix contains data of long message, each level of this matrix related with classical encryption methods with a secret keys that generated by (Diffie–Hellman) key exchange protocol that utilized in our cryptosystem. The statistical analysis that applied on encrypted data indicates that cipher text passed in each statistical test. Moreover, this cryptosystem removed an important problem which is (Man in the middle attack) in that protocol and (Brute force attack). Then efficiency of data transfer security has been increased when using the hash function (SHA256) and separating number of characters, the proposed method has improved the mechanism of protocol (D-H), making it possible to send long messages more confidentially.

**Keywords**: Multi-Resolution, Hashed Diffie - Hellman Protocol, Data transmission security, Long messages encryption, Brute force attack.

Diyala Journal for Pure Science

# Multi-Resolution Like Data Transmission Security Using Diffie-Hellman Protocol

Hanaa Mohsin Ahmed and Rozeen Waheed Jassim

# امنية تناقل البيانات المتعددة الدقة لعدة جهات بأستخدام بروتوكول تبادل المفاتيح

# (الديفي هيلمان Diffie- Hellman)

هناء محسن احمد و روزين وحيد جاسم

قسم علوم الحاسبات – الجامعة التكنولوجية – بغداد – العراق

## الخلاصة

في هذه المقالة، تم اقتراح مخطط او نظام خاص لإرسال الرسائل الطويلة التي تم تشفيرها بطريقة فعالة تسمى بنهج تشفيري (متعدد الاستبانة او متعدد الدقة)، وهذا يعني أن مستويات متعددة تحتوي على بيانات الرسائل الطويلة التي تمثلت في مصفوفة كل مستوى أو صورة فرعية لهذه المصفوفة تتضمن طرق تشفير كلاسيكية ومفاتيحها السرية التي تم إنشاؤها بواسطة بروتوكول تبادل المفاتيح Diffie –Hellman. حيث يشير التحليل الإحصائي الى أن البيانات المشفرة قد مرت باختبارات إحصائية محددة ونجحت فيها. علاوة على ذلك، فقد حل نظام التشفير المقترح هذا مشكلة مهمة الا وهي (الرجل المتلصص في المنتصف) لهذا البروتوكول. وقد تمت زيادة كفاءة أمان نقل البيانات عند استخدام دالة التجزئة SHA256، والتي تفصل بين الاعداد من الأحرف، والطريقة المقترحة قد حسنت آلية البروتوكول D-H، مما يجعل من الممكن إرسال رسائل طويلة بسرية أكبر.

**الكلمات المفتاحية:** تشفير الرسائل الطويلة، مفهوم الدقة المتعددة، هاش بروتوكول ديفي-هيلمان، امنية تناقل البيانات و هجوم القوة الغاشمة.

## Introduction

Recently data can be representation in multiple resolutions, where each resolution can be used for a different purpose e.g., resolution for keys and resolution for multi-level when each level is encrypted using different keys. And utilizing multiple simultaneous encryption methods, thus requiring multiple independent decryption methods with development of encryption algorithms and computer controlled communication needed for data transmission has led to encrypt it, therefore modern cryptography is a remarkable discipline of computer and communications

security, in order to prevent breaking the code by confusion and diffusion or to find information through statistical analysis of the data [1].

Authenticated key exchange is one of the more subtle goals in cryptography, later applying the paradigms or algorithms of modern cryptography to see how to define this goal and provide high-assurance solutions, including Diffie-Hellman key exchange protocol [1].

In this protocol (Whitfield Diffie and Martin Edward Hellman1976) [2] identified concepts for two parties (a two-key cryptosystem) instead of a single key to settle over shared secret in a manner that the secret is going to be inaccessible by an eaves dropper, where this protocol considered as a combination of public and private key cryptographic to generate these secret key. A secret key will be shared by Bob and Alice in symmetric cipher; however, the problem is the insecurity of their way of communication.

The initial step is that protocol, Bob and Alice should agree upon non-zero integer G and prime P. The values of G and P will be a public knowledge by Bob and Alice, for example, in the next step Alice is going to select (a) secret integer that will not be revealed, also an integer (b) will be selected by Bob and it will also be kept secret.

In 2005, Yang et al. [3] Suggested secure authentication system for the protocol of session initiation depending on D-H protocol; in their procedure, four modular exponentiations are computed for the server and user, and ten amounts of transmitted data.

Then in 2010 (Han et al.) [4] Proposed "parallel multi-receiver scheme framework" such system introduced a new paradigm which is referred to as the parallel multi-recipient signcryption that improves the performance in the case when the sender transmits distinct message to multiple-recipients in an imbalanced wireless network. Parallel multi-recipient signcryption system which is referred to as Para SC-GDH is also suggested which is semantically secure.

(JIQIANG LIU et al.) In 2011 [5] Devised "An Identity Based Cryptosystem for encryption long messages", they give first identity-based cryptosystems for the encryption of long

messages, and offer security within random oracle model, in an ID-based public-key cryptosystems, like IP address, ID number, E-mail address, a trusted third-party key generation center (PKG) is used to assign the private key.

At 2013, (ZHANG Minqing et al.) [6] proposed "Secure group communication based on distributed parallel ID-based proxy Re-encryption" A distributed parallel proxy re-encryption system for parallel multi-receivers in the wireless environment, and implemented public-key certificate encryption system for solving the secure multicast problem, it is considered as an extremely effective system which allows a proxy to transform the cipher text computed under sender's identity into the one which may be decrypted by receiver with its own secret key, while the semi-trusted proxy can't decrypt the cipher text message.

## Background

### A. Multi-Resolutions

A method of representing the load data in multiple resolutions may be found in, whereas this method has described the way of splitting the load data in multiple resolutions, which raises a question concerning proper key generation and managements. And it is an approach analysis offers a simple, unified, and theoretically resolution to dealing with data problems [7 and 8].

So, its clusters of data at different scales resolution, adds additional flexibility and control for the user [9 and 10].

### B. Diffie – Hellman key exchange protocol [11]

The D-H protocol named according to Whitfield Diffie and Martin E. Hellman, who in 1976 proposed this protocol in their work and the need for a secure channel was to perform the public key distribution. This protocol was proposed as a method for public key distribution system that differed from the other classical key distribution systems in which the keys need to be

distributed using a previously secured channel. Depending on the public key cryptography, this protocol used the discrete logarithm problem as a base.

For establishing a shared secret with the use of a non-secured channel, the parties exchange their public keys and combine the other party's public key with their own private key in an exponentiation manner and following the multiplicative group's fashion.

As described by Diffie and Hellman in their work; this protocol is illustrated as follows:

Assume that Alice selects a random number $X_A$ (PRIVET KEY FOR ALICE) in the interval $0 < X_A < p$, where $X_A$ keeps secret and calculating.

$$\mathbf{Y_A} = \mathbf{g}^{\mathbf{X_A}} \boldsymbol{mod\ p} \qquad \textbf{(1)}$$

The previous calculation $\mathbf{Y_A}$ = public key for ALICE is sent to Bob.

Now, assume that Bob selects a random number $\mathbf{X_B}$ (private key for BOB) in the interval $0 < X_B < p$, where $X_B$ keeps secret and calculates:

$$\mathbf{Y_B} = \mathbf{g}^{\mathbf{X_B}} \boldsymbol{mod\ p} \qquad \textbf{(2)}$$

The previous calculation $\mathbf{Y_B}$ = public key for bobis sent to Alice.

When Alice and Bob want to communicate, they use the following calculation exchange key

$$\mathbf{k} = \mathbf{g}^{\mathbf{X_A X_B}} \boldsymbol{mod\ p} \qquad \textbf{(3)}$$

Which they are obtained by combining their secrets and the public information from the other party

$$\mathbf{Y_B}^{\mathbf{X_A}} \boldsymbol{mod\ p} = \mathbf{g}^{\mathbf{X_B X_A}} \boldsymbol{mod\ p} = \mathbf{Y_A}^{\mathbf{X_B}} \boldsymbol{mod\ p} = \boldsymbol{k} \qquad \textbf{(4)}$$

One of the well-known problems of Diffie-Hellman protocol is that, it doesn't provide authentication. A man-in-the-middle attack could be performed, where an entity C can intercept the communication and act as the legitimate party by providing a different public key to the legitimate parties and sitting as an intermediate point in their communications in the following figure 1:

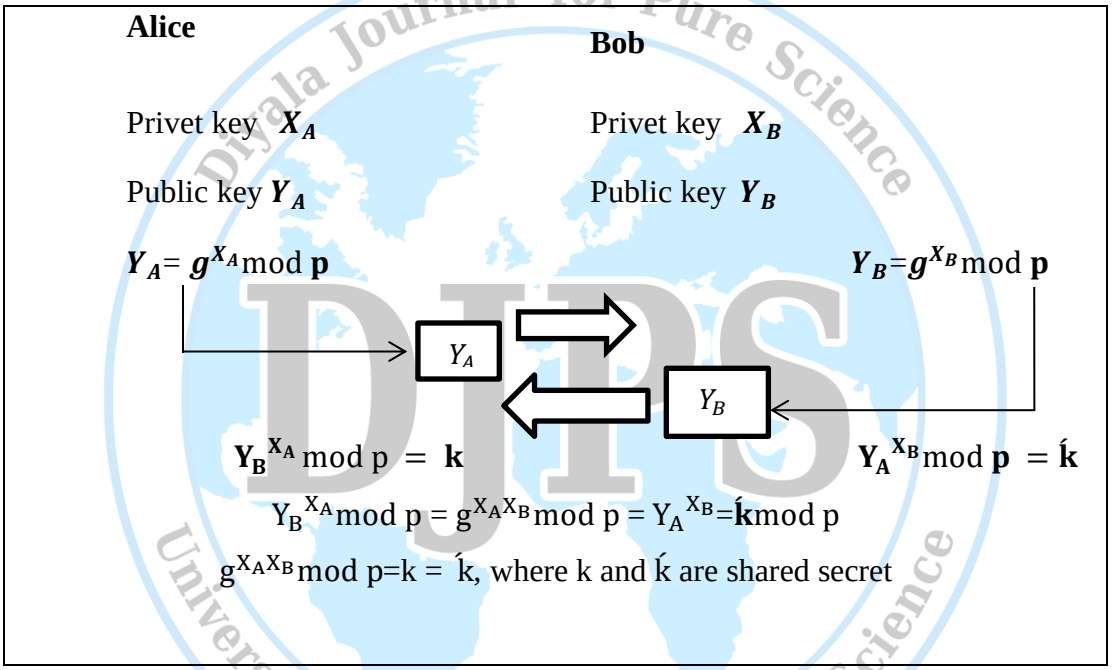



**Figure1:** Diffie-Hellman Protocol of Keys Exchange

## C. Hash Functions [12]

Can be defined as algorithms which have been particularly created for the hashing operations, and It is a function (H) that maps an arbitrary length message M to a fixed length message digest MD is a One-Way Hash Function (OWHF) as in figure 2, it has been mapped a large collection of messages into a small set of message digests and can be used for error detection, by appending the digest to the message during the transmission.

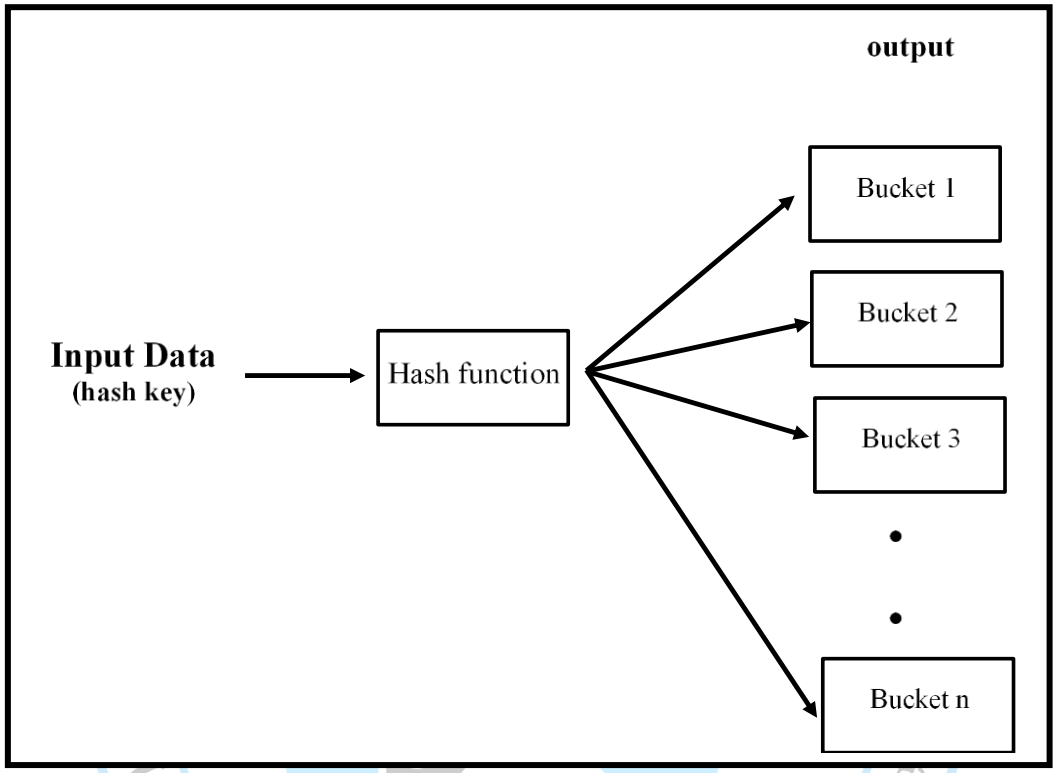



**Figure 2:** Hash function and Hash key [13]

This proposed cryptosystem has been selected (SHA-256) form SHA-2 family, for use in encrypt secret keys that generated in Diffie-Hellman protocol which is take input that has a length of less than 264 and produces a 256-bit hash value, It has a block size of 512 bits which are represented as a sequence of sixteen 32-bit words, and this function gives to the algorithm more efficiency because the most features of the basic components in SHA-2 (SHA256) provide a better security level than other hash function [14].

### D. Statistical Test [15]

These five tests are commonly used for determining whether the binary sequences possess some specific characteristics that a truly random sequence is likely to exhibit, as illustrated in table

1, and we used this testing technique on data has been encrypted, that its equations illustrated in table below:

**1.** Frequency Test. **2.** Serial Test **3.** Poker Test **4**. Runs Test**5**. Auto-correlation Test

**Table 1**: Statistical Test equations

| Benchmark test | Benchmark test equations | Information 5 Benchmark test |
|---|---|---|
| FREQUENCY TEST | $T1 = \frac{(M0 - M1)^2}{M}$ | M0: number of 0's in key stream<br>M1: number of 1's in key stream<br>M : total size of key stream |
| RUN TEST | $T2 = \frac{4}{M-1}\left((M11)^2 + (M00)^2 + (M10)^2 + (M01)^2 - \frac{2}{M}(M1^2 + M0^2)\right) + 1$ | M11: number of 11's in key stream<br>M10: number of 10's in key stream<br>M01: number of 01's in key stream<br>M00: number of 00's in key stream |
| POKER TEST | $p = \frac{M}{N}\frac{M}{N} > (S + 2^N)$<br>$T3 = \frac{2^N}{p}\left(\sum_{j-1}^{2n} Mj^2\right) - p$ | Mj: number of appearances of j of Length N |
| SERIAL TEST | $pj = \frac{M - j + 3}{2^{(j+2)}}$<br>$T4 = \left(\sum_{j-1}^{N} \frac{(Bj - pj)^2}{pj}\right) + \left(\sum_{j-1}^{N} \frac{(Gj - pj)^2}{pj}\right)$ | N:maximum j for which p > s<br>Bj: Amount for blocks(subsequence runs of 1's) of length j in M<br>Gj:Amount of gabs(subsequence runs of 1's) of length j in M |
| AUTO-CORRELATION TEST | $A(k) = \sum_{j}^{m-k-1} (S1 + Sk) \bmod 2$<br>$T5 = \frac{2\left(A(k) - \frac{(M-k)}{2}\right)}{\sqrt{(M-k)}}$ | K: 1<k< [m/2] |

**The Proposed System**

This Proposed Crypto system has been introduced system for encrypting data of long message as well as decrypting long confidential message. This cryptosystem consists of two phases:

**Phase One: Hashed Diffie- Hellman Key Exchange Protocol**

In proposed system Hashed Diffie-Hellman key exchange protocol has been utilized for generate and exchange keys, which is used public information (p, g) and privet keys (a, b) for computing the public keys that has been exchanged between two parties (sender Alice and receiver Bob) to compute secret keys which is used for encrypting the plain text of long messages that exits in colored matrix after inserting (hash function (Sha256)) in to this key, as in algorithm 1.

**Algorithm 1:** Hashed Diffie- Hellman Key Exchange Protocol

**Input:** Generator number (g), prime number (p), Privet key for Alice (a) < P, Privet key for Bob (b) < P.

**Output:** Secret keys with it hash code.

**Begin**

**Step 1: Initialization**

**Step 1.1:** A secret random number is selected by each user, this secret number is called secret key, such that 1 < private key (a, b) < p.

**Step 1.2:** Compute the public key (X) for user1 (Alice) as, the public key (X) = $(g)^{a} \ mod \ p$.

**Step 1.3:** Calculate the public key (Y) for user2 (Bob)as, the public key (Y)= $(g)^{b} mod \ p$.

**Step 1.4:** Exchange key (X send to Bob, Y send to Alice) Send to receiverthrough a public channel.

**Step 1.5:** Compute secret key (Sk1) for Alice as: public key $(Y)^{a} \ mod \ P$.

**Step 1.6:** Compute secret key (Sk2) for Bob as: public key $(X)^{b} \ mod \ P$.

**Step 1.7:** Compute secret key2 (Ska) for Alice = $(Sk1)^{a} \ mod \ p$.

**Step 1.8:** Compute secret key2 (Skb) for Bob = $(Sk2)^{b} \ mod \ p$.

**Step 1.9:** Compute secret key3 (Sca) for Alice = $(Ska)^{a} \ mod \ p$.

**Step 1.10**: Compute secret key3 (Scb) for Bob = $(Skb)^{b} \ mod \ p$.

**Step1.11:** Compute hash function for Secret key (Sca): using (SHA256 ShaHash = SHA256.Create ())

**Step 1.12:** Compute hash function for Secret key (Scb): using (SHA256 ShaHash = SHA256.Create ())

**End.**

**Phase Two: Multi-Resolution Like in Matrix Schema for Data Transmission**

This system contains a form of matrix (n*n) which is divided into three levels, each level contains four parts or quarters. Then a long text message inserted into this matrix and each letter is distributed in each pixel of it, then each quarter is colored into four different colors, which related with four classical encryption methods (shift cipher, multiply cipher, vignner cipher and permutation). Then the secret keys that generated by (Diffie–Hellman key exchange protocol) has been divided to two types (letters and numbers) according to **(**classical encryption techniques) and then distributed by keys distributed button in each field of that methods, each one in it level (Lv1, Lv2 and Lv3)**.** This is described in algorithm 2.

**Algorithm 2:** Algorithm of Multi-resolution Like in Matrix

**Input:** Generator number (g), prime number (p), Privet key for Alice (a) < P, Privet key for Bob (b) <P, Size of matrix (number), insert text (" ").

**Output:** Matrix divided in to four different coloredquarterswiththree levels, encrypted textof long massage.

**Begin**

**Step 1: initialization**

**Step 1.1:** Insert the massage text that will be in matrix.

**Step 1.2:** Create a matrix by insert it size and division it to four colored quarters with three levels.

**Step 2: Distributed** hash code of Secret keys which is generated from (Diffie-Hellman protocol) on every level by click on Distributed key button.

**Step 3: Encryption**

**Step 3.1:** In level one, compute shift cipher in first quarter, computes Vigen`ere cipher in second quarter, Compute permutation in third quarter, and compute multiply cipher in fourth

**Step 3.2:** In level two, divided into four different colors and compute multiply cipher in first quarter, compute shift cipher in second quarter Compute Vigen`ere in third quarter, and compute permutation cipher in fourth quarter.

**Step 3.3:** In level three also divided into another four different colors and compute permutation cipher in first quarter, compute multiply cipher in second quarter, Compute shift in third quarter, and compute Vigen`ere cipher in fourth quarter.

**Step 4: Decryption**

Here execute operations opposite of the encryption processes, where we start from the third level of the matrix then second level finally first level to extract or produce a matrix with clear text

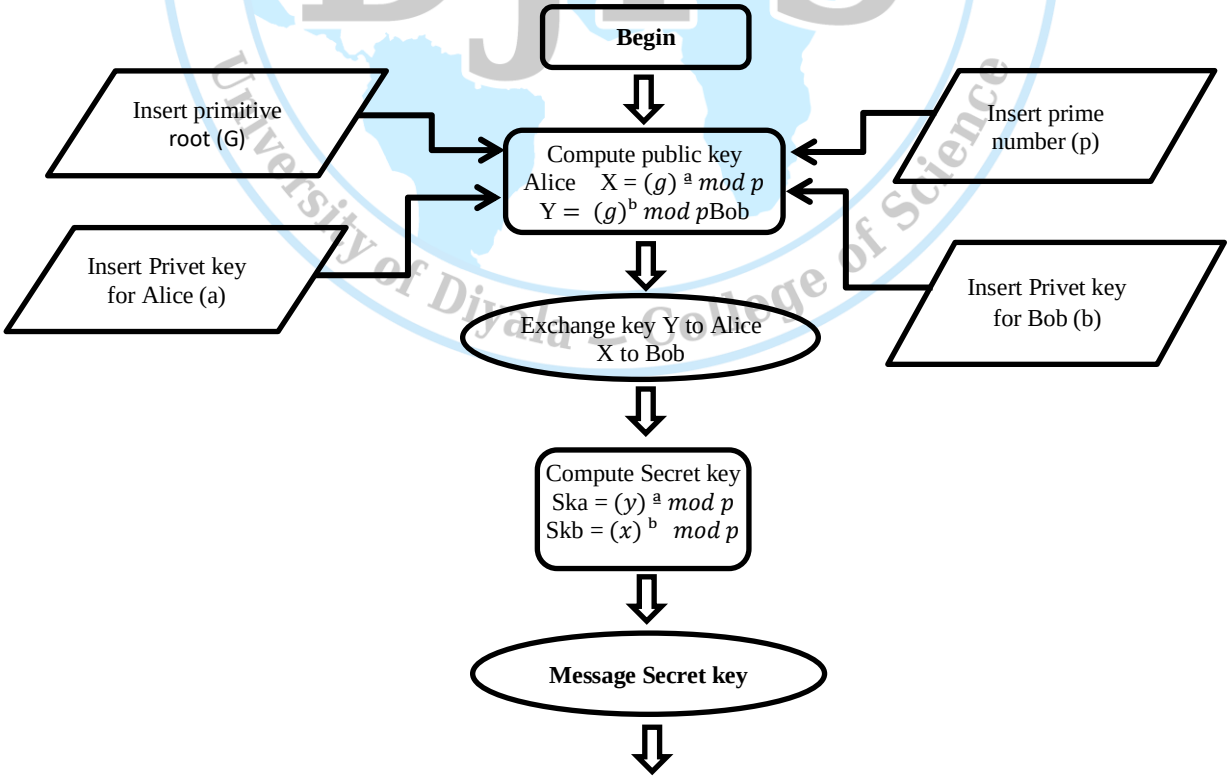
**Step 4.1:** In level three, click on permutation cipher in first quarter, compute multiply cipher in second quarter, Compute shift in third quarter, and compute Vigen`ere cipher in fourth quarter.

**Step 4.2:** In level two, compute multiply cipher in first quarter, compute shift cipher in second quarter Compute Vigen`ere in third quarter, and compute permutation cipher in fourth quarter.

**Step 4.3:** In level one compute shift cipher in first quarter, computes Vigen`ere cipher in second quarter, Compute permutation in third quarter, and compute multiply cipher in fourth quarter.

**End.**

Block diagram of the proposed system is shown in figure 3.



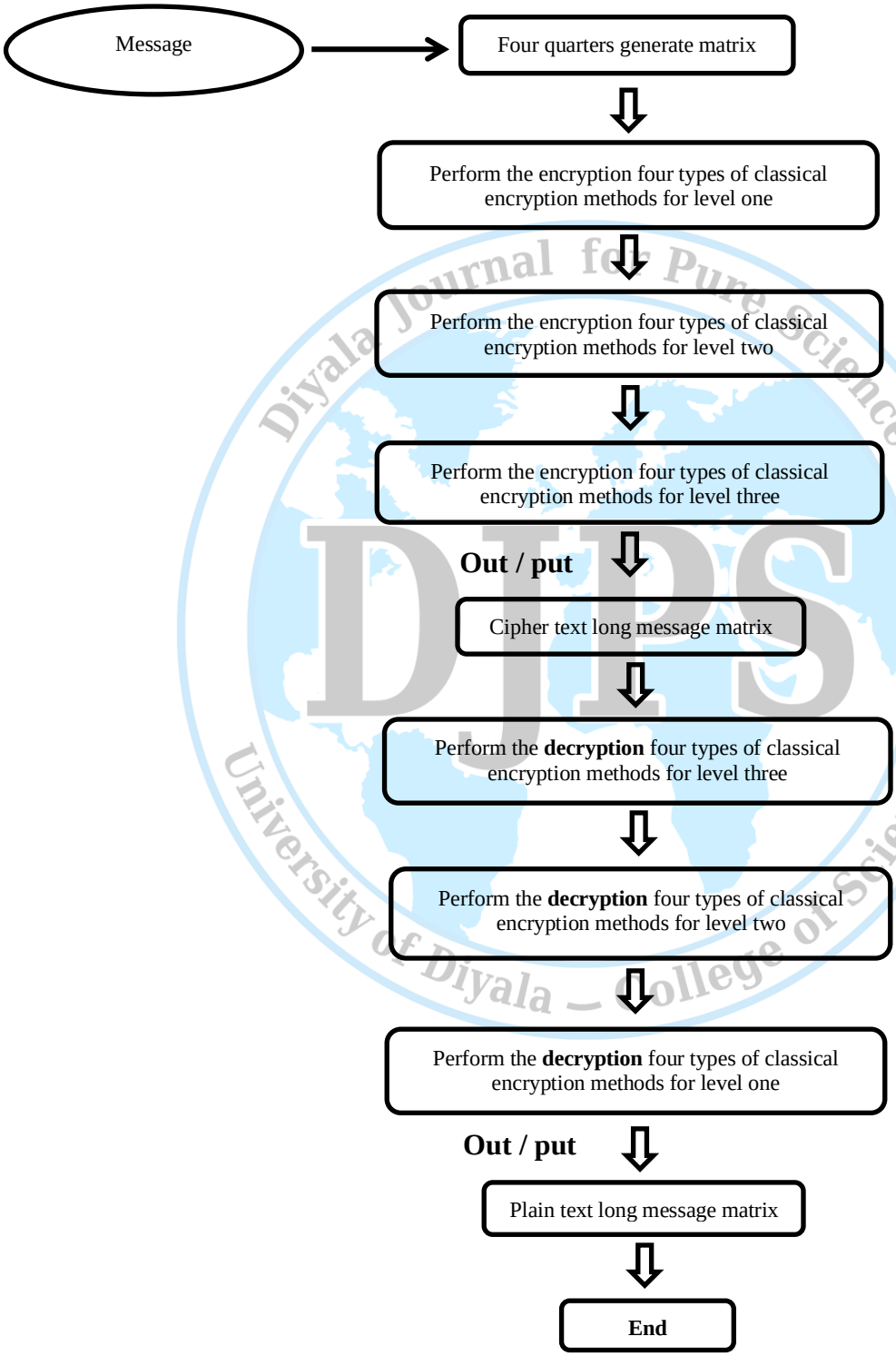



**Figure 3:** Block diagram of the proposed system

## Results and Test

### A. Implementation Example

Through the implementation an example of proposal cryptosystem appeared to us some main result may be utilized for proving that the system is a sufficient crypto-system for the long messages transmission, In addition to that when contact the (multi-resolution like) in the matrix with Diffie–Hellman protocol (D-H) this scheme has supported parallel computations which are all blocks of long message or plaintext (encrypted / decrypted) in a simultaneous manner as bellow:

**a.** Create matrix by insertion long massage in it such as: "This guide is meant for developers new to Dynamic.net twain sdk" and divided into four colored quarter as figure 4:


**Figure 4:** Create matrix

**b.** Generation the secret keys by insertion privet keys and compute public keys with (Diffie-Hellman protocol) which is used in this system.

**c.** Keys distribution into each levels of a matrix that contains classical encryption methods as figure 5.

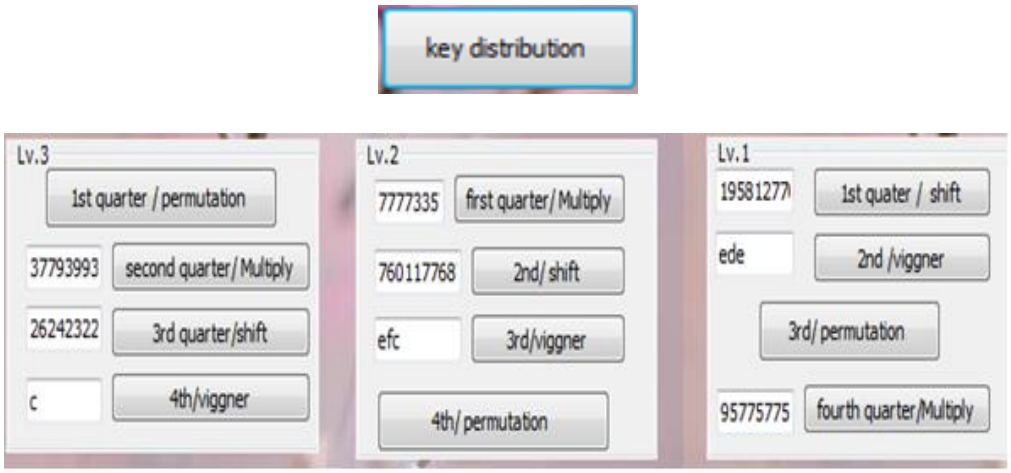



**Figure 5:** Keys Distribution on all classical encryption methods at levels (Lv1, Lv2, and Lv3)

**d.** Encryption the plain text in each levels of the matrix by classical encryption methods with these keys, when each left quarter encrypted for twice by a same way, also divided in to four part colored with different colors too, then results encrypted matrix that contain cipher long message which is presented in figure 6 bellow:

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| S | E | u | | | k | y | m |
| C | Ó | ' | q | w | | q | i |
| g | a | y | % | j | s | v | |
| d | n | z | m | p | s | t | i |
| | ? | à | | K | e | D | w |
| + | à | Ü | ] | S | | | t |
| µ | à | æ | ê | T | n | | a |
| a | Ã | ÿ | | T | m | E | i |

**Figure 6**: Encrypted matrix

**e.** Decryption cipher text of long messages in each quarter of the matrix and return us the original plain text "This guide is meant for developers new to Dynamic .net twain sdk" in same matrix with same classical encryption methods too, as in figure 7:


**Figure 7**: Original matrix

## B. Tests

After Implementation, must be analyzed this data which means make testing cipher text according to Basic Five Statistical Test [15]**:** (Frequency test, Serial test, Poker test, Runs test, Auto-correlation test), and the results are given in the following table 2 with figure 8 of it chart that illustrates the maximum threshold values of this test, and the cipher text has been passed in this test.

**Table2:** Statistical Test Valves

| Benchmark | Test value | Threshold value | Test value < Threshold |
|---|---|---|---|
| FREQUENCY TEST | 2.240 | 3.48 | Pass |
| RUN TEST | 10.375 | 12.309 | Pass |
| POKER TEST | 4.154 | 11.1 | Pass |
| SERIAL TEST | 2.407 | 7.81 | Pass |
| AUTO_CORRELATION TEST | SHIFT NO. 1 = 0.000<br>SHIFT NO. 2 = 0.021<br>SHIFT NO. 3 = 1.840<br>SHIFT NO. 4 = 0.398 SHIFT NO. 5 = 1.849 SHIFT NO. 6 = 1.988<br>SHIFT NO. 7 = 1.365<br>SHIFT NO. 9 = 0.238<br>SHIFT NO.10 = 0.690 | 3.84 | Pass |

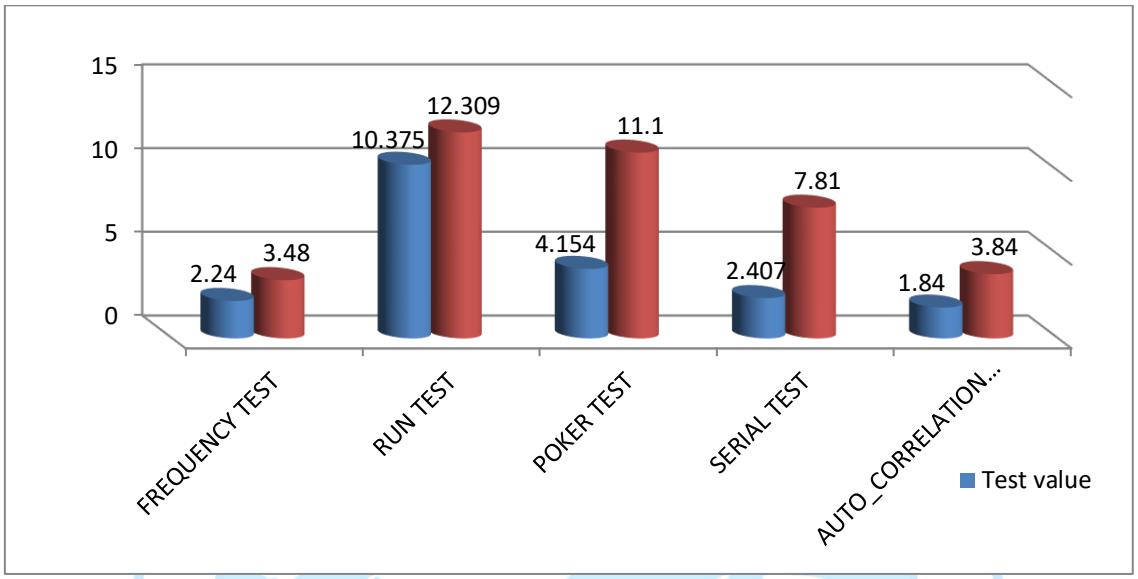



**Figure 8**: Chart of Statistical Test Valves

## Discussion

We notice that is Man in the Middle attack has been removed. That's where the voyeur person even if he obtains the private and public key, can't get all four secret keys, thus he will not know the technique that used for generating other keys from secret key, so for the presence of hash function according to the one-way specialty of this function, It adds complexity to the algorithm because most of the key component features in SHA-2 (SHA256) provide better security than other hash functions, and a non-inverse function that led to the attacker can't retrieve original key value .

In the case of breaking the function by trying all possible keys (Brute force attack), the four keys generated strengthens the numbers for the characters before the start of encryption. And the use of the four classic encryption methods which includes (shift cipher, Vigen`ere cipher, permutation, multiply cipher), although preliminary and distributed on three levels within the matrix, making the possibility of breaking the encrypted text more difficult, only by knowing those methods or classical encryption techniques.

## Conclusion

The proposed system has applied a solid approach for exchanging keys in (D-H) protocol and its algorithm is efficient and reliable because of the required processing at the same time by texts encryption has been done wherever the computational time for the proposed cryptosystem is acceptable when takes a few milliseconds as actual execution time for encrypt & decrypt messages reached to (5 milliseconds). Table 2 with figure 8 Demonstrates how powerful the system is in preventing blade breakage, which can use more robust methods in the future.

## References

1. M. Bellare, P. Rogaway, Introduction to modern cryptography (Ucsd Cse, 207, 2005).
2. J. Hoffstein, J. Pipher, J. H. Silverman, An introduction to mathematical cryptography, Vol.1, (Springer, New York, 2008), pp. 59.
3. J. L. Tsai, IJ Network Security ,9 (1), 12-16 (2009).
4. Y. Han, X. Gui, X. Wu, International Journal of Innovative Computing, Information and Control, 6, 3621-3630 (2010).
5. J. Liu, S. Zhong, L. Han, H. Yao, International Journal of Innovative Computing, Information and Control, 7, 3295-3301 (2011).
6. M. Zhang, X. Wu, Y. Han, Y. Guo, Secure group communication based on distributed parallel ID-based proxy re-encryption, In: Proceedings of the 32nd Chinese Control Conference, pp. 6364-6367. IEEE, 2013.
7. T. D. DeRose, M. Lounsbery, J. Warren, Multiresolution analysis for surfaces of arbitrary topological type. (Department of Computer Science and Engineering, University of Washington, 1993).
8. C. D. Peer, D. Engel, S. B. Wicker, Hierarchical key management for multi-resolution load data representation, In: 2014 IEEE International Conference on Smart Grid Communications (Smart Grid Comm). IEEE, 2014.

**9.** F. Knirsch, G. Eibl, D. Engel, EURASIP Journal on Information Security, 2017 (1), 6 (2017).

**10.** G. Sheikholeslami, S. Chatterjee, A. Zhang, Wavecluster: A multi-resolution clustering approach for very large spatial databases, In: Proceedings of the 24th (1998) VLDB Conference New York, USA, vol. 98, pp. 428-439, 1998.

**11.** W. Diffie, M. Hellman, IEEE transactions on Information Theory ,22 (6), (1976).

**12.** J. Katz, A. J. Menezes, P. C. Van Oorschot, and S. A. Vanstone. Handbook of applied cryptography (CRC press, 1996)

**13.** M. Singh, D. Garg, Choosing best hashing strategies and hash functions, In: 2009 IEEE International Advance Computing Conference, IEEE, 2009.

**14.** R. P. Naik, N. T. Courtois, Optimising the SHA256 Hashing Algorithm for Faster and More Efficient Bitcoin Mining, MSc Information Security Department of Computer Science UCL ,1-65 (2013).

**15.** S. Mathew, P. K. Jacob, A New Fast Stream Cipher: MAJE4, In: 2005 Annual IEEE India Conference-Indicon, Chennai, India, 60-63 (2005).