International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2015): 6.391

An Adaptive Use of Deffie Hellman Algorithm by Digital Images for Generating Secure Shared Key

Zainalabideen Abdullsamd Rasheed

University of Kufa, faculty of Education, Department of CS

Abstract: ***Security is one of the most imperative commentaries in contemporary era. Especially, in terms of creating a safe channel. Create secure channel means the protection of information against an invader during exchanging the data through channel. Many procedures suggested overcoming this problem like cryptography, steganography, biometrics ….etc. In this paper, the propose system consist of two stages: the first is generated key called image key from an image while the second stage is applied Diffie Hellman based on the key which is created from the first step. The image key in our system is used instead of the two numbers which are needed compulsory in Hilleman algorithm. In our proposal two client will not need to agree previously about the two numbers (prime and integer) which are basically the requirements in d,h algorithm to generate the shared key which should be used in ciphering process in advance. Moreover, the exchange key as a part of Diffie Hellman algorithm will be exchanged between parties through hiding it in own copy image of each party in specific position which are selected based on image key that are created in the first stage .The hiding process support a secure channel for two parties to exchange their keys securely. Moreover, support securely channel is done with keeping quality of stego- image due to limitation pixels which are used for hiding process.in addition the hiding process is applied in high bit plane 8th lsb to make it immune against channel effected like noise or compression. The system has been proved truly and the image key generated completely and exchange keys exchanged truly even if the channel has some effect on image like noise or compression.***

Keywords: image key, shared key, exchange key, asymmetric, prime number

1. Introduction

Provision of digital media is one of the key bases for the internet popular. That is principal to exchanging diverse information between societies.in terms of allocation information over an open sources like the internet ,the demanding of generate a protected communication is exceedingly vital necessity. Cryptography is one of the supreme significant technologies which are used in terms of encryption data [1].

Cryptography is the art of ciphering information in away become the information is unreadable. Also Cryptography can be defined as a technique, in which secret messages (unreadable) are transported from one party to another over the communication channel and only the authorize party can decrepit it and read the real message. The cryptography system includes two processes encryption and decryption. an encryption is applied an algorithm on information to make it unreadable to anyone except those owning singular knowledge (key).While the reverse process is referred to as decryption. There are two foremost algorithmic methods to encryption information, symmetric and asymmetric. Symmetric-key algorithms are type of algorithms for cryptography that use the same keys (private key) for both encryption plaintext and decryption cipher text .on other hand, Asymmetric or Public key encryption is an encryption method such as the plain text is encrypted with a public key and cipher text decrypted by private key I.e., the key that kind of public drive be spread to all users, whereas the key that kind of private leftovers secret.

Recently, plain text is not only text can be any type of digital data like text, voice, image…etc. Ciphering algorithms can be classified with respect to the mode of operation of the algorithms into block and stream cipher. A block cipher is a type of symmetric key algorithm that transforms a fixed length of plaintext into the same length cipher text, while stream cipher typically operates on smaller units of plain text, usually bits. Image ciphering can be done either by symmetric or asymmetric algorithms,[2][3][4][5].

Each ciphering system consists of two stages as illustrated in fig. (1)

1- In Cipher stage, the source will cipher an original image to cipher image through apply secret key with the suggested appropriate cipher algorithm.

2-in Decipher stage, the receiver will relocated the cipher image to original image through apply the same secret key with the appropriate decipher algorithm. Nevertheless, decipher will effort in reverse elegance than cipher algorithm. However, in asymmetric ciphering the cipher phase will applied by public key and deciphering phase will applied by private key [6][7].

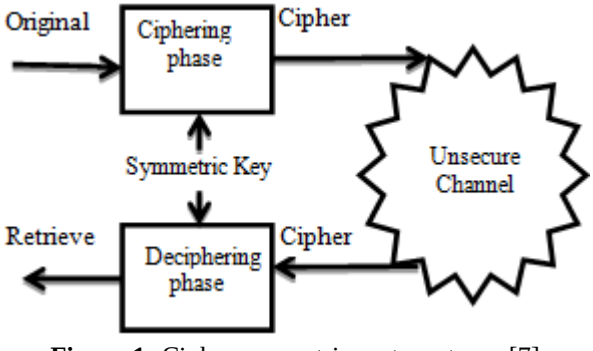

Figure 1: Cipher symmetric system stages [7]

1.1 Public Key Cryptosystem

There are two different keys been used in public cryptosystem; secret key and public key .the first one is kept secret by the holder while the pubic key is well-known. When the different Keyes is used in encryption and decryption process the system called asymmetric. As long as encryption process is done by the public key to encrypted data the decryption process is applied only by cross ponding private key to decrypted data and vice versa. In public key cryptosystem the parties don't need to have a shared secret between them. This cracks the difficult of huge close communication net presented former.

Asymmetric encryptions are computationally far supplementary costly, but the public key can be spread to everybody; fair one party has to picket the undisclosed part of the key [8][9].

1.2 RSA Algorithm

Two Keys are used separately for encryption and decryption processes in public key cryptosystem. However public key is exposed and available for all handlers while private key is keep secrete only for official handler. For that reason the process of transmission secret key is not valuable. That leads to make a public key cryptosystem a very suitable approach for keys administration .RSA is one of the best public key cryptosystem and is commonly used in repetition. Algorithm can be explained in the following steps

The following steps Show how the keys are generated in this algorithm:

1) The two large prime number will be selected randomly (p and q)
2) Find n where n =p*q
3) Find $\phi(n) = (p-1)(q-1)$, where $\phi(n)$ is Euler function of n
4) Select an integer e, such that $1 < e < \phi(n)$ and greatest integer value ($\phi(n)$,e) = 1
5) The encryption process applied by the private key {e,n}.while the decryption process will be applied by private {d,n},where d is the decryption key. Such d is intended by {(d*e) mod $\phi(n)$ =1}.Where mod is modular arithmetic operation. [9][10][11].

1.3 Diffie Hellman Algorthim

The second important algorithm of public key cryptography is Diffie Hellman key exchange.it was created by Whitfield Diffie and Martin Hellman. In 1976, it was the first applied technique for creating a shared secret above an unsafe communication channel. The idea of this procedure is to create shared key that two revelries agree on it and can they custom for a symmetric encryption, such that a listener cannot gain the key. The algorithm can be explained in the following steps:

First, a prime number p and an integer g should be greed between two parties.party1 and party 2.Secondly, for each one of them must randomly pick secret number. Let assume, party1 select a and pary2 select b. Then, party1 will calculate (ga mod p) and send it to the party2.similarly, party2 will calculate (gb mod p) and send it to party1.finaly, party1 compute shared key ((gb mod p)a mod p) and party2 will compute shared key ((gb mod p)b mod p).The shared key for two parties is same and can be used for ciphering and deciphering process. However, exchange key between two parties need secure channel.as well as generate a large prime number is a big challenge. A system founding an encryption key which is used to encrypt data between clients and the system is not itself for encryption data.it is working deeply in collective networking locations and conversation platforms for encryption procedures, [12][13][14].

2. Proposed System

The proposed system in cipher process consists of two stages and the same stages in deciphering process but in reverse. (The threshold is an agreement between two clients and also type of image key):

The generating public key from the image as illustrated in figr. (2)

1) Original Image

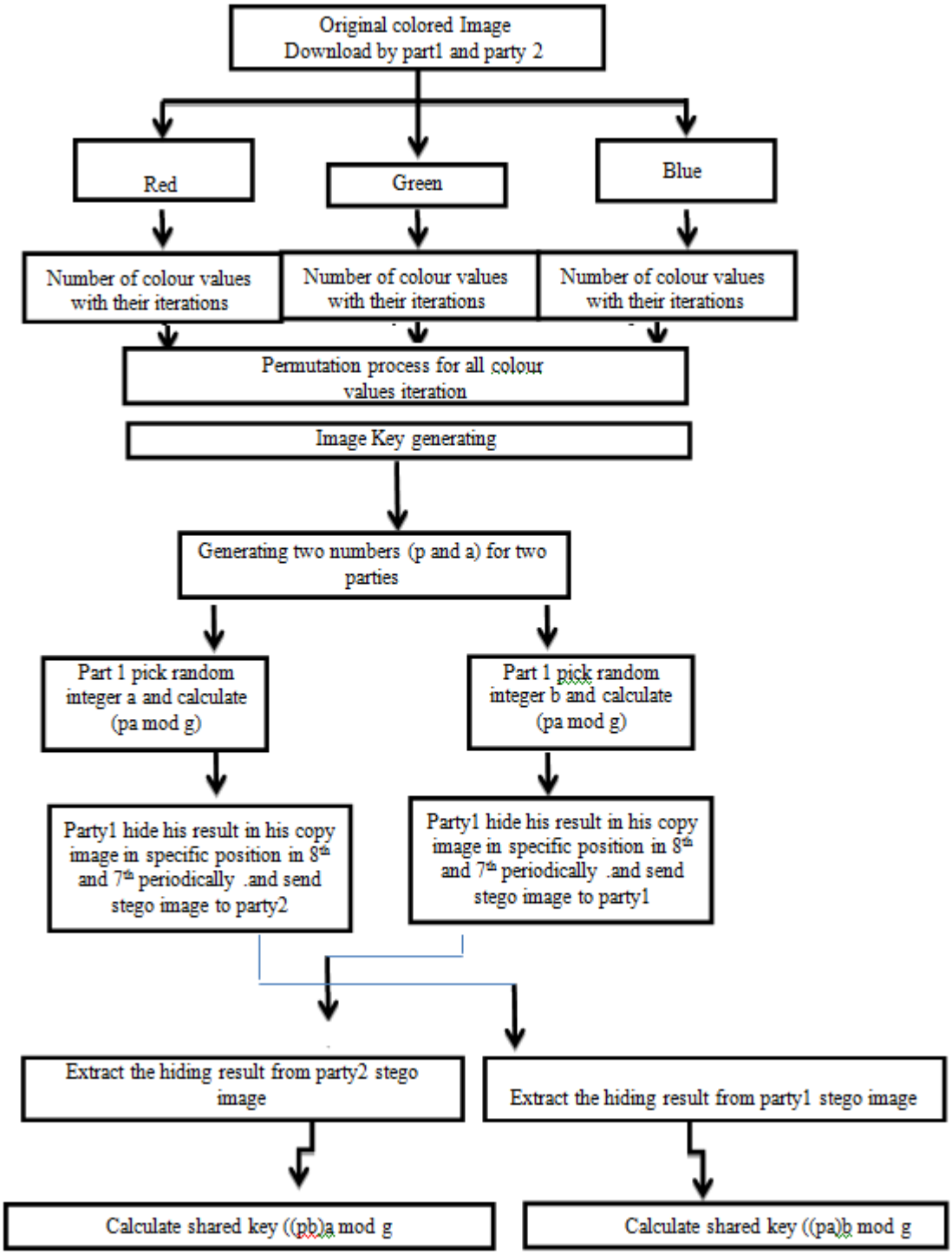

Figure 2: Proposed System

The first stage will be same for two parties to generate image key and two numbers prime p and integer the second stage will be also same from two parties, however each one of them will choose private a random integer and calculate his value and hide it in his copy image and send to other.

As shown from figure above a key is generated from original color image by creating a table of the number of color values and iterations of each band (read, green and blue). Then permutation process is applied on this table to give it heartier. The permutation process applied on maximum iterations to change its color values (I.e. change the index of iteration in the table).Later, color values of maximum iterations of each band are selected and write as digit together and it is called image key.

The last step of first stage is to generate the two numbers (prime p and integer g) from image key which is required for Diffie Hellman Algorithm.

The second stage is to exchange their keys party1 and party2 .

So party1 and party2 will choose randomly two integer numbers (a & b).then, party1 will compute (g^a mod p). And party2 will compute (g^b mod p). After that, each one of them should send his result to the other through his copy of image.

They will hide their result in own copy of image. The image key which is generated in first stage will be used to determine the position pixels of image which be used from both party1 and party2 to hide their results in their own copy of image.
Later, each one will send own copy of stego image to other in terms of exchanging their eyes. Number of color values of each band =255/threshold. The number of color values of maximum iteration = number of color values of three bands/threshold. The new index of iteration = (old index * the number of color values of maximum iteration) mod (Number of color values of one band). As long as threshold decrease number of color values increase

Image key can be generated for any type of non-binary image. However for non-color image the number of values will be generated only for one band .but not recommended.

3. Test and Result

The main text for your paragraphs should be 10pt font. All body paragraphs (except the beginning of a section/sub-section) should have the first line indented about 3.6 mm (0.14"). For testing. Image (flower and Penguin) with size (400*400*3 unit 8) has been used.in mat lab and the threshold has been used it 10.

Below the whole explanation of our proposed; include two stages:

The first stage is shown as below:

Number of color values =255/threshold=255/10=25 colors values for each band.

Table 1: The 75 iterations of three bands color values in order (red, green and blue) flower.png.

16533	14373	6250	8663	7117	6559	27179	5752	4037	3861	7507
3808	3259	18282	15529	3351	3533	4341	5200	5780	5469	5525
6414	13643	12334	7050	6225	3645	5034	7138	10929	12457	3069
10043	9086	6484	5864	6034	5852	5251	3692	3046	2694	2062
4530	4043	5290	5353	4284	4838	5895	2816	7051	4181	2940
2794	2449	2409	2731	5652	5629	3940	4347	4375	4380	4388
4430	3592	4762	3224	5757	6154	6972	8644	12196	Null	null

The number of color values of maximum iteration = (number of color values *3)/threshold= (25*3)/10=7 color values of maximum iteration.

The image key = the number of color values of maximum iteration *color values of maximum iteration (write as digit). The integer result it takes from all division operation.

The 25 colors values is constricting by divide the whole gray level (255) into 25 color values as below:

[5,15,25,35,45,55,65,75,85,95,105,115,125,135,145,155,165 ,174,185,195,205,215,225,235,245]

Below the 75 iterations of three bands color values in order (red, green and blue); the first 25th for red band and the second 25th for blue band while the 3rd 25 for green band and the index is in order from 1 to 75 as shown below. For flower.png

Table 2: The 75 iterations of three bands color values after permutation for flower.png.

12196	6484	6250	8663	7117	6559	5895	5752	4037	3861	7507
3808	3259	2816	3224	3351	3533	4341	5200	5780	5469	5525
6414	7050	12334	12457	6225	3645	5034	7138	10929	13643	3069
10043	9086	14373	5864	6034	5852	5251	3692	3046	2694	2062
4530	4043	5290	5353	4284	4838	27179	18282	7051	4181	2940
2794	2449	2409	2731	5652	5629	3940	4347	4375	4380	4388
4430	2592	3762	15529	6757	6154	7972	8644	16533	null	Null

The permutation process will be applied on table above to swap the maximum iteration (swap mean change the index of maximum iteration in table .the number of maximum iteration will be swapped is 7 iterations and below the permutation process of it.

The new index =old index *7 mod 25, if the new index is occupied we will add 25 to the new index.
The new index (27179) =51*7 mod 25=7 is the new index in the table.
The new index (18282) =52*7 mod 25=14 is the new index in the table
The new index (16533) =75*7 mod 25=0 is the new index in the table is 1
The new index (15529) =70*7 mod 25=15 is the new index in the table
The new index (14373) =36*7 mod 25=2 is the new index in the table
The new index (13643) =32*7 mod 25=24 is the new index in the table
The new index (12457) =26*7 mod 25=7 is the new index will be 7+25=32 because index 7 is
Below the 75 iterations of three bands color values in order (red, green and blue); the first 25th for red band and the second 25th for blue band while the 3rd 25 for green band and the index is in order from 1 to 75 as shown below. For flower.png after permutation process

Table 3: The 75 iterations of three bands color values in order (red, green and blue) flower.jpg

12178	6507	6253	8659	7116	6557	5877	5774	4049	3865	7517
3815	3275	2809	3236	3382	3569	4348	5276	5852	5632	5619
6705	7315	11253	12449	6257	3685	4997	7095	10852	13638	3226
10057	9070	14371	5872	5990	5875	5290	3693	3025	2739	2090
1522	1038	664	348	256	808	27268	18336	7062	4263	3014
2794	2474	2444	2772	2633	5687	2595	2387	2366	2396	2376
2458	2628	2823	14756	3874	4440	5335	4199	15546	null	null

It is shown from table 3 although there is some change in iteration in flower.jpg but the index of maximum iterations is still same and the whole process of permutation is same as the previous process in flower.png. Below the table after permutation the index of maximum iterations of image.jpg

Table 4: The 75 iterations of three bands color values after permutation for flower.jp

14546	13756	6253	8659	7116	6557	27268	5774	4049	3865	7517
3815	3275	18336	12178	3382	3569	4348	5276	5852	5632	5619
6705	13371	11253	7315	6257	3685	4997	7095	10852	12638	12449
10057	9070	6507	5872	5990	5875	5290	3693	3025	2739	2090
1522	1038	664	348	256	808	5877	2809	7062	4263	3014
2794	2474	2444	2772	2633	5687	2595	2387	2366	2396	2376
2458	2628	3236	3226	3874	5440	7335	9199	2823	null	null

The last steps of first stage are to generate image key and two numbers (prime and integer) as shown below:

1) The image key = the number of color values of maximum iteration *color values of maximum iteration (write as digit).
2) The image key =7*(6513551451523565) =45594860160664955
3) The first 7 digits is used to generate prime p while the second 7 digits is used as integer g The first 7 digits (4559486) and it are not a prime there for the previous prime number will be chosen and it is 4559483. And the second 7 digits is an integer g =1606649.Now two parties generate tow number (p and g).p=4559483 and g=1606649

The second stage of ours system are exchange keys between parties and this step is same for two images flower.png and flower.jpg

1) Party1 choose a=5; party2 choose b=8
2) Party1 compute (p^a mod g) =) 4559483(^{5} mod 1606649=556194
3) Party2 compute (p^b mod g) =) 4559483(^{8} mod 1606649=1399215
4) Party1 will hide (556194) in his own copy of image and send to party2.however, party2 will hide (1399215) in his own copy of image and send it to party1.the hiding process will be in 8th lsb bit plane in specific position pixels will determine based on image key.
5) After that, party1 receive an image from party2 and truly extracted this number (1399215).while party2 receive an image from party1 and extracted this number (556194).later.pary1 and party2 compute shared key.
6) Party1 compute($(1399215)^5$ mod 1606649)= 187804.
7) Party2 compute ((556194)^8 mod 1606649)= 187804. The shared key is 187804.this key can be used for ciphering and deciphering process in advance.

Below flower image with two types (flower.png and flower. jpg however all types of image has been proved and truly worked

Figure 3: All types of flower images

The second image has been applied in our testing is Penguin.png and Penguin.jpg and below all tables and images.

Table 5: The 75 iterations of three bands color values before permutation for Penguin.png

26449	3107	2126	2023	1727	1853	1873	33255	1904	2498	7266
11180	14328	11147	8771	8935	5506	7981	6843	5735	3425	3284
6944	6052	8569	4100	10350	11092	15698	16226	2157	4222	13893
9643	8168	4527	3239	3356	18044	1997	2280	2185	2410	5375
3969	7494	7107	6100	5356	5129	17591	2548	11074	6126	5340
2140	2786	6205	4333	4195	3941	2533	2953	2784	2962	2440
4317	5928	5969	3964	5431	2832	1111	1839	1760	null	Null

Table 6: The 75 iterations of three bands color values after permutation for Penguin.png

1873	3107	6100	2023	1727	33255	26449	1853	1904	16226	7266
11180	8935	11147	8771	2280	5506	7981	6843	5735	3425	3284
18044	6052	8569	4100	10350	11092	2126	2498	13893	17591	2157
9643	8168	4527	3239	3356	6944	1997	14328	2185	2410	5375
3969	7494	7107	15698	5356	5129	4222	2548	11074	6126	5340
2140	2786	6205	4333	4195	3941	2533	2953	2784	2962	2440
4317	5928	5969	3964	5431	2832	1111	1839	1760	null	null

Table 7: The 75 iterations of three bands color values before permutation for image Penguin.jpg

25316	3657	2393	1896	1841	1988	1905	32165	2007	2358	7335
10818	13773	11120	9015	8570	5599	8108	6940	5620	3432	3218
7103	6734	8230	4532	9993	10608	14795	15784	1812	4415	12877
9387	8295	4689	3372	3205	17335	2178	2281	2128	2396	5354
4002	7357	7194	5991	5353	5377	16333	2601	8499	6256	5219
2406	3009	6286	5726	3877	4141	3427	2998	2804	2795	4483
4318	5961	6148	5423	4871	4197	2115	1962	2294	null	null

Table 8: The 75 iterations of three bands color values after permutation for Penguin.jpg

1905	3657	2393	1896	1841	32165	25316	1988	2007	15784	7335
10818	13773	11120	9015	8570	5599	8108	6940	5620	3432	3218
17335	6734	8230	4532	9993	10608	5991	2358	1812	16333	12877
9387	8295	4689	3372	3205	7103	2178	2281	2128	2396	5354
4002	7357	7194	14795	5353	5377	4415	2601	8499	6256	5219
2406	3009	6286	5726	3877	4141	3427	2998	2804	2795	4483
4318	5961	6148	5423	4871	4197	2115	1962	2294	null	null

The image key =7*(55652256595225155) =389565796166576085

Now the first 7 digits is used to generate prime p while the second 7 digits is used as integer g

The first 7 digits (3895657) and it is not prime there for the previous prime number will be chosen and it is 3895603
The second 7 digits is an integer g =9616657

Now two parties generate tow number (p and g).p=3895603 and g=9616657. The second stage party1 choose a=85; party2 choose b=137

Party1 compute ($p^a \bmod g$) $=(3895603)^6 \bmod 9616657=678542$
Party2 compute ($p^b \bmod g$) $= (3895603)^9 \bmod 9616657= 1864652$

Party1 will hide (678542) in his own copy of image and send to party2.however, party2 will hide (1864652) in his own copy of image and send it to party1.

After that, party1 receive an image from party2 and truly extracted this number (1864652).while party2 receive an image from party1 and extracted this number (678542).later.pary1 and party2 compute shared key.
Party1 compute $(1864652)^6 \bmod 6716298) = 7263254$
Party2 compute $(678542)^9 \bmod 6296366) = 7263254$. The shared key is 7263254 which can be used for ciphering and deciphering process in advance

Below penguin image with two types (Penguin.png and Penguin. Jpg) however all types of image has been proved and truly worked

Figure 4: All types of flower images

For more proving The system continue for using the shared key in ciphering and deciphering process in terms of more testing for a shared key and below the ciphering images for flower (jpg and png)and the system has been proved correctly and the images is totally has retrieved for both type png and jpg.


Ciphering Forwer.png


Ciphering flower.jpg


Deciphering flower.png


Deciphering flower.jpg

4. Conclusion

First of all images key generating is truly generated for all types of images and as shown from table (1, 3,5 , 7) index of maximum iteration is still same for two types of images(png and jpg).although some change in images .jpg as shown in table 3 and 7 but the index is till same. As shown from stego images for flower and penguin the quality of all of them are over 90 percent due to use less change in images because very few pixels are used.

1) The exchange key which should be exchange between parties has be applied through very secure way by hiding it in image in way make it un seen for intruders to flow it instead of sending exchange key in un secure channel in traditional D.H algorithm.
2) The two numbers prime and integer will not be public for all or agree about it previously which basically requirements for traditional D.H algorithm but will generate it from image itself as result make it more protected.
3) Generated big prime number is a challenge and in our proposal the prime number is generated from the image itself
4) The image key which is used in our proposal to determine the position pixels have been proved is completely generated from image even if image has effected through channel like noise or compression because the way of generating is based on iterations color.
5) Xchange keys between two parties has been proved also and totally true extract from two parties even image has effected through channel due to hid in very high bit plane 8^{th} with keeping quality of stego-image. Keeping quality as result of very few pixels has been used for hiding due to small size of key exchanges.

References

[1] Stinson, Douglas R. *Cryptography: theory and practice*. CRC press, 2005.
[2] Disina, Abdulkadir Hassan. *ROBUST CAESAR CIPHER AGAINST FREQUENCY CRYPTANALYSIS USING BI-DIRECTIONAL SHIFTING*. Diss. Universiti Tun Hussein Onn Malaysia, 2014.
[3] Windley, Phillip J. *Digital identity*. " O'Reilly Media, Inc.", 2005.
[4] Singhal, Nidhi, and J. P. S. Raina. "Comparative analysis of AES and RC4 algorithms for better utilization." *International Journal of Computer Trends and Technology* 2.6 (2011): 177-181.
[5] Aïssa, Belmeguenaï, Derouiche Nadir, and Mansouri Khaled. "Security Analysis of Image Cryptosystem Using Stream Cipher Algorithm with Nonlinear Filtering Function." *Editorial Preface* 3.9 (2012).
[6] Kwang, H. Lee. "Basic Encryption and Decryption." *Department of Electrical Engineering & Computer Science, KAIST* (2000).
[7] Sharif, Suhaila Omer, and S. P. Mansoor. "Performance analysis of stream and block cipher algorithms." *2010 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE)*. Vol. 1. IEEE, 2010. Image Ciphering With enhancement DH by generate public Key from an Image
[8] Goldwasser, Shafi, and Mihir Bellare. "Lecture notes on cryptography."*Summer course "Cryptography and computer security" at MIT* 1999.
[9] Ganesan, Ravi. "Yaksha, an improved system and method for securing communications using split private key asymmetric cryptography." U.S. Patent No. 5,535,276. 9 Jul. 1996
[10] El-Deen, A., E. El-Badawy, and S. Gobran. "Digital image encryption based on RSA algorithm." *J. Electron. Commun. Eng* 9.1 (2014): 69-73.
[11] Sheng, Yuan, et al. "Information hiding based on double random-phase encoding and public-key cryptography." *Optics express* 17.5 (2009): 3270-3284.
[12] Yuan, Sheng, et al. "Simultaneous transmission for an encrypted image and a double random-phase encryption key." *Applied optics* 46.18 (2007): 3747-3753.
[13] Kaur, Navpreet, and Ritu Nagpal. "Authenticated Diffie-Hellman Key Exchange Algorithm."
[14] Rescorla, Eric. "Diffie-Hellman key agreement method." (1999).
[15] Van Oorschot, Paul C., and Michael J. Wiener. "On Diffie-Hellman key agreement with short exponents." *International Conference on the Theory and Applications of Cryptographic Techniques*. Springer Berlin Heidelberg, 1996.