International Journal of Applied Information Systems (IJAIS) – ISSN : 2249-0868
Foundation of Computer Science FCS, New York, USA
Volume 3– No.1, July 2012 – www.ijais.org

Formal Method for Cryptography using DES – Tables

Saad Abdual azize AL_ani
PhD, Associate Prof.
Computer Science Department
Al_Mamon University College
Baghdad - Iraq

Bilal Sadiq Obaid Al-Kubaysi
Computer Sciences and Informatics College
Amman Arab University
Amman - Jordan

ABSTRACT
In this paper, a new approach is presented to develop the DES algorithm. Where DES is un-linear complex encryption system able to encrypt the information in high speed once applied onto entities that manipulate it. DES – Table is the new schema that uses two tables to encrypt the input as 64 bits and the out is 56 bits of cipher, and then use two inverse tables to decrypt the cipher to return the input from 56 bits of cipher to 64 bit of plaintext. By this schema we achieve emboldening results with maximum possible of confusion, non-obviousness and complexity of ciphering.[1]

Keywords
Cryptography, DES - Table, Encryption, Decryption, Network Security, Computer Security, IP Table, IP^{-1} Table, Exp, Exp^{-1}, Cipher, Plaintext, key, ASCII, Decimal, XOR.

1. INTRODUCTION
Network security of computer is an urgent need to protect the information and data that carried out over networks. This requirement became a pressing because the most of threats and risk come from many and many source of networks. So, build and develop the security systems is one of critical task to save the data that became related with every day's life.[2]

Good security system means that the system that protect an applications, systems and users from attackers that originated from inside and outside networks. As mentioned earlier, the main target of security system is to protect the information; this means the information that received is the same that had been sent without any changes or corruption.

1.1 Objective of Network security
In network, movement and exchange of data is most process occurred. So, ensuring the data sent from sender to receiver without any corruption, delaying, losing or stolid from any to make the data is trusted intruder is basic function for any network security system. The power and speed of network security should be considered when create any protection security.[3]

1.2 DES - Table
In this paper, a new developed approach and a new methodology are added to DES algorithms, it is called 'DES - Table'.

DES – Table has derived from DES methodology, but we execute new techniques and development that give us a big power of encryption which made the problems and challenges are very difficult for hackers or any threat source. It gave cryptosystem designers new ideas, think by new methods to produce new versions of cryptosystem especially with DES. We need that to make maximum possible satisfying of security system specifications. In DES – Table approach, we will generate sets of special organized keys as Tables, where the input or Plaintext will be block in 64 bit (8 characters), and the output – cipher text – will be 56 bit as length. In the new algorithm, the cipher text will be decrypted to produce the plaintext using inverse of very had keys that it's very difficult to exactly understand how it had been built.

2. METHODOLOGY
In this article, the input is in binary mode as it considered in DES algorithm. Whereas it known, each character represents 8 bits has ASCII code started from 65 of 'A' and end with 90 of character Z. note that the last ASCII code for last character symbol is 127 (other character symbol such as @, small letters as 'm', 'space', $,], … etc). Then, convert the ASCII code for each character to 8 – bit binary. Then, match each character with his binary 8 bit mode of his ASCII code, and create new table to produce from locations values between the key table and 8 bit binary array to find the valued of the array. Finally a new key will make reconstruction of the array. All these points and others will be mentioned and displayed in algorithm and implementation sections.

3. ALGORITHM
The following is steps for encryption

1) Consider the plaintext.
2) Divide the input as 8 bit character size.
3) Convert each character of input to ASCII code.
4) Find the 8 bit binary of each ASCII code that contains 64 bit.
5) Broadcast the 64 bits of the step 4 array into IP table.
6) Write row by row to get a new 64 bits.
7) Broadcast the 64 bits array of step 6 into Exp table.
8) Write row by row to get one dimension array of 56 bits.

9) Apply the XOR properties between *k* and the array. The output will be the cipher bits. Here, the considered key is SECURITY word.

IP =

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

EXP =

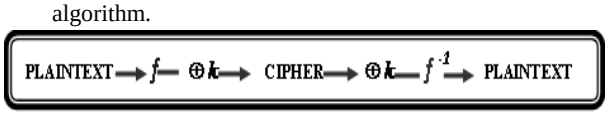
64	63	62	61	60	59	58	57
56	55	54	53	52	51	50	49
48	47	46	45	44	43	42	41
32	31	30	29	28	27	26	25
24	23	22	21	20	19	18	17
16	15	14	13	12	11	10	9
8	7	6	5	4	3	2	1

the 7 bit of ASCII code of key word "SECURITY" as follows.

k = 10100111000101100001110101011010010100100110101001011001

The following steps for Decryption

1) Apply the XOR properties between *k* and the cipher bits.
2) Broadcast the inverse array bits onto exp table.
3) Build the inverse of exp table to input the cipher inverse array and produce inverse table.
4) Add zero rows in inverse table.
5) Apply XOR properties between the ASCII 8 bit codes of the *key* with the inverse table.
6) Produce the inverse key of IP (IP^{-1} table).
7) Broadcast the inverse table into IP^{-1} table to find the plaintext. The follows are DES – Table equation and flowchart to view the scenarios of DES – Table algorithm.

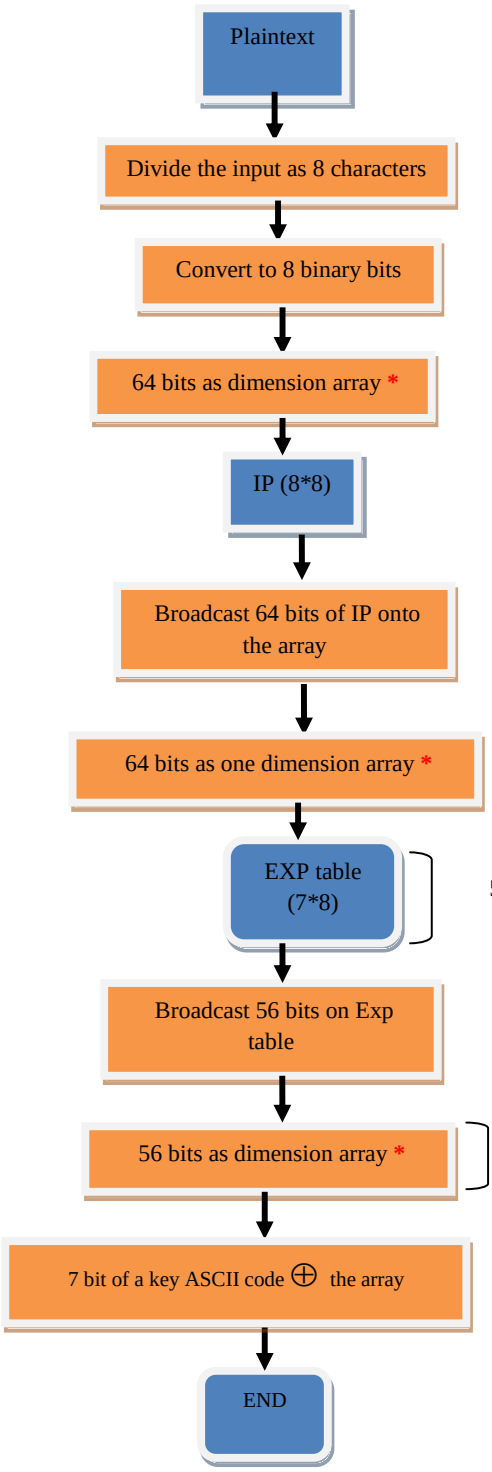


IP^{-1} =

40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

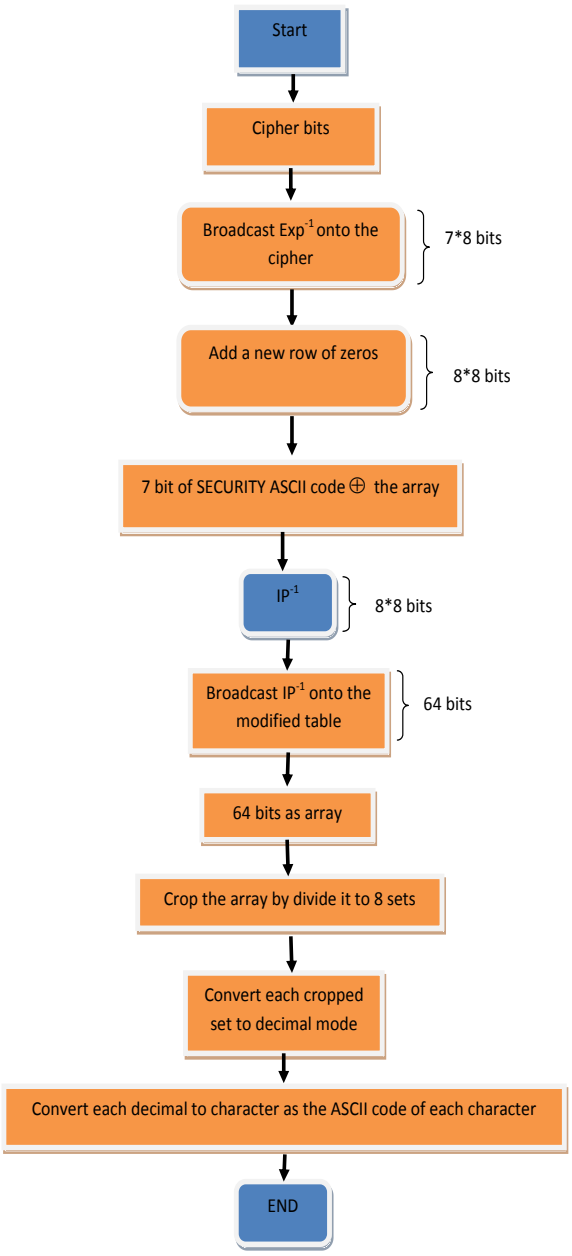
Flowcharts

- Encryption Flowchart



- In this point, we reshape the output of the previous step from table to array by write it row by row

- Decryption Flowchart



4. IMPLEMENTATION

Here, to show how implement the methodology, we consider one simple sample to more understand and clear for encryption of DES – Table algorithm

1. Plaintext = computer

h1=

c	99	1	1	0	0	0	1	1
o	**111**	1	1	0	1	1	1	1
m	**109**	1	1	0	1	1	0	1
p	**112**	1	1	1	0	0	0	0
u	**117**	1	1	1	0	1	0	1
t	**116**	1	1	1	0	1	0	0
e	**101**	1	1	0	0	1	0	1
r	**114**	1	1	1	0	0	1	0

2. Convert the 8 character to 64 bits

h2 = 0110001101101111011011010111000001110101011101000110010101110010

IP =

58	50	42	34	26	18	10	2
60	**52**	**44**	**36**	**28**	**20**	**12**	**4**
62	**54**	**46**	**38**	**30**	**22**	**14**	**6**
64	**56**	**48**	**40**	**32**	**24**	**16**	**8**
57	**49**	**41**	**33**	**25**	**17**	**9**	**1**
59	**51**	**43**	**35**	**27**	**19**	**11**	**3**
61	**53**	**45**	**37**	**29**	**21**	**13**	**5**
63	**55**	**47**	**39**	**31**	**23**	**15**	**7**

Then, write the output row by row to get h3

h3=1111111110111000011101100101011100000000111111110000011010000011

the output as row by row as one dimension array

Exp =

64	63	62	61	60	59	58	57
56	**55**	**54**	**53**	**52**	**51**	**50**	**49**
48	**47**	**46**	**45**	**44**	**43**	**42**	**41**
32	**31**	**30**	**29**	**28**	**27**	**26**	**25**
24	**23**	**22**	**21**	**20**	**19**	**18**	**17**
16	**15**	**14**	**13**	**12**	**11**	**10**	**9**
8	**7**	**6**	**5**	**4**	**3**	**2**	**1**

h4 = 1100000101100000111111111110101001101110000111011111111111

5. apply the X-OR properties between arrays of h4 and *k* as follow

h4: 1100000101100000111111111110101001101110000111011111111111

X-OR

K: 1010011100010110000111010101101001010010011010100101100 1

= Ci: 0110011001110110111000101011000000111100011101111010011 0

Now, the Decryption process starts since we end the cipher (ci) as the input

1. Apply the X-OR properties between arrays bits of the Cipher and *k* as follows

Ci= 01100110011101101110001010110000001111000111011110100110

X-OR

K: 10100111000101100001110101011010010100100111010100101100 1

= h5: 11000001011000001111111111110101001101110000111011111111 1

2. Create the inverse of exp and broadcast it onto Exp $^{-1}$ (7*8) table

Exp $^{-1}$ =

1	1	1	1	1	1	1	1
1	**0**	**1**	**1**	**1**	**0**	**0**	**0**
0	**1**	**1**	**1**	**0**	**1**	**1**	**0**
0	**1**	**0**	**1**	**0**	**1**	**1**	**1**
1	**1**	**1**	**1**	**1**	**1**	**1**	**1**
0	**0**	**0**	**0**	**0**	**1**	**1**	**0**
1	**0**	**0**	**0**	**0**	**0**	**1**	**1**

3. Add new row with zero value onto Exp^{-1}

1	1	1	1	1	1	1	1
1	**0**	**1**	**1**	**1**	**0**	**0**	**0**
0	**1**	**1**	**1**	**0**	**1**	**1**	**0**
0	**1**	**0**	**1**	**0**	**1**	**1**	**1**
1	**1**	**1**	**1**	**1**	**1**	**1**	**1**
0	**0**	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**1**	**1**	**0**
1	**0**	**0**	**0**	**0**	**0**	**1**	**1**

Exp^{-1}_0 =

4. Write the Exp $^{-1}$_0 as row by row in one dimension array to get h6 array

5. Create the inverse IP table

h6 = 11111111101110000111011001010111111111110000000000000110100000 11

Broadcasting output =

0	**1**	**1**	**0**	**0**	**0**	**1**	**1**
0	1	1	0	1	1	1	1
0	1	1	0	1	1	0	1
0	1	1	1	0	0	0	0
0	1	1	1	0	1	0	1
0	1	1	1	0	1	0	0
0	1	1	0	0	1	0	1
0	1	1	1	0	0	1	0

6. Then, broadcast h6 into IP^{-1} table

IP^{-1}

40	8	48	16	56	24	64	32
39	**7**	**47**	**15**	**55**	**23**	**63**	**31**
38	**6**	**46**	**14**	**54**	**22**	**62**	**30**
37	**5**	**45**	**13**	**53**	**21**	**61**	**29**
36	**4**	**44**	**12**	**52**	**20**	**60**	**28**
35	**3**	**43**	**11**	**51**	**19**	**59**	**27**
34	**2**	**42**	**10**	**50**	**18**	**58**	**26**
33	**1**	**41**	**9**	**49**	**17**	**57**	**25**

h7 = 01100011011000110010101011100010

7. .

8. divide h7 to 8 parts
9. convert each part to decimal number mode
10. classify the numbers and equalize it to characters as his ASCII code

The plaintext = computer

5. CONCLUSION

As it showed in this paper, we note how we implement the new developed approach that given strong boost for DES cryptosystem by DES – Table schema. This schema makes breaking processes not easy. DES – Table is more lucid for users' point of view. Here we are applying set of Table keys to reconstruct and reshape all values of plaintext, and then convert each entered block (as 64 bit) to 56 bit of cipher, where the output cipher will displayed as maximum rate possible of confusion, non – obviousness and complexity. So, this schema is more applicable for sensitive applications.

6. REFERENCES

[1] William Stallings. "**Cryptography and Network Security**", 2006. Person Education, New Jersey.

[2] WadeTrappe, Lawrence C. Washington. "**Cryptography with Coding Theory**". 2002, Preafice-Hall, New Jersey.

[3] Sean Simmons. "**Algebraic Crypto Analysis of Simplified AES** ", 2004, 33, 4. Proquest Science Journals.

[4] D.R. Stinson, "**Cryptography Theory and Practice**" CRC Press, Inc., 2002.

[5] T. Sheldon and B. Sur, "**Cryptography**", Pan American and International, 2001. http:// www.Linktionary.com/cryptography.htm

[6] F. Piper and S. Murphy, "**Cryptography: A very short introduction**", Oxford university press, 2002.

http:// www.Books24×7.com/Refrenceware for professionals.htm