IOSR Journal of Computer Engineering (IOSRJCE)
ISSN: 2278-0661, ISBN: 2278-8727Volume 7, Issue 5 (Nov. - Dec. 2012), PP 55-60
www.iosrjournals.org

Transaction Security Using Input Based Shared Key Cryptography

[1]Mayank Swarnkar, [2]Shivani Singh, [3]Dr. Shekhar Verma
[1, 2, 3] Indian Institute of Information Technology-Allahabad

Abstract: *Mobile devices are growing day by day, so the mobile database. Transactions from ATM machines are a good example of wireless Transactions. Since these transaction flows using medium as air hence security is a issue till the transaction reaches the Base station from where the transaction is much more secure as compared to wireless network. In this paper a new scheme of securing database transaction is proposed till it reaches the Base station. This is also required to provide security against frequent disconnections [2][5].*

Keywords: *transactions, Shared key Cryptography, Wireless Network, Security.*

I. Introduction

Security is an important issue in mobile database transactions. The general problem that mobile database suffers is frequent disconnections. Hence protection of transactions such as prevention of revealing of identities such as transaction Id's and spoofing AVS(Address verification services) is an important issue. Various models on database transactions have been proposed such as kangaroo model[2], Two tier model, Multi checkout timestamp ordering model[1], security model[3] etc. All models are not so efficient in solving such security issues. Problem of transaction security between Mobile unit and base station has been figured out in this paper. The data set with transaction ID and TAN i.e. Transaction Authentication number are important attributes of transaction and hence to be protected. Therefore a private key encryption[4][5] technique is applied in this paper to make transaction secure between mobile host and base station. The Security replication management is one of the current issues in distributed database that has yet to be solved. This paper focus on the aspects of protecting the transactions from the eavesdroppers present between the Mobile host and the base station.

In the figure 1 as shown above the transaction from mobile host passes to base station is to be made secured. Since mobile banking is very popular now a days. Since public key cryptography[4] requires heavy calculations i.e. processing as well as power. Since mobile devices works on battery hence power becomes a constraint. So in this paper private key cryptography is used which depends on the transaction Id's which acts as input in the Cryptography. Private Key cryptography allows the lesser calculations and it is shared between the two ends. The problem in private key cryptography is that if any of the two parties who share the key reveals the key makes the cryptographic system corrupted, but here both the ends are handled by the same party hence the problem of revealing the secrecy of the private key is also not applicable.

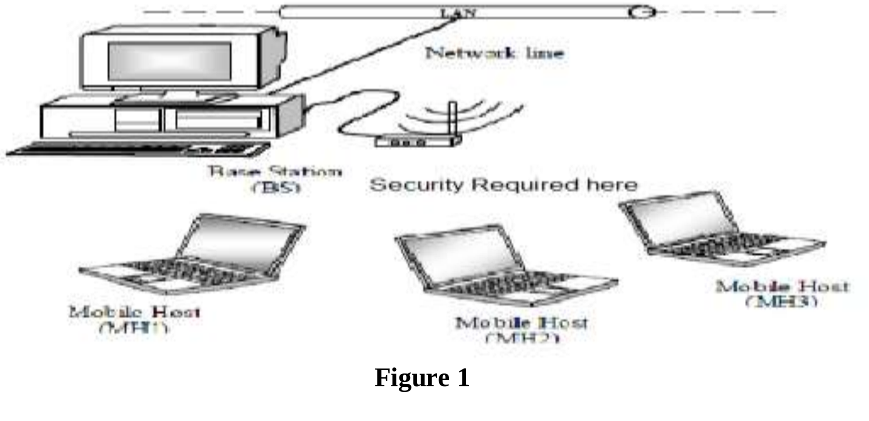


Figure 1

II. Proposed Cryptography

Transaction security consists of both encryption and decryption algorithms that are located at the BS and the MH(s) as shown in Figure 2. The encryption algorithm is started when the data transferred. The decryption algorithm is started when encrypted data is received.

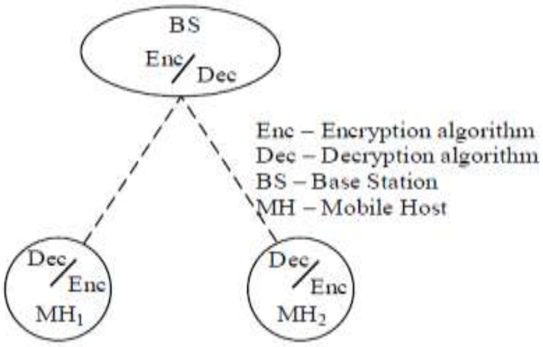
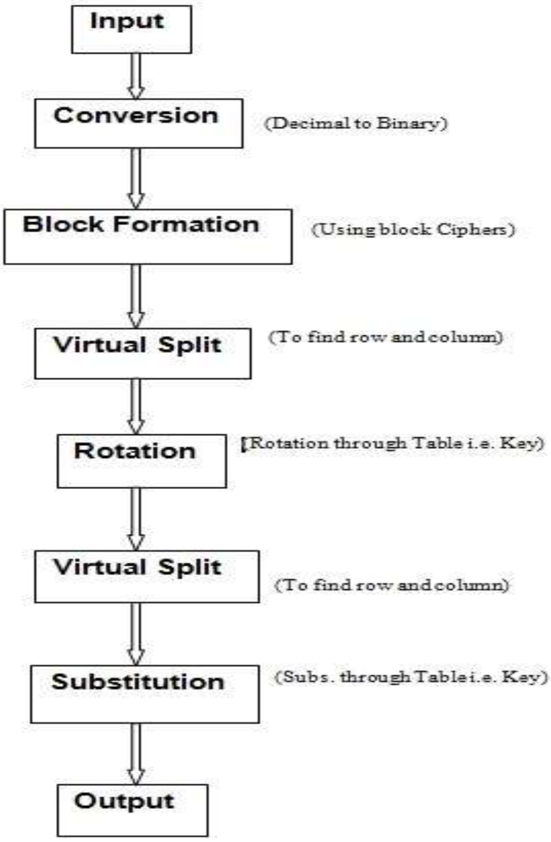


Figure -2

III. Encryption Algorithm

Encryption algorithm consists of the steps: Conversion, Block Formation, Virtual Split-1 , Rotation , Virtual Split-2 and Substitution. These steps are fair enough to provide security against attacks in the wireless networks.

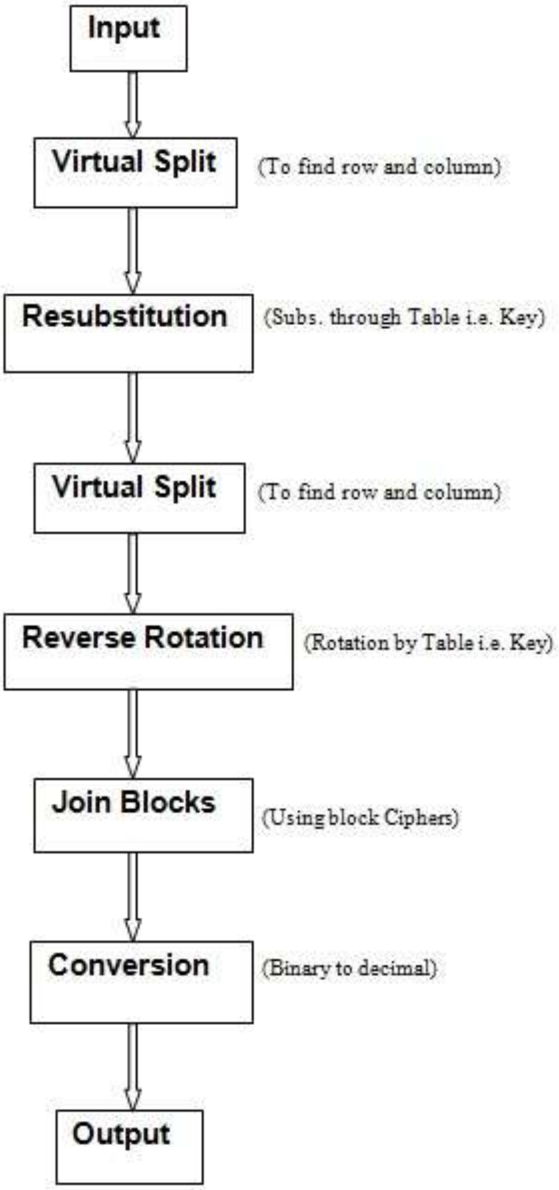
Input which is obtained is first converted from decimal form to the binary form. This step is known as conversion step. Next comes the block formation step. In this encryption scheme Block ciphers are used. Hence the input which is in binary form is converted to fix eight bit blocks. In the virtual split step the eight bit sequence is virtually broken into two four bit sequences. Each bit sequence is converted to respective decimal form. When both the sequence is converted to binary form they represent row number and column number of the Rotation table. This table is one private key used in the encryption algorithm. The Row number and column number finds an operation which is to be performed in the original bit sequence. When the operation of rotation is completed on the bit sequence then one more time the virtual splitting is done. This virtual splitting is same as that was done before. Then the row number and column number obtained from the virtual split matches the unique substitution from the substitution table. This substitution table is the another private key used in this algorithm. The key obtained from substitution table is EX-or with the bit stream to obtain the output . With this step encryption process completed and data is then transmitted in the medium.



IV. Decryption Algorithm

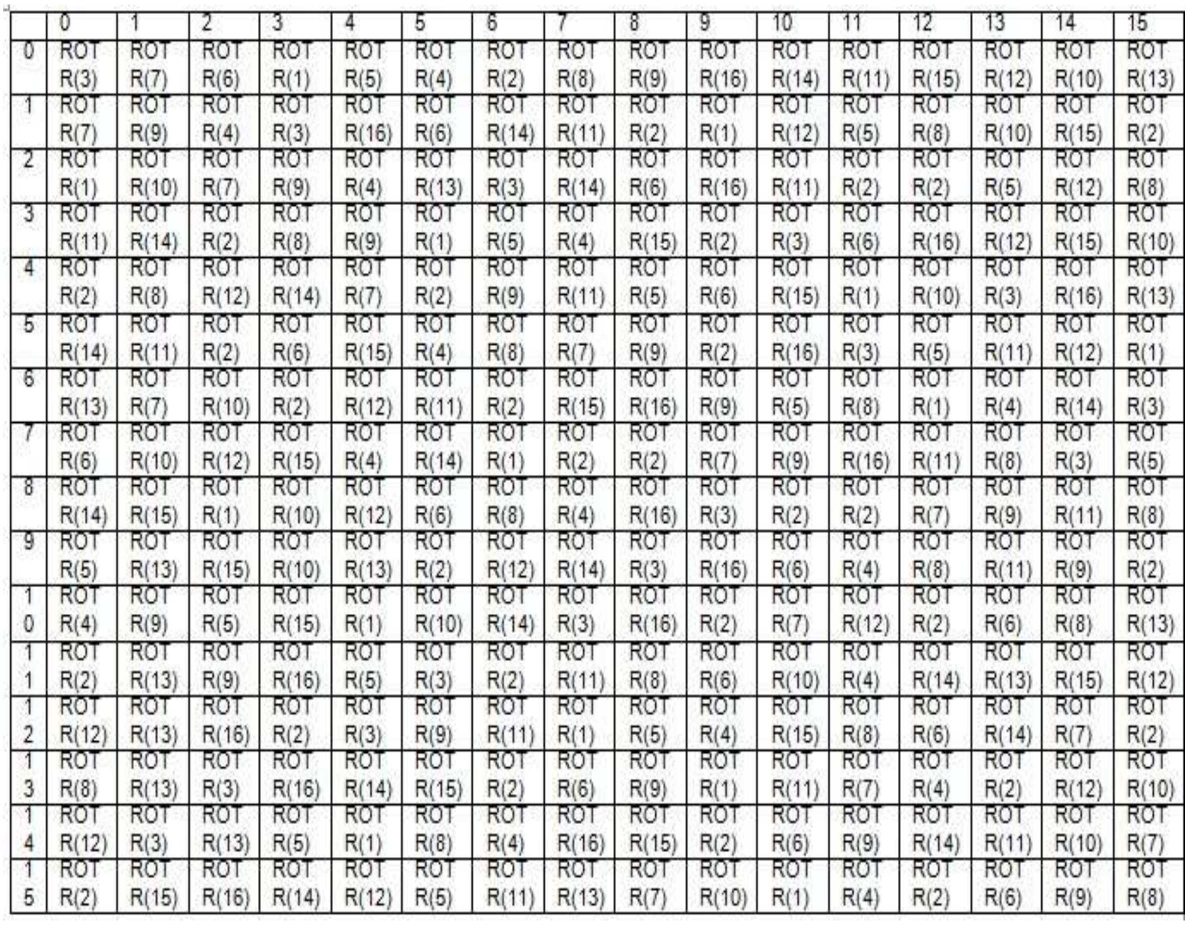
Decryption algorithm is the reverse of encryption algorithm. Steps involved are : Virtual split-2 , Resubstitution , Virtual split-2 , Reverse Rotation , Join Blocks , Conversion.

Binary sequence which is obtained is the virtually split as described before. This will find row and column to match the Resubstitution table. According to table the rotation operation is performed on the bit sequence. Again a virtual split is performed on the bit stream to find the row and column number for the substitution table. After finding the number of rows and columns it is matched to the Resubstitution table to resubstitute the original transaction which is finally obtained at the base station. This completes the decryption algorithm at the reception side.



V. Illustration

Encryption:
Consider an input transaction di as 155
10011011 (Conversion from decimal to binary)
Virtual split-1:
10011011 split as (1001 and 1011)
1001 in decimal is 9 which determines 9th row.
1011 in decimal is 11 which determines 11th column
Consider rotation table which is a private key as:



	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	ROT R(3)	ROT R(7)	ROT R(6)	ROT R(1)	ROT R(5)	ROT R(4)	ROT R(2)	ROT R(8)	ROT R(9)	ROT R(16)	ROT R(14)	ROT R(11)	ROT R(15)	ROT R(12)	ROT R(10)	ROT R(13)
1	ROT R(7)	ROT R(9)	ROT R(4)	ROT R(3)	ROT R(16)	ROT R(6)	ROT R(14)	ROT R(11)	ROT R(2)	ROT R(1)	ROT R(12)	ROT R(5)	ROT R(8)	ROT R(10)	ROT R(15)	ROT R(2)
2	ROT R(1)	ROT R(10)	ROT R(7)	ROT R(9)	ROT R(4)	ROT R(13)	ROT R(3)	ROT R(14)	ROT R(6)	ROT R(16)	ROT R(11)	ROT R(2)	ROT R(2)	ROT R(5)	ROT R(12)	ROT R(8)
3	ROT R(11)	ROT R(14)	ROT R(2)	ROT R(8)	ROT R(9)	ROT R(1)	ROT R(5)	ROT R(4)	ROT R(15)	ROT R(2)	ROT R(3)	ROT R(6)	ROT R(16)	ROT R(12)	ROT R(15)	ROT R(10)
4	ROT R(2)	ROT R(8)	ROT R(12)	ROT R(14)	ROT R(7)	ROT R(2)	ROT R(9)	ROT R(11)	ROT R(5)	ROT R(6)	ROT R(15)	ROT R(1)	ROT R(10)	ROT R(3)	ROT R(16)	ROT R(13)
5	ROT R(14)	ROT R(11)	ROT R(2)	ROT R(6)	ROT R(15)	ROT R(4)	ROT R(8)	ROT R(7)	ROT R(9)	ROT R(2)	ROT R(16)	ROT R(3)	ROT R(5)	ROT R(11)	ROT R(12)	ROT R(1)
6	ROT R(13)	ROT R(7)	ROT R(10)	ROT R(2)	ROT R(12)	ROT R(11)	ROT R(2)	ROT R(15)	ROT R(16)	ROT R(9)	ROT R(5)	ROT R(8)	ROT R(1)	ROT R(4)	ROT R(14)	ROT R(3)
7	ROT R(6)	ROT R(10)	ROT R(12)	ROT R(15)	ROT R(4)	ROT R(14)	ROT R(1)	ROT R(2)	ROT R(2)	ROT R(7)	ROT R(9)	ROT R(16)	ROT R(11)	ROT R(8)	ROT R(3)	ROT R(5)
8	ROT R(14)	ROT R(15)	ROT R(1)	ROT R(10)	ROT R(12)	ROT R(6)	ROT R(8)	ROT R(4)	ROT R(16)	ROT R(3)	ROT R(2)	ROT R(2)	ROT R(7)	ROT R(9)	ROT R(11)	ROT R(8)
9	ROT R(5)	ROT R(13)	ROT R(15)	ROT R(10)	ROT R(13)	ROT R(2)	ROT R(12)	ROT R(14)	ROT R(3)	ROT R(16)	ROT R(6)	ROT R(4)	ROT R(8)	ROT R(11)	ROT R(9)	ROT R(2)
10	ROT R(4)	ROT R(9)	ROT R(5)	ROT R(15)	ROT R(1)	ROT R(10)	ROT R(14)	ROT R(3)	ROT R(16)	ROT R(2)	ROT R(7)	ROT R(12)	ROT R(2)	ROT R(6)	ROT R(8)	ROT R(13)
11	ROT R(2)	ROT R(13)	ROT R(9)	ROT R(16)	ROT R(5)	ROT R(3)	ROT R(2)	ROT R(11)	ROT R(8)	ROT R(6)	ROT R(10)	ROT R(4)	ROT R(14)	ROT R(13)	ROT R(15)	ROT R(12)
12	ROT R(12)	ROT R(13)	ROT R(16)	ROT R(2)	ROT R(3)	ROT R(9)	ROT R(11)	ROT R(1)	ROT R(5)	ROT R(4)	ROT R(15)	ROT R(8)	ROT R(6)	ROT R(14)	ROT R(7)	ROT R(2)
13	ROT R(8)	ROT R(13)	ROT R(3)	ROT R(16)	ROT R(14)	ROT R(15)	ROT R(2)	ROT R(6)	ROT R(9)	ROT R(1)	ROT R(11)	ROT R(7)	ROT R(4)	ROT R(2)	ROT R(12)	ROT R(10)
14	ROT R(12)	ROT R(3)	ROT R(13)	ROT R(5)	ROT R(1)	ROT R(8)	ROT R(4)	ROT R(16)	ROT R(15)	ROT R(2)	ROT R(6)	ROT R(9)	ROT R(14)	ROT R(11)	ROT R(10)	ROT R(7)
15	ROT R(2)	ROT R(15)	ROT R(16)	ROT R(14)	ROT R(12)	ROT R(5)	ROT R(11)	ROT R(13)	ROT R(7)	ROT R(10)	ROT R(1)	ROT R(4)	ROT R(2)	ROT R(6)	ROT R(9)	ROT R(8)

Hence 9th row and 11th column is ROTR(4) i.e. Rotate Right 4 times
Hence operation performed on 10011011 is ROTR (4) which results as 10111001
Virtual split-2:
10111001 split as (1011 and 1001)
1011 in decimal as 11 determines 11th row
1001 in decimal as 9 determines 9th column
Consider the substitution table as:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	C1	B4	83	08	A8	50	D1	E6	F8	46	29	73	64	08	27	18
1	29	4C	2A	7F	9E	FE	AB	29	45	10	37	30	43	CC	BD	EA
2	18	94	75	03	40	F2	B7	AA	49	A8	C6	E3	05	07	34	BE
3	C5	B2	A7	34	19	AC	BD	34	87	30	22	59	55	45	23	68
4	BD	AF	A8	A5	12	BF	49	22	11	FB	BA	20	10	56	BB	12
5	CA	CC	10	57	65	47	A4	FF	FE	EF	D4	D7	A8	AD	27	4E
6	6A	4B	8C	27	23	97	A5	4D	9A	8A	4D	DA	C4	B4	FF	A4
7	45	68	62	57	59	19	35	76	40	00	14	03	50	48	72	04
8	A1	B4	5C	6D	9F	45	10	E4	A5	B8	A6	AF	FF	13	76	14
9	3A	7D	8F	AF	F4	A7	E4	EE	25	68	95	75	98	26	41	47
10	D3	A8	CC	CD	75	97	00	90	45	32	BA	E9	0E	15	28	1B
11	36	10	A2	A4	CA	DE	DF	BD	84	B2	D9	A7	3F	2F	2A	1E
12	12	14	19	24	10	2E	1F	1A	7A	5A	19	14	31	2B	8A	9A
13	57	A9	EE	4D	6F	A4	55	EA	A2	98	62	47	35	15	02	12
14	A2	BD	EA	FE	AB	BB	AE	AD	DC	CD	CB	CA	DE	CE	BA	AD
15	15	74	25	35	95	75	32	45	16	00	45	22	55	88	48	35

11th row and 9th column in the table is B2.
When we convert B2 in binary it becomes 10110010.
10111001 ⊕ 10110010 = 00001011

This 00001011 is sent as output. This will be output at the decryption's end. Along with this the position (Row number and column number) at which B2 is stored in decryption's table is also sent. Note that the position of B2 at encryption end and decryption end is not same i.e.B2 belongs to different position in both the tables and sender side knows the position of B2 at receiver side.

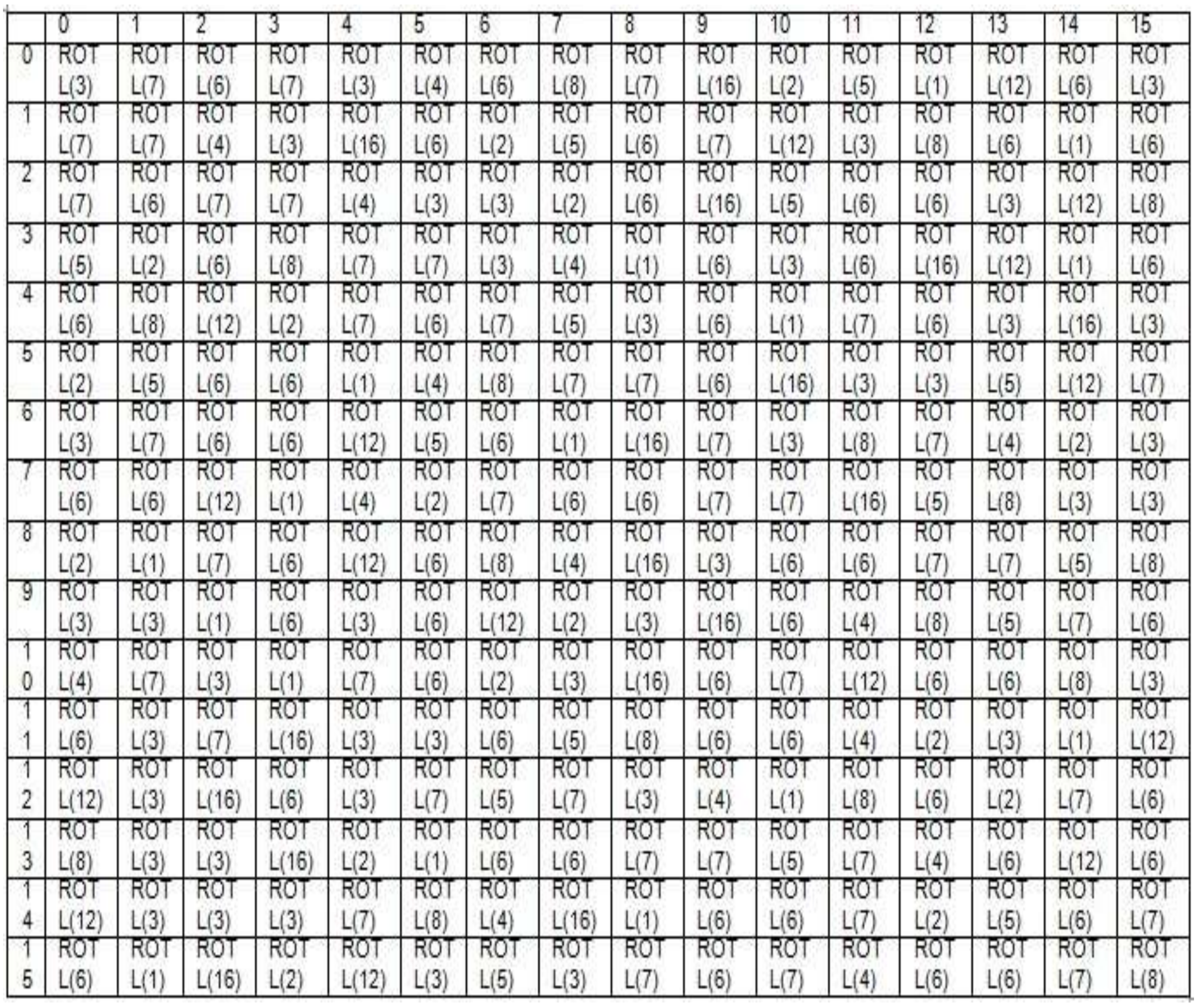
Decryption:

Input is 00001011 and row number and column number for decryption table.

Reconstitution is done by
00001011 ⊕ B2
00001011 ⊕ 10110010 = 10111001

Virtual split-2 is done to obtain the row number and column number as 11thcolumn and 9th row (this is transpose of virtual split-1)

Rotation is done using below table which is the complement of rotation table used in encryption method:



	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	ROT L(3)	ROT L(7)	ROT L(6)	ROT L(7)	ROT L(3)	ROT L(4)	ROT L(6)	ROT L(8)	ROT L(7)	ROT L(16)	ROT L(2)	ROT L(5)	ROT L(1)	ROT L(12)	ROT L(6)	ROT L(3)
1	ROT L(7)	ROT L(7)	ROT L(4)	ROT L(3)	ROT L(16)	ROT L(6)	ROT L(2)	ROT L(5)	ROT L(6)	ROT L(7)	ROT L(12)	ROT L(3)	ROT L(8)	ROT L(6)	ROT L(1)	ROT L(6)
2	ROT L(7)	ROT L(6)	ROT L(7)	ROT L(7)	ROT L(4)	ROT L(3)	ROT L(3)	ROT L(2)	ROT L(6)	ROT L(16)	ROT L(5)	ROT L(6)	ROT L(6)	ROT L(3)	ROT L(12)	ROT L(8)
3	ROT L(5)	ROT L(2)	ROT L(6)	ROT L(8)	ROT L(7)	ROT L(7)	ROT L(3)	ROT L(4)	ROT L(1)	ROT L(6)	ROT L(3)	ROT L(6)	ROT L(16)	ROT L(12)	ROT L(1)	ROT L(6)
4	ROT L(6)	ROT L(8)	ROT L(12)	ROT L(2)	ROT L(7)	ROT L(6)	ROT L(7)	ROT L(5)	ROT L(3)	ROT L(6)	ROT L(1)	ROT L(7)	ROT L(6)	ROT L(3)	ROT L(16)	ROT L(3)
5	ROT L(2)	ROT L(5)	ROT L(6)	ROT L(6)	ROT L(1)	ROT L(4)	ROT L(8)	ROT L(7)	ROT L(7)	ROT L(6)	ROT L(16)	ROT L(3)	ROT L(3)	ROT L(5)	ROT L(12)	ROT L(7)
6	ROT L(3)	ROT L(7)	ROT L(6)	ROT L(6)	ROT L(12)	ROT L(5)	ROT L(6)	ROT L(1)	ROT L(16)	ROT L(7)	ROT L(3)	ROT L(8)	ROT L(7)	ROT L(4)	ROT L(2)	ROT L(3)
7	ROT L(6)	ROT L(6)	ROT L(12)	ROT L(1)	ROT L(4)	ROT L(2)	ROT L(7)	ROT L(6)	ROT L(6)	ROT L(7)	ROT L(7)	ROT L(16)	ROT L(5)	ROT L(8)	ROT L(3)	ROT L(3)
8	ROT L(2)	ROT L(1)	ROT L(7)	ROT L(6)	ROT L(12)	ROT L(6)	ROT L(8)	ROT L(4)	ROT L(16)	ROT L(3)	ROT L(6)	ROT L(6)	ROT L(7)	ROT L(7)	ROT L(5)	ROT L(8)
9	ROT L(3)	ROT L(3)	ROT L(1)	ROT L(6)	ROT L(3)	ROT L(6)	ROT L(12)	ROT L(2)	ROT L(3)	ROT L(16)	ROT L(6)	ROT L(4)	ROT L(8)	ROT L(5)	ROT L(7)	ROT L(6)
10	ROT L(4)	ROT L(7)	ROT L(3)	ROT L(1)	ROT L(7)	ROT L(6)	ROT L(2)	ROT L(3)	ROT L(16)	ROT L(6)	ROT L(7)	ROT L(12)	ROT L(6)	ROT L(6)	ROT L(8)	ROT L(3)
11	ROT L(6)	ROT L(3)	ROT L(7)	ROT L(16)	ROT L(3)	ROT L(3)	ROT L(6)	ROT L(5)	ROT L(8)	ROT L(6)	ROT L(6)	ROT L(4)	ROT L(2)	ROT L(3)	ROT L(1)	ROT L(12)
12	ROT L(12)	ROT L(3)	ROT L(16)	ROT L(6)	ROT L(3)	ROT L(7)	ROT L(5)	ROT L(7)	ROT L(3)	ROT L(4)	ROT L(1)	ROT L(8)	ROT L(6)	ROT L(2)	ROT L(7)	ROT L(6)
13	ROT L(8)	ROT L(3)	ROT L(3)	ROT L(16)	ROT L(2)	ROT L(1)	ROT L(6)	ROT L(6)	ROT L(7)	ROT L(7)	ROT L(5)	ROT L(7)	ROT L(4)	ROT L(6)	ROT L(12)	ROT L(6)
14	ROT L(12)	ROT L(3)	ROT L(3)	ROT L(3)	ROT L(7)	ROT L(8)	ROT L(4)	ROT L(16)	ROT L(1)	ROT L(6)	ROT L(6)	ROT L(7)	ROT L(2)	ROT L(5)	ROT L(6)	ROT L(7)
15	ROT L(6)	ROT L(1)	ROT L(16)	ROT L(2)	ROT L(12)	ROT L(3)	ROT L(5)	ROT L(3)	ROT L(7)	ROT L(6)	ROT L(7)	ROT L(4)	ROT L(6)	ROT L(6)	ROT L(7)	ROT L(8)

This operation is ROTL (4) which will give result as 10011011 which is the required output.

VI. Conclusion

In this paper, we proposed the Encryption and Decryption algorithm to be secured data transmission between the base station BS and the mobile host MH(s) and to ensure our database at the consistency level once the data are secure. The aim of this paper to made balance between minimum time requesting for any transaction management to avoid the abort transactions and maximum time need to avoid any hacker decryption. We have adapted algorithm in both side (the BS and MH(s) sides) to save the transaction during the transmission between the BS and MH(s).

As compared to the previously known security schemes[1][2][3][6][7] this scheme takes lesser time for calculations and provides good level of security. The security used in this paper is free from number of loops and heavy calculations which usually consumes a lot of power of mobile devices[3]. The encryption decryption method is secure from attacks in the wireless medium and can be easily implemented on database systems with crucial transactions. It does not effect the normal procedure of transaction at all and can be used easily on database and transaction systems.

References

[1] Ziyad Tariq Abdul-Mehdi FIST-MMU Ali Bin Mamat& Hamidah Ibrahim
FSKTM-UPM Mustafa. M.Dirs FITM," MULTI-CHECK-OUT TIMESTAMP ORDER TECHNIQUE (MCTO) FOR PLANNED DISCONNECTIONS IN MOBILE DATABASE",2006.

[2] Margaret H. Dunham a,_, Abdelsalam Helal b,__ and Santosh Balakrishnan c a Department of Computer Science and Engineering Southern Methodist University, Dallas, "A mobile transaction model that captures both the data and movement behaviour",1997

[3] Ziyad.T.Abdul-Mehdi , Ramlan Mahmod, "Security Management Model for Mobile Databases Transaction Management",2008

[4] [1] Forouzan. B. 2007. "Data Communications and Networking", Fourth edition, Mc Graw Hill, Singapore, ISBN 007- 125442-0.

[5] Lubinski. A. 1999. "Adaptation Concepts for Mobile Database Security" University of Rostock,
Rostock, Germany.

[6] Abdul-Mehdi, Z.T. Mamat, A.B. Ibrahim, H. Dirs, Mustafa.M. 2006."Multi-Check-Out Timestamp Order Technique (MCTO) for Planned Disconnections in Mobile Database", The *2nd* IEEE International Conference on Information & Communication Technologies*: from Theory to Applications* , 24-28 April, Damascus, Syria,Vol.1, p.p 491-498.

[7] Abdul-Mehdi.Z.T, Mamat.A, Ibrahim.H and Deris.M. 2006. "Transaction Management ModelFor Mobile Databases". PhD Thesis in Computer Science, Faculty of Computer Science and Information Technology, University Putra Malaysia, P.P.3.