PERFORMANCE ANALYSIS OF MULTIPLE KEY IN MULTINODE NETWORK

Ajay Kakkar*

ABSTRACT

Data security involves encryption and key management skills for a given packet transmission in a network. As the number of users and nodes are increased in the network; the security level has tended to fall. In the recent trends; there is drastically increase in the number of user, therefore, the need of data security mechanisms seek continues attention. Cryptography is the best technique used to avoid unauthorized access of data. It involves encryption algorithm and the keys which are being used by the users. Key management is very essential part of cryptography. From the experimental results it has been observed that single key is not able to provide sufficient security to the data. The use of multiple keys has the power to make the cryptographic model secured from the hacker. Multiple key also consumes more power and may cause congestion; therefore, it is not worthy to support all the nodes with multiple keys. Only such nodes having less security level needs to be supported with multiple keys. The performance analysis of multiple keys in a network needs to done in order to evaluate the security level of all the nodes. This paper is based upon the utility of multiple keys in network in order to improve the security level and reduce congestion.

Keywords: *Encryption, Keys, Nodes, S-Boxes.*

*Assistant Professor, Thapar University, Patiala.

1. INTRODUCTION

It is a technique used to avoid unauthorized access of data. The encryption process consists of single or multiple keys to hide the data from the hackers. The original text before the encryption process is known as Plaintext. The text obtain after encoding the data with the help of a key is known as cipher text. In the encryption process single or multiple keys can be used for the encryption of data. Key needs to be transported over a secured channel from transmitter to receiver via various nodes. In both the cases fixed as well as variable lengths for the keys can be considered. The single key having fixed length (8-16bits) cannot offer much resistance to the hacker. In order to increase the security level the key length has to be increased which creates more overheads. The second approach is to use multiple keys of variable length which provide a good security level with moderate overheads. The main aim of data security is to make the information available and to protect the same for confidentially and integrity.

2. LITERATURE SURVEY

The work done carried out by the researchers so far in the field of data security, multiple keys and node failure in a network has been critically analyzed. The various issues related with key management in multinode network have also been considered in the literature survey.

In 1980 R.L. Rivest et. Al. proposed a method for obtaining digital signatures and public key cryptosystems. They assumed that the deciphering function for the received message in order to obtain the original message was known to receiver. If key size was increased then hacker required more number of attacks in order to get an access of the data, hence it took more time to recover the data. The detail of time required to decrypt the data from a pool with random keys is shown in table1

Digits	Number of operations	Time
50	1.4 * 1010	3.9 hours
75	9.0 * 1012	104 days
100	2.3 * 1015	74 years
200	1.2 * 1023	3.8* 1009 years
300	1.5 * 1029	4.9* 1015 years
500	1.3 * 1039	4.2* 1025 years

Table 1: key size, number of operations and time required by the hacker to break the system

In 1997, Min Shiang Hwang, presented a cryptographic key assignment scheme for the access control in multiuser system. Users were not able to create and exchange the keys randomly which was the main limitation of the work. Shiang proposed a modified algorithm which enabled the use of partially ordered structure. It also permits the users to generate and exchange their own keys randomly in MN. They failed to determine the key failure rates and did not provide any recovery mechanism for the faulty keys. In 1988, R.S. Sandhu implemented a cryptographic tree hierarchy approach for the access control in different security levels. Each user has the power to store a single key of fixed size corresponding to the level of security. The two main limitations of the work were deriving new keys from existing key and the use of fixed length keys. In 2008, Huawei Huang, Bo Yang, Shenglin Zhu, and Guozhen Xiao, presented a generalized public key cryptosystem based upon a new Diffie Hellman problem. The scheme provides almost double the message expansion by using secret key length without any additional computational cost. The scheme was based upon one way key exchange protocols which degraded the security level of the model. In 2011 Jason Crampton proposed a time storage trade off for cryptographically enforced access control. The encryption time and delay caused by various nodes were determined by using graph theory. For timely encryption the security level was comprised with processing time. It was assumed that each user had a single secret value used to derive the key. The work includes the study and analysis of the trade off between the space required for the information and the time taken to derive the keys. The practical implementation of keys for encryption and decryption in cryptographic enforcement algorithm of graph based authorization policies were not covered in the work. From the literature survey the following observations have been drawn:

- Single key with fixed length cannot be used to provide secure communication in MN. By knowing the data and key length the hacker is able to generate side channel and middle line attacks.
- Multiple keys having different failure rates can be achieved by varying the key length. They are always preferred for encrypting the data in MN having large number of nodes.

3. ANALYSIS AND SIMULATION RESULTS FOR PROCESSING AND HACKING TIME

In a network having eight S-Boxes used for encryption of the data with multiple keys having same length are considered in this section. The results are verified on DSP Kit (TMS 320

ADP6713). The strength of model in a network depends upon the key selection and replacement of faulty keys with new key. Key shifting time is very important; there are three main process in the key shifting time operation; i) Key generation time, ii) key testing time and iii) key processing time. The analysis and simulation results for the failure rate of key having fixed key length has been determined with key shifting time (δ)=0.01ns.

Case –I : Node = 5, S-Boxes= 8, 1st Key Length=8 (Bits), 2nd key length 8 (Bits)

S. No.	Data Length (bits)	Processing time (ns)	Hacking Time (min)
1.	16	13.36	71.29
2.	32	7.99	59.65
3.	64	11.44	65.88
4.	128	14.90	72.12
5.	512	18.36	78.35

Table 2: Processing and Hacking time for 16, 32, 64, 128 and 512 data length by using multiple keys having 8 bit key length for 1st and 2nd respectively designed by 8 S-Boxes

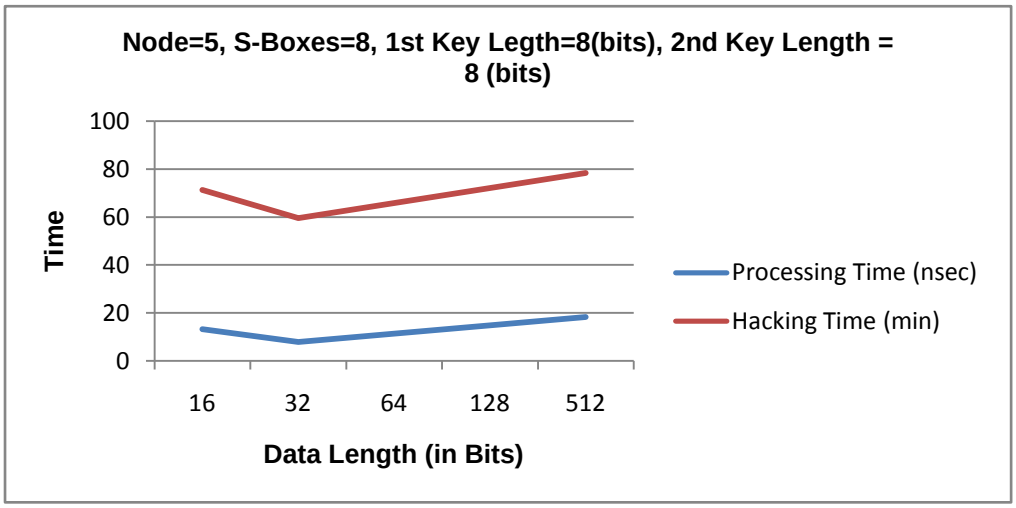

Figure 1: Processing and Hacking time Vs Data length for Node=5, S-Boxes=8 and Key length=8 bits

Followings observations has been drawn from table 2

- For 8 S-Boxes and 8 bit key length, the system takes more processing time (13.36 ns) which provides more time to the hacker. Initially, the system requires some additional time in order to set up the algorithm and run boot files. Further the encryption time for the same or different data streams is less because it does not include any boot time.
- If data length is increased from 16 to 32 bits for same the key length (8 bits), in figure 5.3, the dip shows that the hacking time reduces significantly from 71.29 to 59.65 minutes. When multiple keys k_1 & k_2 having same failure rates (for both the keys

having 8 bit key length) have been used to encrypt the data then the actual performance of the system has been observed. It does not include any booting time, and the time taken by the system is only concerned with encryption process. If data length is further increased from 32 to 64 bits for the same key length then the hacking time is increased from 59.65 to 65.88 minutes with nominal increase in processing time 7.99 to 11.44 ns. It shows that in case of long data sequences when encrypted by multiple keys having same key length then secured model is achieved. So, there is need to change the parameters in order to reduce the hacking time. There are two ways: a) increase the key length, b) use more number of S- Boxes.

- The single key having length 8,16,32,64 and 512 bits is not preferred due to its poor response in the encryption process. A very low security level and high processing time has been observed when single key has been used for the encryption of 8 to 512 bit data sequences.

Case –II: For Nodes=5, S-Boxes= 8, 1st key length= 8 bits, 2nd key length= 16 bits

S. No.	Data Length (bits)	Processing time (ns)	Hacking Time (min)
1.	16	12.95	72.13
2.	32	10.15	59.96
3.	64	17.90	65.90
4.	128	19.98	72.98
5.	512	22.01	79.15

Table 3: Processing and Hacking time for 16, 32, 64, 128 and 512 data length by using multiple keys having 8 and 16 bit key length for 1st and 2nd resp. designed by 8 S-Boxes

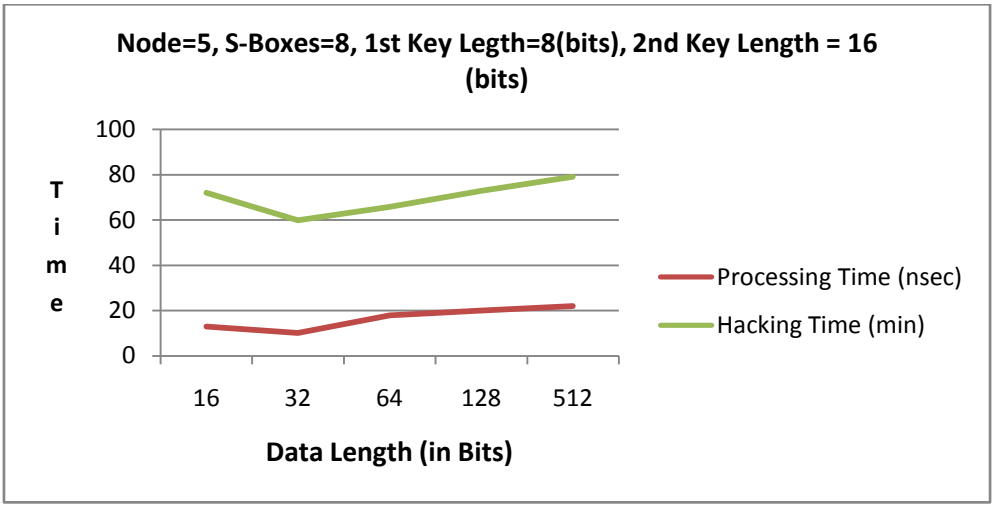

Figure 2: Processing and Hacking time Vs Data length for Node=5, S-Boxes=8 with two keys having 8 and 16 bits respectively

For the encryption of 16, 32, 64, 128 and 512 data sequence with two keys (1st key length= 8 bits and 2nd key length= 16 bits) designed by eight S-Boxes for MN having 5 nodes; the hacking time decreases very slightly and there is a small increase in the processing time. The initial response of the model remains same due to the use of same 1st key length, the 2nd key plays its part whenever the failure rate of 1st key increases abruptly, therefore, the processing time also raises, which is very minute. 16 iterations (round functions) have been done by the S-Boxes for the generation of 2nd key which also results in more processing time.

Case –III: For Nodes=5, S-Boxes= 8, 1st key length= 16 bits, 2nd key length= 8 bits

S. No.	Data Length (bits)	Processing time (ns)	Hacking Time (min)
1.	16	13.91	71.35
2.	32	12.09	50.16
3.	64	19.81	63.13
4.	128	23.95	68.71
5.	512	28.05	76.91

Table 4: Calculations of Processing and Hacking time for 16, 32, 64, 128 and 512 data length by using multiple key having 16 and 8 bit key length for 1st and 2nd respectively

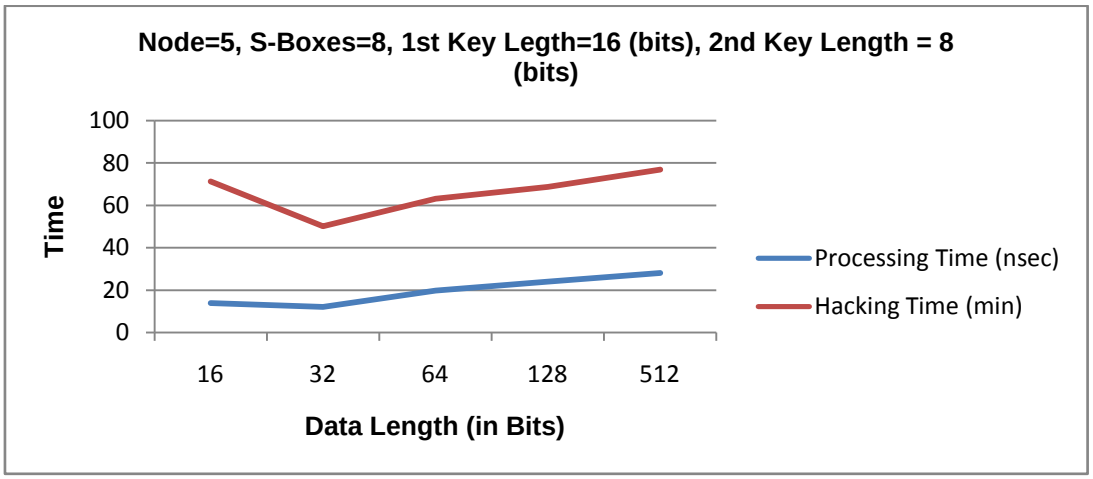

Figure 3: Processing and Hacking time Vs Data length for Node=5, S-Boxes=8 and with two keys having 16 and 8 bits respectively

For the encryption of 16, 32, 64, 128 and 512 data sequence with two keys (1st key length= 16 bits and 2nd key length= 8 bits) designed by eight S-Boxes for MN having 5 nodes; there is significant decrease in the hacking time with slight increase in the processing time. e.g. for 32 bit data, the hacking time is decreased from 59.96 to 50.16 minutes with 1.94 ns increase in processing time which is very nominal and can be ignored. The initial response of the model is very good due to the use of long 1st key length; the 2nd key plays its part whenever

the 1st key fails. The 16 iterations (round functions) have been used for the generation of 1st key by using eight S-Boxes. Each S-Box performs dual iterations for the generation of 1st key and it takes more processing time for each data sequence.

Case –IV: For Nodes=5, S-Boxes= 8, 1st key length= 16 bits, 2nd key length= 16 bits

S. No.	Data Length (bits)	Processing time (ns)	Hacking Time (min)
1.	16	44.88	81.71
2.	32	33.07	14.41
3.	64	36.52	20.65
4.	128	39.99	26.88
5.	512	43.44	33.11

Table 5: Processing and Hacking time for 16, 32, 64, 128 and 512 data length by using multiple keys having 16 bit key length for both 1st and 2nd respectively designed by 8 S-Boxes

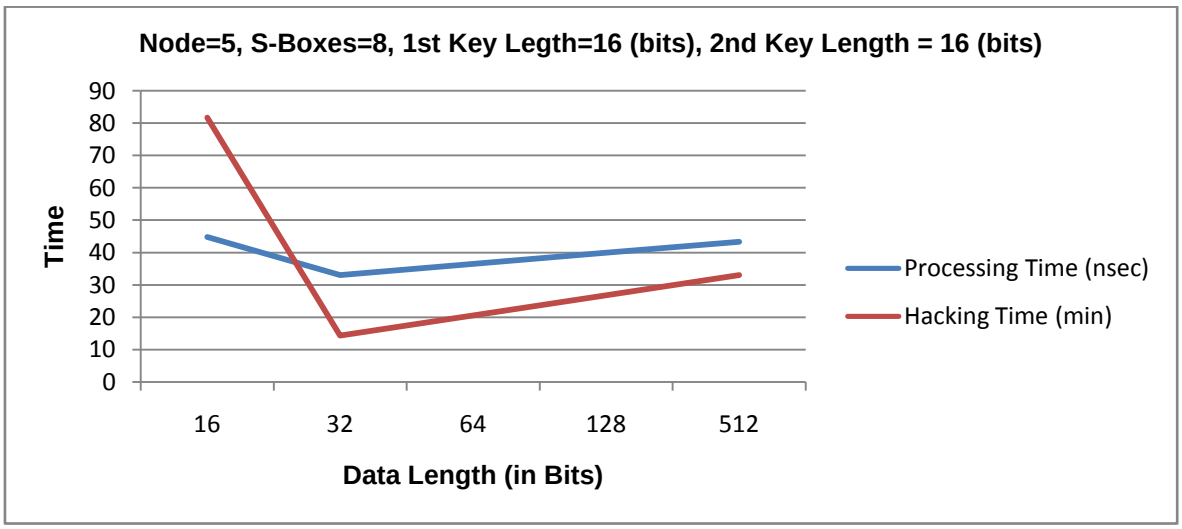

Figure 4: Processing and Hacking time Vs Data length for Node=5, S-Boxes=8 and having same Key length=16 bits

From the table 5 it has been observed that for 8 S-Boxes and 16 bit key length, initially, the system takes more processing time (44.88 ns) which provides more time to the hacker. If data length is increased from 16 to 32 bits for same key length (16bits) then it shows that the hacking time reduces significantly (81.71min to 14.41 min.). If data length is further increased 32 to 64 bits for the same key length then the hacking time is increased from 14.41 to 20.65 minutes with nominal increase in processing time 33.07 to 36.52 ns.

CONCLUSIONS AND FUTURE SCOPE

The performance of the proposed model in regard of processing time and hacking time and multiple key failure rates at various nodes has been evaluated and found much better than the existing cryptographic models. For 5 Nodes, if eight S-Boxes have been used to generate two keys of 16 bit length for the encryption of 512 bit data length sequence then the hacking time has reduced by a factor 2.38. It shows that system is 42% more secure as compared to previous case. At the same time the overheads (processing time) increases, but this will not affect the model much. Depending on the particular attacking scenarios, various combinations may be employed associated with integrity and confidentiality of the data. The hacking time can be further reduced if more number of S-Boxes can be explored for the same task.

REFERENCES

1) B. Schneier. (1996). Applied Cryptography. Second edition. John Wiley & Sons.
2) C. H. Kim, (2012). Improved Differential Fault Analysis on AES Key Schedule. IEEE Transactions on Information Forensics and Security.7.1: pp.41-50.
3) H. Chien, and J. Jan, (2003) New hierarchical assignment without public key cryptography. Computers & Security. 22.6: pp.523–526.
4) Huawei. Huang, Bo. Yang, Shenglin. Zhu, and Guozhen. Xiao, (2008).Generalized ElGamal Public Key Cryptosystem Based on a New Diffie-Hellman Problem. Proceedings of the 2nd International Conference on Provable Security. pp.1-21.
5) Jason. Crampton, Time-Storage Trade-Offs for Cryptographically-Enforced Access Control. (2011). Lecture Notes in Computer Science, Springer. 6879: pp.245-261.
6) M. Hwang, and W. Yang, (1997). Multilevel secure database encryption with subkeys. International Journal of Data & Knowledge Engineering archive. 22.2: pp.117-131
7) R. L. Rivest, A. Shamir, and L. Adleman, (1980). A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. securespeech.cs.cmu.edu/reports/RSA.pdf.
8) R. S. Sandhu, Cryptographic Implementation of a Tree Hierarchy for Access Control. (1988). Information Processing Letters. 27. pp.95-98.
9) V.R.L. Shen, and T. S. Chen, (2002). A novel key management scheme based on discrete logarithms and polynomial interpolations. Computers & Security. 21.2: pp.164–171.