ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 4, April 2018

Performance Analysis of DOSN Based on Privacy Preserving and Information Sharing

Nikita V. Ghodpage[1], Prof. R. V. Mante[2], Dr. K. P. Wagh[3]

PG Scholar, Department of Computer Science and Engineering, Government College of Engineering, Amravati, Maharashtra, India[1]

Assistant Professor, Department of Computer Science and Engineering, Government College of Engineering, Amravati, Maharashtra, India[2]

Assistant Professor, Department of Information Technology, Government College of Engineering, Amravati, Maharashtra, India [3]

ABSTRACT: Nowadays, people are aware of Online Social Networks. Several online social networks are benefitted by regular people for their piece of work. Currently, almost online social networks are based on the centralized systems. As these centralized systems gathers huge information at the single provider, that is risk for the privacy of user's personal data. To knock out the problem of centralized systems, new concept is brought to existence i.e., Decentralized Online Social Network. As a substitute of centralized systems, Decentralized Online Social Network distributes its data among numerous servers or within multiple users. This paper concentrates on the challenges of privacy and secrecy for the information of users that is communicated on the group. Fresh solutions have to be discovered to lessen existing difficulty. First solution can be the reduction of re-encryption process of documents before sharing it, when the group's size keeps varying. Second solution to the existing difficulty can be to moderate the dependency of keys for the fear of any unknowing sudden changes in the group. Present solutions lack in achieving the above standards. Our methodology insists single time encryption for each file and there is no necessity to revise keys after user's addition or removal from the group.

KEYWORDS: Decentralized Online Social Networks(DOSNs); Group Communication; Information Sharing; Privacy Preserving

I. INTRODUCTION

At present days, people adores the use of Online Social Network. Online Social Networks are centralized systems usually. Specifically, centralized system collects all the user's data at the solitary provider. Due to this functionality of the centralized systems, there is possibility that user's data is in danger. To cut off the problem of centralized systems, Decentralized Online Social Network (DOSN) has taken into account. By making use of Decentralized Online Social Network (DOSN) various issues such as user's confidentiality, operating expenditure and scalability can be disciplined as competed to the centralized systems. Therefore, DOSN can be a recent trend on which user's data is moved on. For providing OSN services, some authenticated and trusted providers are needed such that user's data can be blindly handed over to those faithful providers.

While confidentiality of user's content must be promised, when user reveals their data on the group with regarding to other users in the group. Data Management is the recent argument in the decentralization. Most of the admired OSNs permit their users to arrange connections with different users in several groups. All the users accessing OSNs are allowed to create new groups according to their interest. After creating the groups other users who shares related interest can join that group by sending request for joining that group. If group creator accepts the group joining request, then only user will be able to access the documents or data that has been shared by the other group members. As providing privacy of the content shared by the group members is the main subject of this paper.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 4, April 2018

Generally, in groups, online social networking allows group creator to add or remove other users in that group any time. As group creator will always be aware of trickster is fictional. So groups on online social networking must be shielded. Each and every time after file or data is being uploaded by user on group must be safe such that confidential data of users doesn't gets disclose to unauthorized users. Encryption is one of the solution to guarantee the privacy of user's confidential data and since long time ago the concept of encryption is being used. Currently, every user is being tested whether those are authenticated and that data belongs to them only or not, along with they have permission of access those documents that are provided on group.

Above mentioned solutions falls short due to increasing number of encryption operations to boost security in the case when any group member is added to or removed from the group. Thus, we propose a system which will lessen the computational overhead of current encryption method and eliminate the need of changing encryption keys all the time whenever the user is added to or removed from the group. The foremost idea of our proposed system is to grant security on the user's data and files that has been circulated on the group of online social network using our devised encryption method.

The organization of this paper is as follows: Section II provides Literature Survey. Section III provides newly devised algorithms that is used for implementation of system. Section IV provides pseudo code for Group Key Generation. Section V provides Experimental Evaluation. Section VI concludes the paper.

II. LITERATURE SURVEY

Logical Key Hierarchy (LKH) model is the existing system that maintains hierarchical structure of keys for every available group. Users are asked for the group key, when they wish to access or download the documents those are distributed on group. In LKH, public group key is shared between all the members of the group [1]. LKH has only pitfall that it changes encryption keys of the document that has been circulated in the group every time when any group member is added to or removed from the group. This pitfall of LKH triggers the computational overhead. To conquer the problems of LKH we have proposed the new system which take control of that problem.

In [2], Twitter is one of the famous Online Social Networking service for distribution of short messages i.e., tweets amongst other users of Twitter. Its users often use URL curtailing services that provides compressed alias of long URL for sharing via tweets and public analytics of concise URLs. To maintain the privacy of specific users public click analytics has been provided in an aggregated form. Inference attacks to infer which compresses URLs clicked on by a target user has been proposed. Only public information is required in their attacks. To evaluate their attacks, they have crawled and monitored there clicked analytics of URL shortening services and Twitter data. Throughout their experiments, they have shown their attacks can infer the candidates in the most cases.

In [4], three solutions for preserving privacy to the user's profile matching with homomorphic encryption technique and multiple servers has been provided. Their solutions permit some matching users can be searched with the use of the multiple servers exclusive of disclosing to anyone privacy of the query and the queried user profiles. Analyses of security have been exhibited that their privacy-enhanced protocol and the optimized two-party protocol achieves user profile privacy and user query privacy. They have presented that their protocols specifically optimized two-party protocol are practical and feasible.

In [12], Tree data structure has been used to model the contents where nodes contain the encrypted data by a symmetric resource key, while different symmetric access key is used to encrypt edges list. There is mapping between access and the resource keys and those keys are shared with the different members of the group using a shared symmetric key exchanged during friendship request. Shared access key is initialized for the empty group. After joining group by the user, the corresponding access key and the mapping between access key and the resource keys of the contents are securely dispensed to the new member. When user is provided access permission, the access key is changed and the disturbed documents published in the profile hierarchy are re-encrypted with the new access key. Decisively, the new access key is circulated to the left members of the group. The eliminated members will not be able to access the disturbed documents, nevertheless they have access to the corresponding resources since they persist encrypted with the same resource keys.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 4, April 2018

III. DEVISED ALGORITHMS

Data uploaded on the group of Online Social Network is encrypted before loading it onto the server and before downloading decryption is performed over the data. Using following encryption and decryption algorithms we secure the user's private data:

A. Encryption:

1. Read Document into Byte Array b[ ].
2. Divide b[ ] into two Arrays.
3. Set b1 = [b.length/2]
4. Set b2 = [b.length - (b.length/2)]
5. Swap Byte Arrays b1[ ] with b2[ ] and b2[ ] with b1[ ]:
   i. Set temp[ ] = b1[ ]
   ii. Set b1[ ] = b2[ ]
   iii. Set b2[ ] = temp[ ]
6. Encrypt Byte Array b2[ ] with b1[ ]:
   $Enc_{B2[\,]} = b2[\,] \wedge b1[\,]$
7. Encrypt Byte Array b1[ ] with Key:
   $Enc_{B1[\,]} = b1[\,] \wedge Key$
8. Initialize Encrypted Byte Array EncByte[ ].
9. Concatenate Encrypted Byte Arrays B1[ ] and B2[ ]:
   $EncByte[\,] = Enc_{B1[\,]} + Enc_{B2[\,]}$
10. Write EncByte[ ] into File.

B. Decryption:

1. Read Encrypted File into Byte Array benc[ ].
2. Divide benc[ ] into two Arrays.
3. Set benc1 = [benc.len/2]
4. Set benc2 = [benc.len - benc.len/2]
5. Decrypt Byte Array benc1 using Key k:
   $b1[\,] = benc1[\,] \wedge k$
6. Decrypt Byte Array benc2 using b1[ ]:
   $b2[\,] = benc2[\,] \wedge b1[\,]$
7. Swap Byte Arrays b1[ ] with b2[ ] and b2[ ] with b1[ ]:
   i. Set temp[ ] = b1[ ]
   ii. Set b1[ ] = b2[ ]
   iii. Set b2[ ] = temp[ ]
8. Initialize Decrypted Byte Array Out[ ].
9. Concatenate Decrypted Byte Arrays b1[ ] and b2[ ]:
   $Out[\,] = b1[\,] + b2[\,]$
10. Write Out[ ] into File on Server.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 4, April 2018

TABLE I.
DESCRIPTION OF TERMSFROM ABOVE ALGORITHMS

Terms	Description
b[]	Byte Array
b.length	Length of Byte Array
temp[]	Temporary Byte Array
benc[]	Encrypted Byte Array
Out[]	Decrypted Byte Array
benc.len	Length of Encrypted Byte Array

In the below Fig. 1, When user uploads the file, that file will be stored on the server in encrypted format. We have used our own devised encryption algorithm for encrypting those files. When user uploads the file initially, document is read into byte array. Then, that byte array is divided into two byte arrays. First byte array will be exactly first half of the original byte array. And Second byte array will be next other half of the original byte array i.e., remaining byte array of the original byte array. We can't assume second byte array size i.e., whether it is even sized byte array or odd sized byte array. After getting two byte arrays first byte array is swapped with the second byte array. Further, second byte array is XORed with first byte array. Then, first byte array is XORed with the key. Subsequently, new byte array is initialized to store the encrypted byte arrays that we get after performing XOR operation. We get two encrypted byte arrays. The byte array that is XORed with key results encrypted first byte array. So, first encrypted byte array is concatenated with the second encrypted byte array. And the resulted byte array after concatenation is stored into the new byte array that was initialized earlier. And at last resultant byte array is written into the file on server.

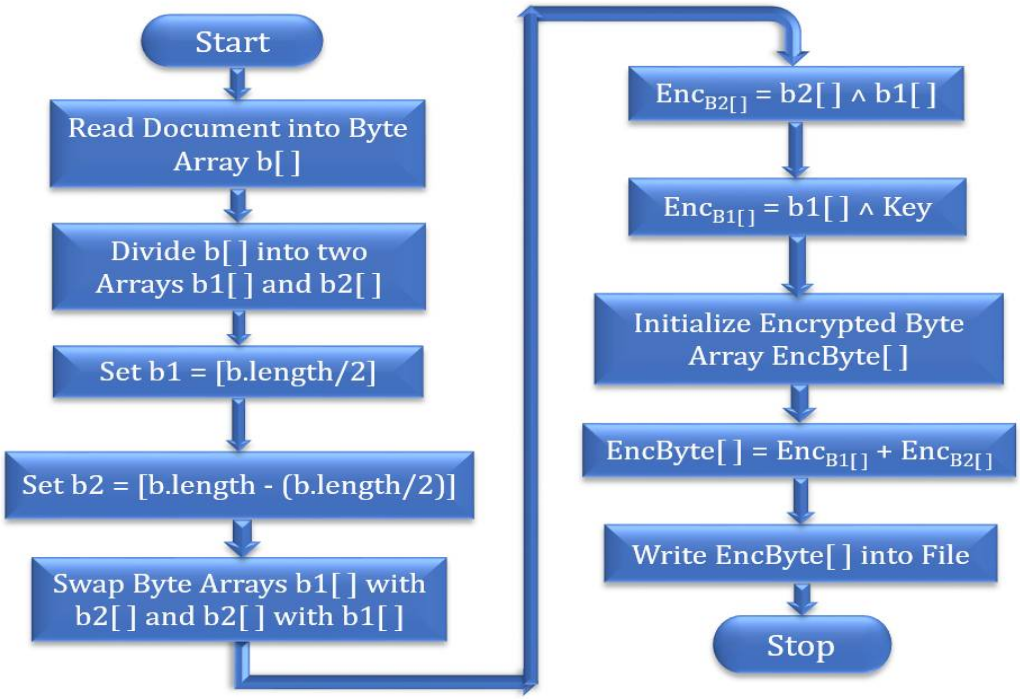

Fig. 1 Flow of Encryption Algorithm

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 4, April 2018

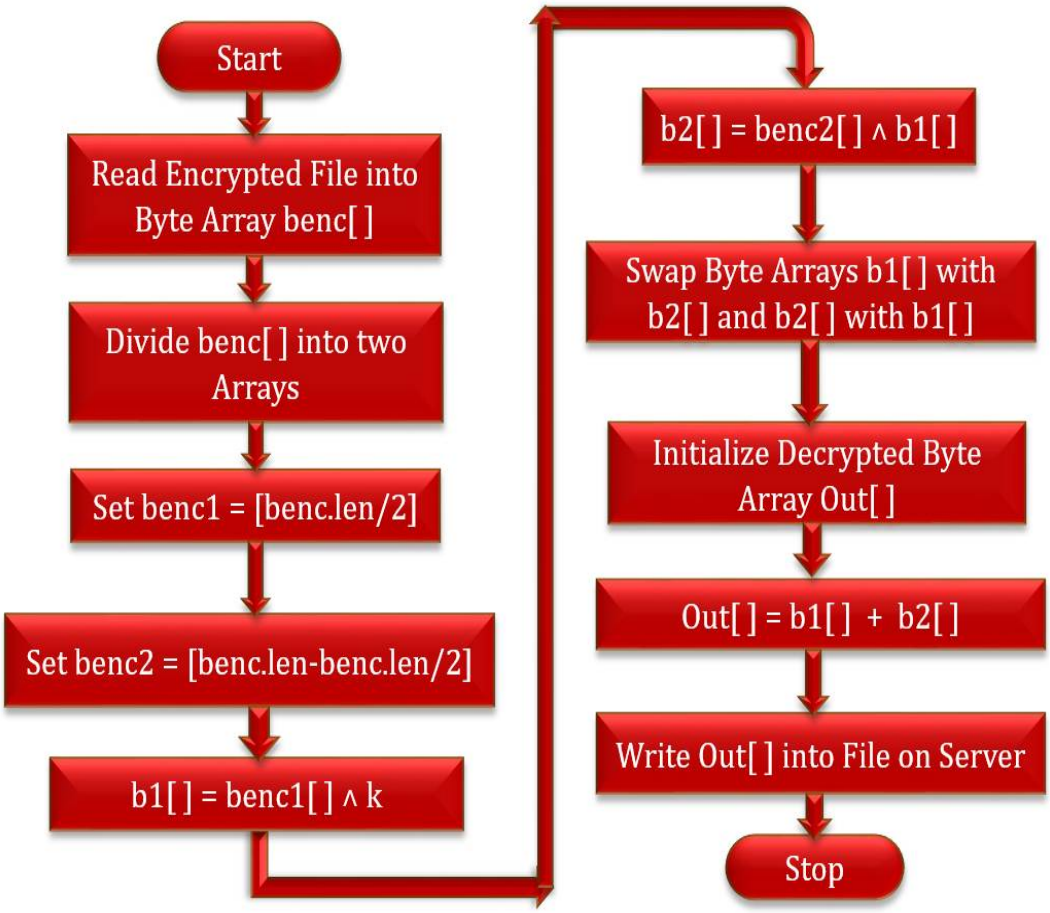

Fig. 2 Flow of Deryption Algorithm

In the above Fig. 2, Initially, user uploads file on server, before storing document on server file, encryption is performed on the file. In decryption process, encrypted file is fetched from the server, we have used our own devised decryption algorithm for decrypting those files encrypted files. Firstly, Encrypted file is read into the byte array. Then, that byte array is divided into two byte arrays. First byte array will be exactly first half of the original byte array. And Second byte array will be next other half of the original byte array i.e., remaining byte array of the original byte array. We can't assume second byte array size i.e., whether it is even sized byte array or odd sized byte array. After getting two byte arrays first byte array is XORed with key to get first decrypted byte array and the second byte array is XORed with the first decrypted byte array to get second decrypted byte array. Further, decrypted byte arrays will be swapped to one another i.e., first decrypted byte array is swapped with second decrypted byte array. Then, new decrypted byte array is initialized. Afterwards, first decrypted byte array is concatenated with the second encrypted byte array. And the resulted byte array after concatenation is stored into the new byte array that was initialized earlier. And at last resultant byte array is written into the file on server.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 4, April 2018

IV. PSEUDO CODE

Following is the pseudo code for the Group Key Generation that has been used while implementation of the system:

1. Set GroupName = Name of the Group.
2. Set GroupID = ID of the group i.e., Private Key (PK)
3. Set MemID = PK of group member i.e., unique group ID
4. Set MemUID = PK of user details i.e., unique integer userID
5. Set gnm = Reverse (GroupName)
   If gnm.len > 3
   Set gnm1 = substring (gnm,3)
   Else
   Set gnm1 = gnm
6. Set UID = Reverse (MemUID)
7. Set GroupKey = gnm1 + "@" + UID + "$" + MemID + "*" + GroupID

V. EXPERIMENTAL EVALUATION

Following Fig. 3 shows the Graph for Time that is required to encrypt the document as well as decrypt same document. Our system takes less time to encrypt and decrypt documents competed to the existing LKH model. As in LKH model document's keys used for encryption were dependent on group keys due to hierarchy maintenance. Our system has independent keys due to which number of operations are reduced while encryption and decryption performance.

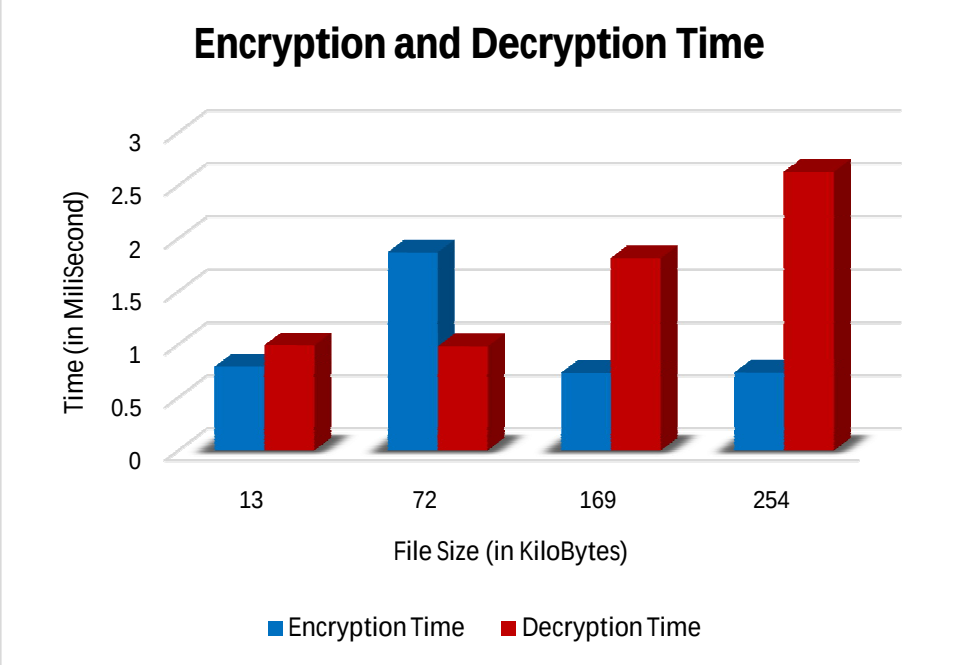

Fig. 3 Graph for Encryption and Decryption Time

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 4, April 2018

VI. CONCLUSIONS

In our paper, we have proposed Decentralized Online Social Networks that consists of two Online Social Networking sites on which users can form groups and communicate to other users of different groups, account holder on site 1 will also be allowed to access documents from the groups of site 2 if and only if access permission is granted for that document and vice versa.Hence, access permissions can be assigned to reduce the strain regarding secrecy of the user's data.Only authenticated users can either upload documents or download documents those are distributed on the group. Communication held between groups of Online Social Network will be secured using Encryption and Decryption algorithms that has been developed. As in existing system number of operations to provide security to the documents is plentiful. Hence, to knock out the enigma of existing system our system has been proposed.

REFERENCES

[1] Andrea De Salve, Roberto Di Pietro, Paolo Mori, and Laura Ricci, "A Logical Key Hierarchy Based approach to preserve content privacy in Decentralized Online Social Networks", IEEE Transactions on Dependable and Secure Computing, 2017, pp. 1–20.
[2] Jonghyuk Song, Sangho Lee, and Jong Kim, "Inference Attack on Browsing History of Twitter Users using Public Click Analytics and Twitter Metadata", IEEE Transactions on Dependable and Secure Computing, 2016, pp. 1-16.
[3] A. De Salve, P. Mori, and L. Ricci, "A Privacy-Aware Framework for Decentralized Online Social Networks," in Database and Expert Systems Applications. Springer, 2015, pp. 479–490.
[4] Xun Yi, Elisa Bertino, Fang-Yu Rao, and Athman Bouguettaya, "Practical Privacy-Preserving User Profile Matching in Social Networks", IEEE 32nd International Conference on Data Engineering, 2016, pp. 373-384.
[5] Sanaz Taheri Boshrooyeh, Alptekin Kupcu, and Oznur Ozkasap, "Security and Privacy of Distributed Online Social Networks", IEEE 35th International Conference on Distributed Computing Systems Workshops, 2015, pp. 112-119.
[6] L. M. Aiello and G. Ruffo, "Lotusnet: tunable privacy for distributed online social network services," Computer Communications, 2012, pp. 75–88.
[7] P. Maymounkov and D. Mazieres, "Kademlia: A peer-to-peer information system based on the xor metric", in International Workshop on Peer-to-Peer Systems. Springer, 2002, pp. 53–65.
[8] Chung Kei Wong, Mohamed Gouda, and Simon S. Lam, "Secure Group Communications Using Key Graphs", IEEE/ACM Transactions on Networking, Vol. 8, No. 1, February 2000, pp. 16–30.
[9] Bo Tuan, Lu Liu, and Nick Antonopoulos, "Efficient Service Discovery in Decentralized Online Social Networks", IEEE/ACM 3rd International Conference on Big Data Computing, Applications and Technologies, 2016, pp. 73-78.
[10]Lorenz Schwittmann, Matthaus Wander, Christopher Boelmann and Torben Weis, "Privacy Preservation in Decentralized Online Social Networks", IEEE Internet Computing, 2014, pp. 16-23.
[11]Arun Thapa, Weixian Liao, Ming Li, Pan Li, and Jinyuan Sun, "SPA: A Secure and Private Auction Framework for Decentralized Online Social Networks", IEEE Transactions on Parallel and Distributed Systems, 2015, pp. 1-14.
[12]L. A. Cutillo, R. Molva, and T. Strufe, "Safebook: A privacy-preserving online social network leveraging on real-life trust", Communications Magazine, IEEE, 2009, pp. 94–101.
[13]Nikita V. Ghodpage, and R. V. Mante, "Privacy Preserving and Information Sharing in Decentralized Online Social Network", in Proceedings of 2nd International Conference on Inventive Communication and Computational Technologies (ICICCT 2018)IEEE, pp. 150-153.