Volume: 7 Issue: III Month of publication: March 2019

DOI: http://doi.org/10.22214/ijraset.2019.3138

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 7 Issue III, Mar 2019- Available at www.ijraset.com

Group Key Agreement Scheme with Privacy Preservation for Social Media Platform

Sundari K[1], Shahaya Ashwini A[2], Dharani R[3]
[1, 2, 3]Department of Computer Science and Engineering, University College of Engineering Nagercoil,TamilNadu.

Abstract: Network security is protection of the access to files and directories in a computer network against hacking, misuse and unauthorized changes to the system. Online social network becomes the most important medium of information propagation. As the number of social network users grows rapidly and information sharing behavior becomes more frequent, the propagation range of personal information becomes more widespread. This phenomenon gradually arouses the user's high concern for the privacy protection of personal information. In existing a trust-based mechanism to realize collaborative privacy management. Basically, a user decides whether or not to post a data item based on the aggregated opinion of all involved users. The trust values between users are used to weight users' opinions, and the values are updated according to users' privacy loss. To enhance the trustworthiness of the online social network systems, propose a secure chaotic maps-based group key agreement scheme. In this proposed scheme, provide member anonymity to ensure the privacy of the communication between the social networking platform and the members

The proposed solution does not rely on a centralized online key centre or a trusted group chairman, thus ensuring fairness.

The mechanisms of message encryption and member verification into the scheme to allow the members to anonymously interact

I. INTRODUCTION

Online social networks (OSNs), such as Facebook, Google+ ,and Twitter, have become the most important platforms for people to make social connections with others. Thousands of

millions of users post data about their daily lives

in terms of text messages, photos, or videos on OSNs. Such data often contain sensitive information of users. If the data can be accessed by unauthorized entities, users' privacy will be com promised. The privacy issue has always been a major concern in studies related to OSNs [1], [2], [3], [4].To protect users' privacy, on one hand, the service providers of OSNs need to take measures to prevent data breach. On the other hand, users themselves can control the access to their data by using the privacy setting function implemented in OSNs [5]. An access control policy, also referred to as the privacy policy, defines which users are allowed to access a user's data. Current OSNs often utilize user relationship to distinguish between authorized users and unauthors .For example, Face book users can specify if their data can be accessed by friends, specific groups or everyone. The privacy control mechanisms implemented in current OSNs only impose restrictions on users who want to access others' data. While there is no strict restriction on users who post data. A consequence of this one-side restriction is that the user who posts data may unintentionally violate other users 'privacy. Consider the following example. Suppose that a user A posts a photo of him/her playing with a friend B, and user A specifies that the photo can be accessed by his/her colleagues. If user B considers this photo to be sensitive and user B is not familiar with user A's colleagues, then user B's privacy will be violated. In the above case, the photo is actually co-owned by the two users. Hence, the privacy policy specified by user A should be compatible with user B's privacy policy, otherwise, user B will suffer a loss in privacy. Data which are co-owned by multiple users are quite common in OSNs. Privacy management of such data require a collaboration of all involved user.

II. EXISTING TECHNIQUES

OSNs only impose restrictions on users who want to access others' data. While there is no strict restriction on users who post data. A consequence of this one-side restriction is that the user who posts data may unintentionally violate other users' privacy. Consider the following example. Suppose that a user A posts a photo of him/her playing with a friend B, and user A specifies that the photo can be accessed by his/her colleagues. If user B considers this photo to be sensitive and user B is not familiar with user A's colleagues, then user B's privacy will be violated. In the above case, the photo is actually co-owned by the two users. Hence, the privacy policy specified by user A should be compatible with user B's privacy policy, otherwise, user B will suffer a loss in privacy. Data which are co-owned by multiple users are quite common in OSNs. Privacy management of such data requires a collaboration of all involved users.

The problem of collaborative privacy management in OSNs has attracted much attention in recent years. Most studies deal with this problem by first detecting the conflicts among different users' privacy policies, and then generating an aggregated policy that can resolve the conflicts to the largest extent. Given a data item (e.g. a photo), a user's privacy policy is generally represented by a set of users with whom the user wants to share the data. Usually there is a mediator who collects users' policies and makes a group decision via some aggregation scheme. In most cases, the conflicts among users' privacy policies cannot be completely eliminated, which means the aggregated policy may still cause a privacy loss to some of the users.

III. PROPOSED TECHNIQUES

The proposed DAGKA protocol consists of the following Algorithms or procedures. System Setup: Given a security parameter *k* ∈Z+, the algorithm works as follows. KGC generate a prime *q*, two groups G1, G2 of order *q* and an admissible bilinear map *e:* G1 × G1 −→ G2 as described in Section 2.1. Next, KGC chooses random *P*, *Q* ∈G1, a random *s* ∈*Z∗q* and a cryptographic hash function *H:{*0, 1*}∗*−→ G1. Then KGC sets *Ppub* = *sP* as its public key. *s* is set as the master secret key.

Extract: On input identifier ID ∈ {0, 1}∗, KGC computes the private key of ID as SID = sH(ID) for ID. -Setup: {u1, u2, ..., un} is a set of users who want to establish a session key.

1) User i (1 ≤ i ≤ n) chooses random ri ∈Z∗q, computes and broadcasts Pi = r_iP, Vij =r_i(Q + Qj)(1 ≤ j ≤ n, j _= i), keeping ri secret.

2) Upon receiving Pj, Vji (1 ≤ j ≤ n, j _= i), user I computes the session key as sk = e (_j=n j=1, j_=i Vji +ri(Q+ Qi), Ppub)e(Si,−_j=n j=1 r_jP) = e(_j=n j=1 r_jQ, Ppub).

3) Finally, each user k stores Pi, Vij

(1 ≤ i ≤ n, 1 ≤ j ≤ n, j _= i) and his ephemeral secret value rk used in this session (1 ≤ k ≤ n).

-Join: Assume that u_{n+1}, ..., u_{n+m} will join the group *{u*1, ..., *un}* resulting in a group *{*u_1, ..., u_{n+m}*}*. (1) Each user *i*(1 ≤ *i* ≤ *n*) computes V_{ij}= r_i(*Q*+*Qj*)(*n*+1 ≤ *j* ≤*n*+*m*) and broadcasts P_i, V_{ij}(1 ≤ *j* ≤ *n*+*m*, *j* _= *i*), where r_i, *Pi* = $\mathbf{r_i P}$ and *Vij* (1 ≤ *j* ≤ *n*, *j* _= *i*) are the values stored in the past session. Each user *n* + *i* (1 ≤ *i* ≤ *m*) chooses

random r_{n+i} ∈Z∗*q* and computes

P_{n+i}= r_{n+iP}, $V_{(n+i)\,j}$= r_{n+i}(Q + Q_j)

for (1 ≤*j* ≤ *n* + *m*, *j* _= *n* + *i*).

Leave: Suppose that u_{m+1}, ..., $u_n$will leave the group *{u*1, ..., *un}* resulting in a group *{*u_1, ..., u_m*}*. (1) Each user *i* (1 ≤ *i* ≤ *m*) uses *Pj*, *Vji* stored in the past session to compute the group session key as

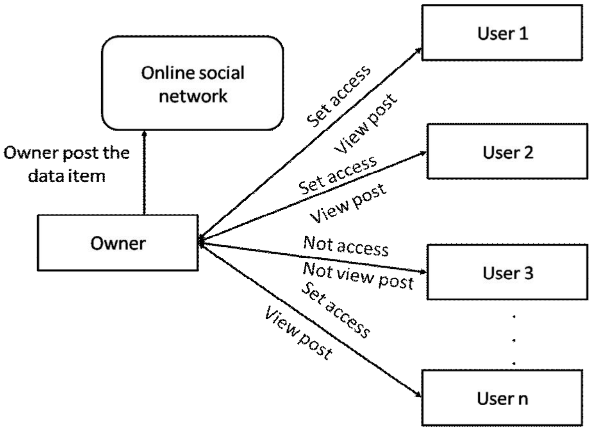

Fig 1. Block Diagram

$$sk = e(\sum_{j=1}^{j=m} V_{ji} + r_i(Q + Q_j) Ppub)e(S_{i,} - \sum_{j=1}^{j=m} r_j P = \sum_{j=1}^{j=m} r_j Q, Ppub$$

).

Then each user *i* deletes *Pj*, *Vji* (*m* + 1 ≤ *j* ≤ *n*, 1 ≤ *i* ≤ *n* + *m*, *i* _= *j*).

IV. IMPLEMENTATION

A. *Software Environment*

1) *Features of. Net:* Microsoft .NET is a set of Microsoft software technologies for rapidly building and integrating XML Web services, Microsoft Windows-based applications, and Web solutions. The .NET Framework is a language-neutral platform for writing programs that can easily and securely interoperate. There's no language barrier with .NET: there are numerous languages available to the developer including Managed C++, C#, Visual Basic and Java Script. The .NET framework provides the foundation for components to interact seamlessly, whether locally or remotely on different platforms. It standardizes common data types and communications protocols so hat components created in different languages can easily interoperate.".NET" is also the collective name given to various software components built upon the .NET platform. These will be both products (Visual Studio.NET and Windows.NET Server, for instance) and services (like Passport, .NET My Services, and so on).

B. *Database And Sample Implementation*

1) *Features of SQL-Server:* The OLAP Services feature available in SQL Server version 7.0 is now called SQL Server 2000 Analysis Services. The term OLAP Services has been replaced with the term Analysis Services. Analysis Services also includes a new data mining component. The Repository component available in SQL Server version 7.0 is now called Microsoft SQL Server 2000 Meta Data Services. References to the component now use the term Meta Data Services. The term repository is used only in reference to the repository engine within Meta Data Services

SQL-SERVER database consist of six type of objects, They are,

a) TABLE
b) QUERY
c) FORM
d) REPORT
e) MACRO

C. *Sybil Identification Algorithm*

A Sybil identification algorithm that takes the social graph G (V, E), a known honest node h, and a suspect node u as input, and outputs whether u is Sybil or not. This algorithm is based on random walks. A sequence of moves of a particle between nodes of G is term as random walk. If the particle is at node i with degree di, then the probability that the particle follows the edge (i, j) and moves to a neighbour j is 1/di.

The main idea behind this Sybil identification algorithm is that, as there is a small cut between the honest region and the Sybil region, the random walks originating from a Sybil node tend to get "trapped" into the Sybil region. Also, because it assumes that the size of the Sybil region is not comparable to the size of the honest region. The number of nodes traversed by the random walks originating from an honest node will be larger than the number of nodes traversed by the random walks originating from a Sybil node, as long as the random walks are long enough to exhibit the difference between the Sybil region and the honest region, and it performs the random walks many times. For simplicity, it defines the number of times one node being traversed by a set of random walks as the frequency of that node.

D. *Sybil Community Detection Algorithm*

After one Sybil node is identified, The Sybil community detection algorithm can be used to detect the Sybil community surrounding it. The Sybil community detection algorithm takes the social graph G(V, E) and a known Sybil node as input, and outputs the Sybil community around us. The Sybil nodes can be identified by using Sybil identification algorithm or any previous scheme. It defines a Sybil community as a subgroup of G consisting of only Sybil nodes, and there is no small cut in this sub graph.

The reason it makes this definition is that if a small cut does divide the Sybil region into two parts S1 and S2, and the known Sybil nodes is s in S1, then, from the point of view of us, the honest region and S2 are similar, because there is already a small cut between S1 and the honest region and also a small cut between S1 and S2. When there is a small cut in the Sybil region, this algorithm can detect the Sybil community s. This algorithm based on performing partial random walks originating from s. Each partial random walk behaves the same as the simple random Walks used in the Sybil identification algorithm, except that it does not traverse the same node more than once. Therefore, when a partial random walk reaches a node with all the neighbours traversed by itself, this partial random walk is "dead" and cannot proceed.

This property makes a partial random walk originating from a Sybil node less likely to leave the Sybil region, compared with a simple random walk, because many such walks "die" when they hit the border of the Sybil region. Similar to the Sybil identification algorithm, the intuition behind this algorithm is that the partial random walks originating from a Sybil node tend to be trapped within the Sybil region, and thus, it can detect the Sybil
community by examining the nodes traversed by the partial random walks.

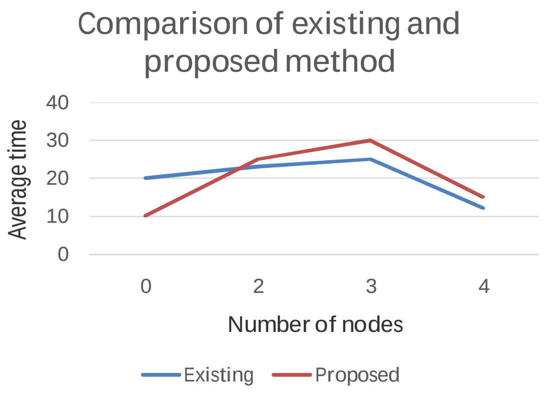

Fig 2: Performance evaluation

Number of users	Existing (Time)	Proposed (Time)
0	2.5	1.5
2	2.3	1.4
3	2.8	2.5
4	3.5	2.4

V. CONCLUSION

DAGKA protocol was constructed. The proposed protocol requires only one round in Setup and Join algorithms. In Leave algorithm, there is no message transmissions among remaining members. Previous session key are protected from joining members and subsequent session keys are protected from leaving members. Its AKE-security was proved under DBDH assumption. It resists key control attack and provides forward security.

REFERENCES

[1] Ingemarsson, D. Tang, and C. Wong, "A conference key distribution system," IEEE Transactions on Information Theory, vol. 28, no. 5, pp. 714–720, 1982.

[2] E. Bresson, O. Chevassut, D. Pointcheval, and J.- J. Quisquater, "Provably authenticated group diffiehellman key exchange," in 8th ACM Conference on Computer and Communications Security, CCS '01. ACM, 2001, pp. 255–264

[3] Y. Zhang, J. Chen, H. Li, J. Cao, and C. Lai, "Groupbased authentication and key agreement for machine type communication," International Journal of Grid &Utility Computing, vol. 5, no. 2, pp. 87–95, 2014

[4] N. B. Bhavesh, S. Maity, and R. Hansdah, "An authentication protocol for vehicular ad hoc networks with heterogeneous anonymity requirements," InternationalJournal of Space-Based and Situated Computing, vol. 4, no. 1, pp. 1–13, 2014.

[5] J. Katz and M. Yung, "Scalable protocols for authenticated group key exchange," in Advances in Cryptology– CRYPTO 2003, ser. Lecture Notes in Computer Science 2729. Springer, 2003, pp. 110–125.

[6] K. Neupane and R. Steinwandt, "Communication efficient 2-round group key establishment from pairings," in Topics in Cryptology – CT-RSA 2011, ser. Lecture Notes in Computer Science 6558. Springer, 2011, pp. 65–76.

[7] M. C. Gorantla, C. Boyd, J. M. Gonz´alez Nieto, and M. Manulis, "Generic one round group key exchange in the standard model," in Information, Security and Cryptology – ICISC 2009, ser. Lecture Notes in Computer Science 5984. Springer, 2010, pp. 1–15.

[8] J. M. Bohli, M. I. G. Vasco, and R. Steinwandt, “Secure group key establishment revisited,” InternationalJournal of Information Security, vol. 6, no. 4, pp. 243–254, 2007.

[9] S. Li and F. Zhang, “Leakage-resilient identity-based encryption scheme,” International Journal of Grid &Utility Computing, vol. 4, no. 2/3, pp. 187–196, 2013.

[10] S. Luo and Z. Chen, “Hierarchical identity-based encryption without key delegation in decryption,” International Journal of Grid & Utility Computing, vol. 5,no. 5, pp. 71–79, 2014.

[11] X. Sun, Z. T. Jiang, M. R. Zhou, and Y. Wang, “Versatile identity-based signatures for authentication in multi-user settings,” International Journal of Grid& Utility Computing, vol. 5, no. 3, pp. 156–164, 2014.

[12] K. C. Reddy and D. Nalla, “Identity based authenticated group key agreement protocol,” in Progress inCryptology – INDOCRYPT 2002, ser. Lecture Notes in Computer Science 2551. Springer, 2002, pp. 215–233.

[13] K. Y. Choi, J. Y. Hwang, and D. H. Lee, “Efficient id-based group key agreement with bilinear maps,” in Public Key Cryptography – PKC 2004, ser. Lecture Notes in Computer Science 2947. Springer, 2004, pp. 130–144

[14] Y. Shi, G. Chen, and J. Li, “Id-based one round authenticated group key agreement protocol with bilinear pairings,in International Conference on Information Technology: Coding and Computing. IEEE, 2005, pp.757–761.

[15] E. Bresson, O. Chevassut, and D. Pointcheval, “Dynamic group diffie-hellman key exchange under standard assumptions,” in Advances in Cryptology – EUROCRYPT 2002, ser. Lecture Notes in Computer Science 2332. Springer, 2002, pp. 321–336.

[16] R. Dutta and R. Barua, “Provably secure constant round contributory group key agreement in dynamic setting,” Information Theory IEEE Transactions on, vol. 54, no. 5, pp. 2007–2025, 2008.

[17] H. J. Kim, S. M. Lee, and H. L. Dong, “Constant-round authenticated group key exchange for dynamic groups, ”in Advances in Cryptology - ASIACRYPT 2004, ser. Lecture Notes in Computer Science 3329. Springer,2004, pp. 245–259.

[18] J. K. Teng, C. K. Wu, and C. M. Tang, “An id based authenticated dynamic group key agreement with optimal round,” Science China Information Sciences, vol. 55, no. 11, pp. 2542–2554, 2012.

[19] Q. Cheng and C. Tang, “Cryptanalysis of an id-based authenticated dynamic group key agreement with optimal round,” International Journal of Network Security, vol. 17, no. 6, pp. 678–682, 2015.

[20] L. Zhou, W. Susilo, and Y. Mu, “Efficient id-based authenticated group key agreement from bilinear pairings,” in Mobile Ad-hoc and Sensor Networks, Second International Conference, MSN 2006, ser. Lecture Notes in Computer Science 4325. Springer, 2006, pp. 521–532.

[21] E. Bresson, O. Chevassut, and D. Pointcheval, “Provably authenticated group diffie-hellman key exchange - the dynamic case (extended abstract),” Office of Scientific & Technical Information Technical Reports, pp. 290–309, 2001.