Collusion Free Trusted Access Control For Social Networks

S.R.Krithika Bharathi

PG Scholar
Department of Computer Science and Engineering
Sri Shakthi Institute of Engineering and Technology
Coimbatore, Tamilnadu, India

Kavitha V.Kakade

Assistent Professor
Department of Computer Science and Engineering
Sri Shakthi Institute of Engineering and Technology
Coimbatore, Tamilnadu, India

Abstract - Social network plays important role in our life. It is used to stay connected with friends and share what is happening right now. It provides a good digital interaction and information sharing. Still there is no sophisticated technique to enforce privacy concerns over data associated with multiple users. And there is no trust management among various kinds of users, it leads to unwanted behaviors such as phishing attacks, spreading of rumors which lead to serious consequences. We propose an approach to enable the collusion free model to achieve security of sharing data among multiple users and it has efficient content sharing and trust management. Content sharing avoids spreading of rumors and it protects from phishing attack when URL is shared to our space.

Key words: access control, collusion, social network rumor, phishing attack

I.INTRODUCTION

A typical Online Social Network(OSN) provides each user with a virtual space containing profile information which has user's birthday, gender, interest, education, work history and contact. And a list of the user's friends and web pages such as wall in facebook. To ensure privacy of user data, current OSN makes users to be system and policy administrator for control their own data. Although simple access control mechanisms allowing users to govern the data which is resided in their own space but they can't regulate data residing outside their spaces. For example we can't specify who can view our comment which is posted in our friend's space, it will be visible to all.OSN only has binary decision (to keep or to delete) from OSN mangers is either too loose or too restrictive. So it is very essential to implement an effective and flexible access control mechanism for OSNs. And also rumors can be spread through social networks rapidly. It is an unwanted thing that must be reduced. Attackers use Social networks to share fake URL with attractive advertisement, when the user who is unaware of it will click the URL, it will be automatically redirected to the attackers site, in that they will hack the users information like credit card numbers. There is no efficient technique to detect these kinds of fraudulent behaviors. In this paper a systematic solution to achieve trust management among multiple associated users is proposed and it has additional features to avoid social network rumors and phishing attack by efficient content sharing.

II. MULTIPARTY ACCESS CONTROL MODEL

OSNs are multi user environment. It should allow multiple controllers who are associated with the shared data, to specify the access control policies. There are four types of controllers as follow:

Owner: If d is a data item in the space of a user u in the social network, Then the user u will be the owner of d

Contributor: If d is a data item published by a user u in someone else's space in the social network, Then user u is called the contributor of d.

Stakeholder: Let d is a data item in the space of a user in the social network. Let T be the set of tagged users associated with d. if u∈ T, then user u is called a stakeholder of d.

Disseminator: Let d is a data item shared by a user u from someone else's space to his/her space in the social network. The user u is called a disseminator of d.

III. POLICY SPECIFICATION

To enable a collaborative authorization management of sharing data in OSNs, it is vital for MPAC policies to be in place to regulate access on shared data, representing authorization requirements from multiple associated users. Policy specification scheme is built upon the MPAC model.

Accessor specification: Accessors are a users' set who are granted to access the shared data in OSN. Accessors can be represented with a set of user names(UNs), a set of relationship names (RNs) or a set of group names (GNs) in OSNs.

Definition: Let ac ∈ U ∪ RT ∪ G be a user u ∈ U, a relationship type rt ∈ RT, or a group g ∈ G. Let at ∈ {UN, RN,GN}be the type of the accessor specification (user name, relationship type, and GN, respectively).Accessor specification is a set, accessors= {a1,...,an}, where each element is a tuple <ac, at>

Policy: A multiuser access policy has 5-attributes as,

P=<controller, ctype, data, accessor, effect> where

- Controller is a user who regulates the data access;
- Ctype describes the type of the controller;
- Data represents a specification of data
- Accessor is a set of authorized users
- Effect will be{permit, deny}

Likewise each user has own policy.

IV. POLICY EVALUATION

To evaluate an access request on policies, the first step is to check the access request a to yield the decision from the controller according to their policy.The accessor attribute in the policy will decide whether the policy is applicable to a request. When user who sends the request and he/she belongs to the user set, the policy is applicable and the evaluation process returns a response with the decision (permit or deny) indicated by the effect element in the policy. Else, the response will return deny decision if the policy is not applicable to the request. Finally, decisions from all controllers are aggregated to make a final decision for the access request.

Fig. 1 shows the policy evaluation mechanism and decision making while requesting.

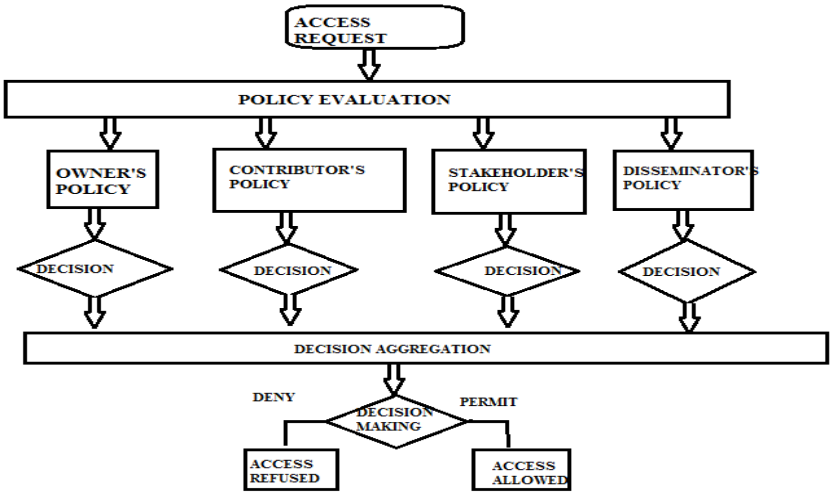

Figure. 1.policy evaluation

V. SYSTEM ARCHITECTURE:

Fig. 2 shows the architecture of MController, which is divided into two major parts: Facebook server and application server. The Facebook server is an entry point via the Facebook application page, and provides references to photos, friendships. It accepts inputs from users, and then it forwards them to the application server. The application server is in charge for the input processing and collaborative management of shared data. User data such as user identifiers, friend lists, groups, and contents are stored in the application database.

Users can access the MController application through Facebook. When access requests are made to the decision-making portion in the application server, results are returned to access photos or proper information about access to photos.

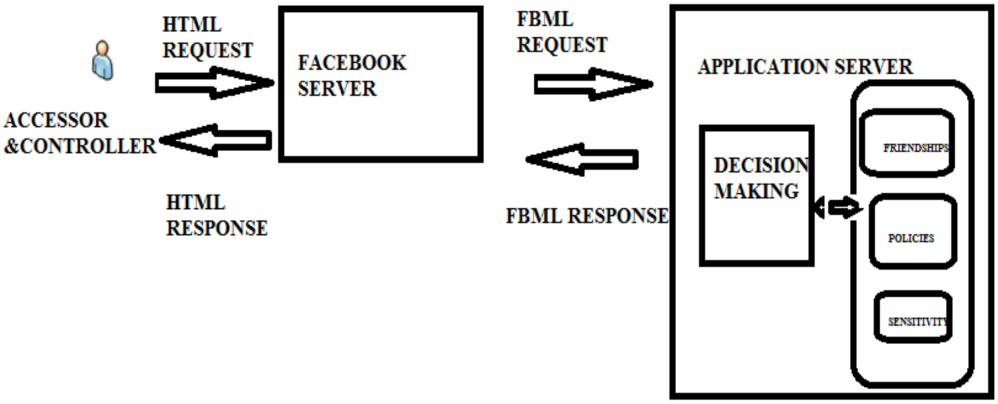

Figure. 2.overall architecture

When privacy changes are made, the decision-making portion will return change impact information to alert the user. Users can control the analysis services to perform complicated authorization queries. MController is developed as a third-party Facebook application, it can be hoseted in an Apache Tomcat application server supporting PHP and MySQL database.

VI. IMPLEMENTATION AND EVALUATION

MController is developed as a third-party Facebook application, which is hosted in an Apache Tomcat application server supporting PHP and MySQL database. User can install MController in her/his Facebook space and accepts the necessary permissions, MController can access a user's basic information and contents. It can retrieve and list all photos of user, or where the user was tagged. After information is imported, the user can access MController through its application page on Facebook, where she/he can query access information, set privacy for photos that she/he is a controller, or view photos she/he is allowed to access.

A. Decision making-

Decision making module is a core component of a MController which deals with multiparty access control by processing access request and returns responses (permit or deny). To estimate an access request, first the policies of each controller of the targeted content are enforced to generate a decision for the controller. Finally, the decisions of all controllers are aggregated to get a decision. Configured conflict resolution mechanism is used to resolve Multiparty privacy conflicts when aggregating the decisions of controllers. Sensitivity values are used to take decisions. For each data or photo, controller saves his/her own privacy setting by specifying how many users is allowed and how many users are restricted. MController can also display the details of all users who violate against the controller's privacy setting. The purpose of such feedback information is to guide the controller to evaluate the impact of collaborative authorization. If the current privacy policy is not fulfilled, then the controller can adjust her/his privacy setting, contact the owner of the photo to ask her/him to change. Advanced queries are to know if there is any undersharing or oversharing.

B. Collusion avoidance and trust management-

While sharing the content with friends some time there may be more than one person with the same name, in such situation there is a chance of collusion, and the controller has to view the profiles of all the stakeholders, it will increase the delay. To avoid it pattern matching is used, it displays the details of those users, from that appropriate user is identified. This feature is not available in current OSNs, and also there are no trust management mechanisms.

Consider the scenario, when a controller wants to share an item with particular group of users, and in that group there are some unknown people are there. In this case controller can't share the item with that group since it will not be secure. To predict trust rate of the unknown member, the controller will send request to other members of that group. If the member doesn't know about the person, again the same request is forwarded to other member (friend of friend) According to the response from the members, trust value is calculated and decision is made.

Trust value is calculated from the following equation:

$$t_{is} = \frac{\sum\limits_{j \in adj(j) \ | \ t_{ij} \ \geq \ max} t_{ij} t_{js}}{\sum\limits_{j \in adj(j) \ | \ t_{ij} \ \geq \ max} t_{ij}}$$

C.Efficient content sharing-

When contributor shared any information which is a rumor in the owner's space, then again there is a chance to disseminate the rumor. Likewise it is redisseminated by other disseminators and spread to all. To avoid this behavior, the content is restricted to post in the controller's space temporarily if it has any sensitive words. The controller reviews the content or sends trust request to friends to know about the person before the controller allows to post the content in his/her space. So rumor is not spread in the network.

And the next scenario is, when we post any comment in the friend's space we can't specify who can view the comment, because it is not present in the controller's space since it is out of the controller's space. Proposed system has the feature to specify who can view the comment.

VII. CONCLUSION

In this paper, we have proposed a novel solution for trust management and collusion of user while sharing data among multiple associated users. And also this model has additional features like efficient content sharing which has rumor filter that is essential to avoid unwanted spreading of fake information.

REFERENCES

[1] B. Carminati and E. Ferrari, "Collaborative Access Control in On-Line Social Networks," Proc. Seventh Int'l Conf. Collaborative Computing: Networking, Applications and Worksharing (Collaborate-Com),pp. 231-240, 2011

[2] B. Carminati, E. Ferrari, and A. Perego, "Enforcing Access Control in Web-Based Social Networks," ACM Trans. Information and System Security,vol. 13, no. 1, pp. 1-38, 2009.

[3] P. Fong, "Relationship-Based Access Control: Protection Model and Policy Language,"Proc. First ACM Conf. Data and Application Security and Privacy,pp. 191-202, 2011.

[4] P. Fong, M. Anwar, and Z. Zhao, "A Privacy Preservation Model for Facebook-Style Social Network Systems," Proc. 14th European Conf. Research in Computer Security,pp. 303-320, 2009

[5] H. Hu, G. Ahn, and K. Kulkarni, "Anomaly Discovery and Resolution in Web Access Control Policies,"Proc. 16th ACM Symp. Access Control Models and Technologies,pp. 165-174, 2011.

[6] A. Squicciarini, M. Shehab, and F. Paci, "Collective Privacy Management in Social Networks,"Proc. 18th Int'l Conf. World Wide Web,pp. 521-530, 2009