An Application for Protecting Personal Information on Social Networking Websites

Mehmet Erkan Yüksel
Computer Engineering Department
Faculty of Engineering
Istanbul University
Istanbul, Turkey
e-mail: eyuksel@istanbul.edu.tr

Asım Sinan Yüksel
Computer Science Department
School of Informatics and Computing
Indiana University
Bloomington/Indiana, USA
e-mail: asyuksel@umail.iu.edu

***Abstract*— We redundantly share our personal information and applications with people on the Internet. Depending on this, social networking websites have also become indispensable parts of our lives and allow the users to share just about everything: photos, videos, favorite music, and games. Sharing large amounts of information causes privacy problems for the users in these websites. In order to prevent these problems, we can provide trusted and built-in applications that help to protect our privacy by limiting the friends who get access to our personal information and applications. Thus, the security and privacy problem has prompted us to provide a solution that offers the users of these social networking websites an opportunity to protect their information. In this paper, an application that can be used in social networking websites, its design, algorithm and database structure are mentioned. Our application offers a trusted architecture to the social network users. It finds social circles and helps the users to group their friends easily and meaningfully for protecting their privacy and security. This system provides grouping of users through an automated system into different social circles by analyzing the user's social situation and depending on what common information or application they would like to share that should not be accessed by other users.**

***Keywords*— social networking websites; clustering; sharing information; protecting privacy; graph database**

I. INTRODUCTION

Meeting new friends and socializing are parts of our lives. With great advances being made at this age of information technology, socialization has greatly increased with people being able to meet and communicate friends from different regions of the world by using social networking websites. These websites enable friends to easily communicate online, and provide many Internet features and functionalities for social network users such as publishable personal profiles, repositories for sharing information and applications, and the abilities to provide social connectivity between the users. Although social networking websites in the Internet offer an opportunity to meet and communicate many friends; it creates a privacy and security problem because people of all ages, interests and backgrounds have free access to social networking websites, and you may not want to share some of your personal information with some of your friends or network users who you do not know. In these websites, there are a number of cases where the users have been able to identify and locate other users through the personal information that was posted. Inappropriate information might be published that leads computer hackers, sexual predators and other malicious users to alter the person's profile and information or to access their computer. Users can find damaging information about a person's past and they can learn what he/she is doing on the Internet. Therefore, the users must allow as many or as few friends to view their personal web pages by choosing some kind of restrictions. They must determine the accessing permissions using tools on the website.

Building personal web pages and using social networking technologies, services and applications can be a very creative, useful, effective and beneficial outlet for users to share and express their thoughts and opinions, to learn how to manipulate and use large amounts of information, and to learn skills needed to build web pages and applications. Most popular examples including Facebook, Twitter, MySpace, and Hi5 are public social networking websites offering free accounts to the users to share personal information such as "About Me", "My Friends", sexual orientation, emails, message boards, religion, politics, user groups, favorite tunes, movies/videos, interests, preferences, education achieved, networking organizations, photographs, applications and other information about themselves. However, social networking websites have potential effects on people's life, and there are very serious privacy issues when these websites are not used appropriately.

Personal information like your profile that is posted on a social network can be accessed by all your friends that you share the network with. Unauthorized people may also get access to some of your personal information that you do not want to share. We must know what is appropriate to put on the web pages, and be clear about what is not safe to post on the web: full name, address, specific places we go, phone numbers, ethnicity, and anything else that would help someone identify or locate us. Once something is posted on the web, it is no longer private [1, 2].

 ISBN: 978-1-61208-100-7

Social networking websites increase popularity of the Internet usage, for the purpose of socializing and networking with users across the world, and they are becoming a growing issue of concern for researchers. Therefore, protecting privacy, sharing information and applications in social networking websites are really important issues. These websites provide some features for protecting privacy, and controlling what information can be accessed. However, most people are unaware or do not know how to use these features. Even if users were to perform these tasks of categorization, on what basis would they categorize their friends in a meaningful way to set privacy and security policies? Our study proposes an application to help the users to make better decisions about their privacy settings.

The remainder of this paper is organized as follows: In Section 2, we provide a literature review highlighting the works already carried out in this area, explain what we want to achieve, and reveal what was/is missing. We present the details of our application, application design platform, algorithm design, clustering approach and graph based database design in Section 3. In conclusion, we discuss the future directions, limitations, contributions of our study.

II. RELATED WORK

Social network theorists have discussed the relevance of relations of different depth and strength in a person's social network. In a recent study [3], the privacy relevance of these arguments has recently been studied and researchers concentrated on the role and importance of social connections as we call social circles. In a study by [4], researchers reveal the relation between personal information, privacy and a user's social network. They state that a social network provides a visual map of the relevant social connections between the nodes of participation which can be used to measure the degree of connectivity. This work is one of the studies we inspired and supports our idea of protecting personal information by creating social circles with their crucial explanation "Safety must be first and foremost because we want to share information about ourselves to be known only by a trusted circle of close friends, and not by anonymous strangers or distant friends who does not know us better."

In [5], researchers studied the information disclosure in social networks, and they found that by looking at certain characteristics, such as knowing which groups people belong to or their favorite applications, it was possible to predict their political affiliation.

In [6], Canadian Privacy Commissioner published a must-read report about personal information protection on Facebook. This report clearly supports our idea of improving and simplifying the privacy, but it does not go beyond further than being a criticism. We believe that our study will inspire Facebook developers to implement more user friendly, more successful privacy management features.

All of the recent researches show the importance of protecting information in social networks. Lack of the privacy in social networks causes some members to unregister so as to protect their privacy. Our study differs from recent studies. Instead of proving the existence of privacy problems and presenting attacks, we proposed a solution and its implementation for current problems that social network users encounter.

III. APPLICATION

Our application provides an implementation of a web based solution to protect personal information. It helps the users to automatically categorize a large number of friends into meaningful lists. The main assumption we make to build the social circles is that users would mostly present similar information to all friends in a social group, and therefore social circles provide a meaningful and trusted categorization of friends for setting privacy policies [7, 8].

Our application interface design has two aims. The first aim is to discover whether social circles exist on a social networking website. The second aim is to discover whether these social circles would help the users in social networking applications in setting effective privacy and security policies. In our system, we have developed a trusted application which is shown in Fig. 1 to identify the social circles in social networking websites. The users can add this application to their personal web pages on any social networking website (e.g., Facebook, Twitter or MySpace). The users have been asked randomly generated questions about their willingness to share a piece of their information with a social network friend of theirs. These questions are based on the fields of social networking website database tables that are available for application developers. Each question is formed in a way which does not reveal the real aim of the study, and does not disturb the users. This is to prevent the bias such as evaluating the concept of trusted social circles in the context of privacy and security. The answers to the questions are saved in our secure, anonymized graph database. This data collection method provides us with quantitative results that we can statistically analyze. When all questions are answered, the application runs the clustering algorithm and finds the visual graph of users.

Figure 1. Main Page of Our Application

We have developed a web application which finds social circles of the users in their social networks. Users can add this application to their personal pages which are stored on social networking websites such as Facebook, Twitter, and MySpace. Users are able to delete this application after they have completed their studies. Our application is built on a trusted structure and suitable for protecting privacy. It provides the following features:

1) Creating Visual Graph of Social Circles

As shown in Fig. 2, our application produces such a graph that helps users to see each social circle and to make better decisions about their applications and privacy settings.

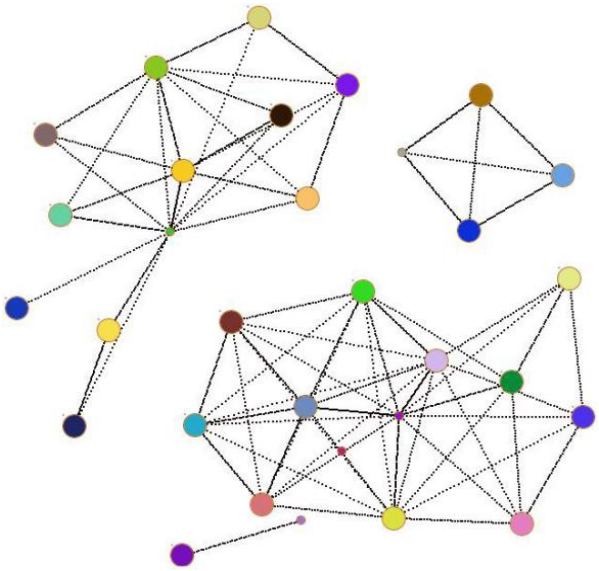

Figure 2. Visualizing Users' Social Graphs.
(Friend pictures are anonymized by using circles)

2) Suggested-Settings For Proteting Privacy

Our application suggests the set of friend lists that users should create, and the friend lists into which they should put each of their current friends based on the identified social groups. It is also designed to attach importance to user's privacy and security on social networking websites.

3) Graph Database For Effective Data Representation

We used a simple, robust, massive scalability, and convenient object-oriented graph database structure that provides an intuitive graph-oriented model for data representation and collection. This database is an embedded, disk-based, fully transactional, more effective and flexible system that stores data structures in graphs rather than in tables. Instead of static and rigid tables, rows and columns, our application works with a flexible graph network consisting of nodes, relationships and properties.

A. Application Design Platform

Applications for social networking websites can be created by using a variety of software technologies, including HTML, XML, OpenSocial Templates, JavaScript, CSS, Flash, Python, Java, Perl, PHP, .NET, or Ruby on Rails. This section gives us several approaches for developing our application, depending on our project requirements.

Most social network application designs in the Internet have similar structures: social network data, application data, and an appropriate template are used to provide a rendered view to the user. In many social networking websites, these components can come from several places. Client-side applications can use scripting language such as JavaScript and VBScript to render data into a template. Social networking websites can store both social network and application data, and server-side applications can take advantage of databases and server-side frameworks to produce rendered output.

Our application runs inside of a social networking website such as Facebook and MySpace but relies on an external server or host for processing and rendering data. It can provide advanced functionality but may run into scaling problems when the users increase so much. Its platform consists of some hardware and software components. These components are given below:

- A markup language derived from HTML
- General-purpose scripting languages PHP
- JavaScript scripting language
- MySQL database language for interacting with social networking website database.
- Object-based web API for handling communication between a social network site and our application.
- A set of client libraries (ASP.NET, C++, C#, PHP, Python) for different programming languages.

For our application, we used Linux Fedora Operating System Version 12.0, an open source JavaScript library [9] to draw the edges and nodes, Social Network API to gather necessary information to draw edges and nodes. PHP language is chosen as a server side technology to query database, run the clustering algorithms, and display the results on the social networking website. Our application can be embedded within a social networking website itself, or access a website's social data from anywhere on the Internet.

B. Algorithm Design

Our algorithm consists of two phases. In the first phase, we create the nodes for the users. In the second phase, we create and draw the connections between the nodes to determine the relationship and privacy between users who are registered on a social networking website.

The algorithm collects information such as friends' ids. This structure successfully detects social circles if the users choose to share the similar combination of personal information with friends in the same social circle, and if they choose different combinations with friends in other social circles. By using more data collected from our application, we have been finding out the effectiveness of our algorithm.

1) Creating Nodes

In this phase, we create all nodes of the graphs that we are going to draw. The algorithm for creating the nodes is shown in Fig. 3. In our node creation algorithm, we first go through all friends of the user and create nodes for each

 ISBN: 978-1-61208-100-7

friend. Then, for each friend, we go through all mutual friends and create nodes for each mutual friend. By saying mutual friends, we mean the common friends of the user with a user's friend.

```
for ( i = 0 ; i < Total_Friends ; i++)
{
Create_Node ( friend_ids [i] );
    for ( j = 0 ; j < Total_Mutual_Friends[i] ; j++)
    {
    Create_Node ( mutual_friends_id[i][j] );
    }
}
```

Figure 3. Node Creation Algorithm

2) Creating Edges

In this phase, we create the connections between friends of user and between mutual friends of user. By using the nodes that we created in the first phase of the algorithm, we add the edges according to the following algorithm shown in Fig. 4. In edge creation algorithm, we go through all friends of the user and find out if the friends are friends with each other. If they are friends, we add an edge between those friends. At the same time, we go through the mutual friends of the user and find out if they are friends with each other. If the mutual friends are friends with each other, we again add an edge between those mutual friends.

```
for ( i = 0 ; i < Total_Friends ; i++)
{
friend1 = friends_id[i];
    for ( j=0 ; j < Total_Friends[i] ; j++)
    {
    friend2 = friends_id[j];
        if ( friends.arefriends (friend1,friend2))
        {
        AddEdge(friend1,friend2);
        }
    }
    for ( j = 0 ; j < Total_Mutual_Friends[i] ; j++)
    {
    mutual_friend1 = mutual_friends_id[i][j];
    for ( k = 0 ; k < Total_Mutual_Friends[i] ; k++)
    {
    mutual_friend2 = mutual_friends_id[i][k];
        if(friends.arefriends(mutual_friend1,mutual_friend2)
        {
        AddEdge(mutual_friend1,mutual_friend2);
        }
    }
    }
}
```

Figure 4. Edge Creation Algorithm

Fig. 5 shows the output of node and edge creation algorithm. The colorful circles are the nodes that represent the social network user's friends, and the black lines are the edges that represent the friendship relation.

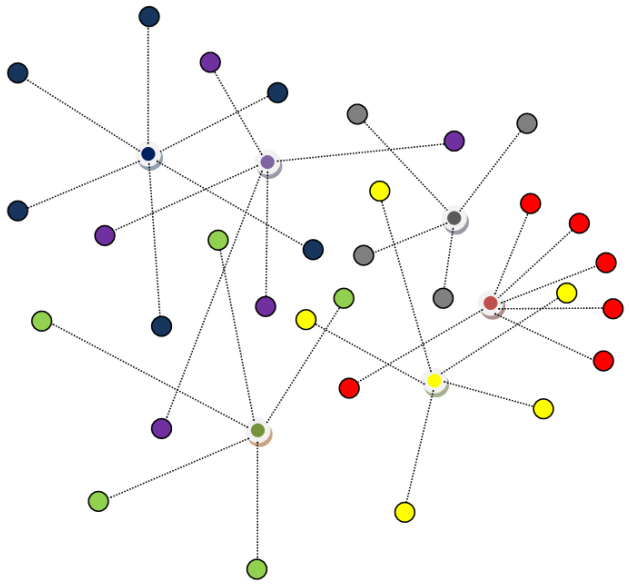

Figure 5. Output of Node and Edge Creation Algorithm

C. Clustering Method To Determine Social Circles

Methods for clustering have been deeply studied. But our aim is not to study them. Clustering is just one of the steps to achieve our privacy goal. Our main aim is to use the right clustering algorithm for social networks and develop an application to provide privacy by adapting this clustering algorithm to our application.

The clustering in social networks requires grouping users into classes based on their attributes, properties of personal relationship, web page links, spreads of messages and other applications. It is the process of organizing users into groups whose members are similar in some way. Our algorithm is different from other clustering algorithms, and it can dynamically group users in a social network into different classes based on their properties and effectively identify relations among classes. It collects some data which are similar between social network users and are dissimilar to the users belonging to other groups. It creates active cells like network grids and builds visual graph of social groups. The similar structure applied in the algorithm [8, 10] for finding (α, β) clusters has been used in our algorithm. Friends sharing common personal information are the adjacent nodes to α-fraction. The α-fraction represents the cluster that has a large density. On the other hand those friends not sharing common personal information are the adjacent nodes to β-fraction. The β-fraction represents the cluster that has a low density. It is therefore possible to use the social graph of network users as an input to our algorithm. One might ask that what if a friend belongs to more than one group? For example, a user can have a friend from high-school or university that is currently his/her work mate. The overlapping sets or being

 ISBN: 978-1-61208-100-7

in more than one group does not cause a problem from the privacy perspective. Our application groups friends according the common information that a user wants to share with his/her friends. For example, if we just want to share our photos and status with our college friends, then we will be showing them a profile where they will only able to see our photos and status. If there are some other friends that we just want to share our photos and status, they will also be in this group. Therefore, it is perfectly normal and possible that a person can be in one or more social circles. The user will show some information in one group and different information in another group. In other words, we limit who sees what. Fig. 6 shows our pseudo-algorithm of our clustering process for the users in a social network.

1. Write all answers of the users to DB
2. Select User's answers from DB and create result_array
3. FOREACH (result_array as value)
    3.1. Get selected friends' friendids for each question
    3.2. Create [question no, friendids] array
4. FOR i=0 to size (result_array)
    4.1. FOR k=i+1to size (result_array)
    4.2. Create the clustering_array[][]
5. Sort (clustering)
6. Create unique values for clustering_array[][]
7. Find how many times a friend is chosen in 10 questions
    7.1. Eliminate the friend: IF NOT a friend chosen >= 3 times
8. Find Min (set of information that the user wants to share)
9. Eliminate the sets: IF sets do not contain mutual friends
10. Display (Groups or circles)
11. Suggest (Privacy Settings)

Figure 6. Clustering Algorithm

D. Database Design

Building the visual graph of a social network user is an expensive task. Instead of creating the graph while executing the social network API calls, we decided to store the necessary information in our own database. The main reason to use our own database is because having too many API calls causes time outs. Another important reason is the difference between our database design and the social networking websites. Current social networking websites use relational databases to store social network data. For better performance, more effective querying, to extend our work and develop a knowledge based approach, we used graph database.

1) Graph Database Design

In graph based databases, information is stored as nodes, edges and properties. Since social networking data has similar properties, graph database is the powerful way of representing social relationships between people. In our application, we used Neo4j, an open source graph database. According to developers' of Neo4j [11], it is an embedded, disk-based, fully transactional Java persistence engine that stores data structured in graphs rather than in tables. More importantly, it includes the database features such as ACID transactions, durable persistence, concurrency control, transaction recovery, and other features of enterprise-strength databases. The following figures show our transition from relational database to Neo4j graph database. As it is seen in Fig. 7 and Fig. 8, it is very easy to see the connection between two people. However, in a relational database, it is hard to see who is friend with whom. In addition to this, whenever we introduce a relationship such as mutual-friends relationship, we need to add one more table to represent this relationship. As a result, the number of table joins increase and the performance decreases.

ID	Name	Sex	Age	Relationship Status
0001	Asim	Male	27	Single
0002	Erkan	Male	30	Single

User Table

ID1	ID2
0001	0002
0002	0001

Friendship Table

Figure 7. Our Social World Modeled in Relational Database

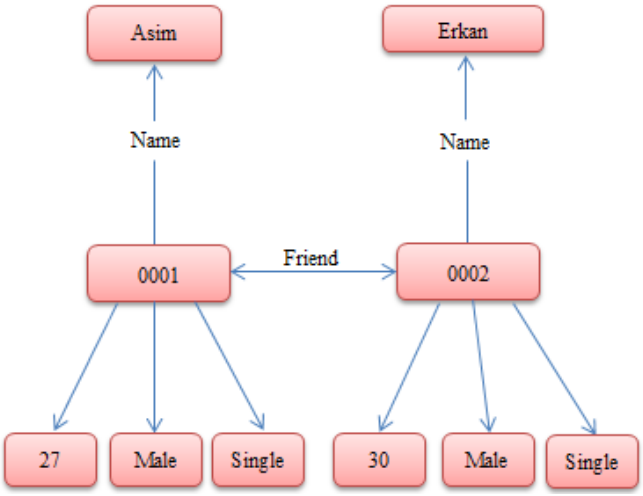

Figure 8: Our Social World Modeled in Neo4j Graph Database

2) Graph Database vs. Relational Database

Relational databases are around for many decades. They are the database choice of most traditional data-intensive storage and retrieval applications. SQL language is used to retreive the data. Relational databases are not efficient, if data contain many relations and require many joining of tables which are expensive operations. Thus, graph base database has better performance than relational database when representing relations. Recent studies by [12, 13] provide a detailed performance evalution of MySQL database and Neo4j. According to their results, graph

 ISBN: 978-1-61208-100-7

database is more flexible, easy to program, and perfoms better.

IV. CONCLUSION

Most of the related work present attacks for social networks and they do not provide a useful solution to protect privacy. We believe that our study is the first study that contains an implemented application for social network privacy. This system is a secure web application for social networking websites such as Facebook, Twitter, MySpace, and it includes an implementation of our original idea. Currently, it is running on Facebook. As a future work, we are planning to develop a general API that can be applicable for any social networking website such as Twitter and MySpace.

Recently, we have been inviting social network users to our study and collecting data. Furthermore, we are helping the users to get acquainted with our application. After collecting enough data, we will evaluate the effectiveness of our approach.

Our study uses a combination of clustering approaches. Firstly, the users are grouped according to their friendship relations (i.e., by using friendship and mutual friendship queries). Secondly, we group them based on the information that a user wants to share with his/her friends. The second one is the heart of grouping, since it will provide the privacy. Privacy is provided by showing different profiles to different combination of groups. For example; if a user wants to share his/her relationship status, photos, date of birth with his/her Friend-A and Friend-B, then Friend-A and Friend-B only see this information. Therefore, we are able to limit who sees what.

Although we successfully create the social graph of a user, we have limitations which affect the performance of our application. Our social graph visualization algorithm works for a subset of friends and mutual friends. We limited the number of friends and number of mutual friends that will participate in our study. The reason behind the limitation is because of large amount of social network API calls. There are millions of social network developers who are querying social network servers, and these queries cause a delay in response time. Drawing the social graph of a user and displaying it takes more time. Sometimes, the queries are even dropped because of the delay, and the graph is not drawn.

In this study, we proposed an application to identify the social circles of the users by using graph database system. In order to see the effectiveness of our algorithm we have been testing our application. As future work, we also want to develop a knowledge base system to provide intelligent decisions about sharing of personal information with people.

REFERENCES

[1] E. Hooper, “Intelligent strategies and techniques for effective cyber security, infrastructure protection and privacy” The 5th International Conference for Internet Technology and Secured Transactions (ICITST-2010), London, UK, 2009, pp. 1-7.

[2] A. Beach, M. Gartrell, and R. Han, “Solutions to Security and Privacy Issues in Mobile Social Networking”, The 12th IEEE International Conference on Computational Science and Engineering (CSE '09). Vancouver, Canada, 2009, pp. 1036 – 1042.

[3] B. Zhou, J. Pei, “Preserving Privacy in Social Networks Against Neighborhood Attacks”, The 24th International Conference on Data Engineering (ICDE’08), Cancún, México, 2008, pp. 506-515

[4] R. Gross, A. Acquisti, “Information Revelation and Privacy in Online Social Networks (The Facebook Case)”, ACM Workshop on Privacy in the Electronic Society, Alexandria, VA, USA, 2005, pp. 71-80

[5] J. Lindamood, R. Heatherly, M. Kantarcioglu, and B. Thuraisingham, “Inferring private information using social network data.”, The 18th International Conference on World Wide Web, Madrid, Spain, ACM 978-1-60558-487-4, 2009, pp. 1145-1146.

[6] E. Denham, “Report of Findings into the Complaint Filed by the Canadian Internet Policy and Public Interest Clinic (CIPPIC) against Facebook Inc. Under the Personal Information Protection and Electronic Documents Act”, http://www.priv.gc.ca/cf-dc/2009/2009_008_0716_e.cfm, Last accessed: July 18, 2010.

[7] E. Zheleva and L. Getoor, “To Join or Not to Join: The Illusion of Privacy in Social Networks with Mixed Public and Private User Profiles”, The 18th International World Wide Web Conference, Madrid, Spain, ACM 978-1-60558-487-4, 2009, pp. 531-540.

[8] F. Adu-Oppong, C. K. Gardiner, A. Kapadia, and P. P. Tsang, “Social Circles: Tackling Privacy in Social Networks (Poster Abstract)”, The 4th Symposium on Usable Privacy and Security (SOUPS), Pittsburgh, PA, USA, 2008.

[9] K. Scholz, “Using Force Directed Graphs in Your App.” http://www.kylescholz.com/blog/2006/06/using_force_directed_graphs.html, Last accessed: July 18, 2010.

[10] N. Mishra, R. Schreiber, I. Stanton, and R. Tarjan, "Clustering Social Networks", The 5th International Conference on Algorithms and Models for the Web-Graph, San Diego, CA, USA, 2007, pp. 56-67.

[11] Neo4j, http://neo4j.org/, Last accessed: July 18, 2010.

[12] M. A. Rodrigez, “MySQL vs. Neo4j on a Large-Scale Graph Traversal.”,http://markorodriguez.com/Blarko/Entries/2010/3/29_MySQL_vs._Neo4j_on_a_Large-Scale_Graph_Traversal.html, Last accessed: July 18, 2010.

[13] C. Vicknair, M. Macias, Z. Zhao, X. Nan, Y. Chen, and D. Wilkins, “A Comparison of a Graph Database and a Relational Database”, The 48th ACM Southeast Conference, Oxford, Mississipi, USA, 2010.

 ISBN: 978-1-61208-100-7