International Journal of Scientific Engineering and Science
Volume 2, Issue 10, pp. 27-30, 2018.
ISSN (Online): 2456-7361

Constrain Based Image Sharing in Timeline

M. A. Gunavathie[1], J. Hemavathy[2]
[1, 2]Assistant Professor, Department of Information Technology, Panimalar Engineering College, Chennai, India-600123

***Abstract*—*Sharing images within online content sharing sites may lead to unwanted disclosure and privacy violations. Further, the persistent nature of online media makes it possible for other users to collect rich aggregated information about the owner of the published content and the subjects in the published Content. We propose a chain graph algorithm system which aims to share data in a secured way among group of people. It includes own privacy setting frame by the user who uploads the data. Where we can avoid overflow of information and to maintain better service to the user. Time limit based image sharing allows user to keep their data storage in a light manner. A graph will be generated by comparing the number members among the group.*

***Keywords*—** *Own privacy setting frame, Time limit based image sharing.*

I. Introduction

Images are one of the key identifiers for the users in social networking sites. Sharing takes place through groups or circles inside the social networks and also between the individuals in the network. Some images or content reveals the sensitive information about the users or information about circle among them. This enables the untrusted third party user to collect such information and make them to use it in illegal way [2], [12], [13]. Further sharing images on social networks may lead to unwanted disclosure or privacy violations [2], [12]. All the social networking sites allow the users to set the privacy preferences for their profile information but recent studies have shown that users are unable to setup and maintain such privacy settings [1], [4], [7], [8], [11]. The existing proposals for automating privacy setting appear to be inadequate to address the unique privacy setting for images [2], [5], [9]. In this paper we propose a chain graph system which aims Precise skeleton structure of calligraphic characters to provide users to set privacy for the individual content they have uploaded

II. Related Work

Our work is related on privacy setting configuration in social sites and privacy analysis of online images. Several works have studied to automate the task of privacy settings (e.g.[4], [6], [7], [10], [13], [14]).Ralph Gross. [1] proposed concept of Information Revelation and Privacy in Online Social Networks which study the patterns of information sharing in social networks and their privacy implications. But it will not able to overcome the advanced search preference option in social networks in case of providing privacy. Shane Ahern. [3] proposed Privacy Patterns and Considerations in Online and Mobile Photo Sharing which aims to use context-aware camera phones as capture devices to explore patterns of privacy. The main draw back in this concept is some users remain unconcerned about privacy settings and also all the users not able to use such devices for privacy. Hong-Ming Chen. [5] proposed Group and Tag Recommendation Flickr Photos by Automatic Search-based learning which allows to automatically add photos into appropriate groups and recommend suitable tags for users on social sites. This approach focus on sharing rather than protecting the content. The unprotected shared content makes the untrusted third party to reveal the users valuable information. Morgan Ames [3] proposed Motivations for Annotation in Mobile and Online Media motivates users to annotate content. Point-of-capture annotation (*e.g., on the mobile device)* can encourage the addition of tags. In some cases, the suggestions can inspire users to tag their photos and give them guidance for how best to annotate. The main problem is additional data from more ZoneTag users remain for a longer period of time which may lead to overflow of memory on online sites.

III. Existing System

The online networking sites allocates particular TB of memory to individual users. If the usage exceeds that limit the admin of particular site will remove the older post from the user database. This is to maintain the performance of the site. For this to happen the sites have all rights they will not need to get permission from the particular user. Their privacy policy setting ensures this. Contents are removed from the users profile without permission. Due to this some valuable information, contents or images may deleted from the user profile. This concept is same in case of posting comments also. Some posts may get more number of comments. When we search for previous comments it will not be displayed properly, automatically it will be removed. Regarding privacy on uploaded images the mutual friends or friends of friends may download the image and they may use it illegally. They may also tag or share it to another friends. The owner of the image will not able to set privacy for their individual posts. The present system only allows users to set privacy like friends, friends of friends and only me options. Sharing images on group also not trustworthy. Present options of privacy settings to upload images is not applicable for groups or circles. An individual user in a particular social networking site may be in different groups. The present system is not able to compare the number of members among different group.

IV. Proposed System

By considering the privacy policy of social networking sites we propose a system to set the privacy preference for the uploaded contents by the user itself. While posting images the owner of the content is able to set the date for the content. For example. If Rahim is going to celebrate his birthday by tomorrow. He want to invite his friends. He post a birthday party invitation today and he is able to set that invitation to be displayed in that site for particular period. After that period that post will be automatically removed. By setting this options in the uploaded content, the unwanted information are automatically removed and the user will able to maintain the TB allocated to them. Some valuable information will remain in the sites. We can retrieve it later. While uploading images the owner will able to set privileges for the friends who are going to be tagged in that photo. They can set tag, command, download and all privileges options for each and every friends. So that they are not able perform operations other than the provided privileges. Even they can't able to use print screen button they are not provided with download privilege. The posted comment will remain for the fixed period of time. Positing contents or images on the groups also provided with same privileges. A graph will be generated by comparing the number members among the group.

V. MODULES

There are five important modules in this system and they are described below.

- Contact & Group
- Image Sharer Module (Set Permissions)
- Image Filter Module
- Admin
- Graph Module

a. Contact & Group:

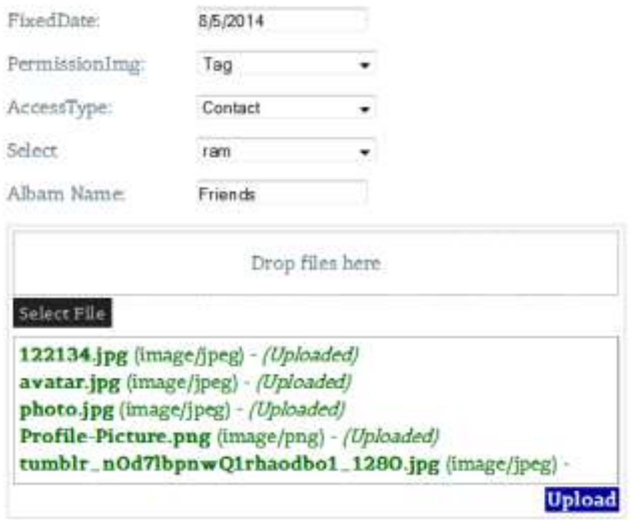
In this module user must register in this application to get a Membership. Once user known the membership id they can able to login our application and also they can able to use our application like Social Network. User can find friends and send request to the particular user for the joining in the account. Once user accepts the request after that only they can able to add in contact. There are a several members in contact to differentiate each user we will create a group for sharing information.

b. Image & Album Sharer Module:

In Image sharer module mainly focusing the secure image transfer with the friends as per poster willingness. Once user had chosen the image later that will set privacy like tag, share, download etc. and Share the image to their friends even a single person (or) in group. They can able to create album and Share also with the same manner.

Tag
Contact

Fig. 1. Image Sharer Module.



FixedDate:	8/5/2014
PermissionImg:	Tag
AccessType:	Contact
Select	ram
Albam Name:	Friends

122134.jpg (image/jpeg) - (Uploaded)
avatar.jpg (image/jpeg) - (Uploaded)
photo.jpg (image/jpeg) - (Uploaded)
Profile-Picture.png (image/png) - (Uploaded)
tumblr_nOd7lbpnwQ1rhaodbo1_1280.jpg (image/jpeg) -

Fig. 2. Album Sharer Module.

c. Image Filter Module:

Once user shared the post to their friends like Wedding Invitation, Birthday invitation. It will show to the particular friend's up to the particular period which indicates date in the invitation. Even though one of the friend made access unfriend of us it will reflect in notification and download the image but in proposed we had overcome the download access.

Fig. 3. Image Filter Module.

d. Admin Module:

In admin module, admin will able to monitor the user and their post. Once Post had been made by the user unwanted activity admin will have a rights to block the particular user in the application.

e. Graph Module & Notification module:

In graph based view, User will have a multiple group in the application to know the strength of the group we can able to monitor it. A graph will be generated by comparing the number members among the group. At the same time we have a notification alert for their friends of any one made the comment the post of their friend.

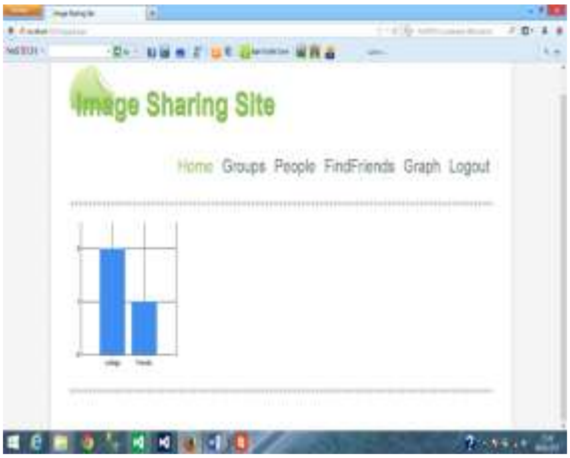


Fig. 4. Graph Module

VI. Data Flow Diagram

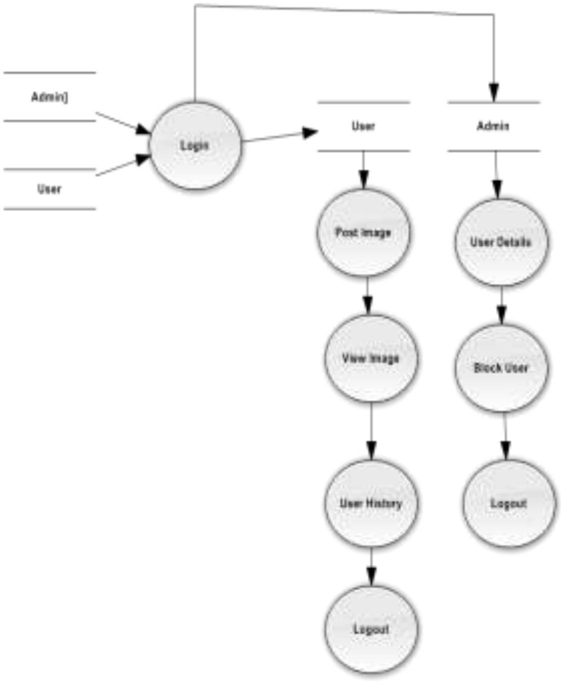


Fig. 5. Data Flow Diagram

VII. Algorithm

Cluster Algorithm:

1. Write all the 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+1 to 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)

7.1 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. In our node creation algorithm, we first go through all friends of the user and create nodes for each friend. Then, for each friend, we go through all mutual friends and create nodes for each mutual friend.

7.2 Creating Edges:

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] );
}
}
```

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. 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);
}
}
}
}
```

VIII. REQUIREMENTS

8.1 Software Requirements:

OS	:	Windows 2007
Front End	:	Microsoft Visual Studio 2012
Back End	:	SQL Server 2012
Language	:	C#.Net

8.2 Hardware Requirements:

Processor	:	Pentium IV
Hard Disk	:	40GB
RAM	:	2GB or more
Monitor	:	LED Monitor

TABLE 1. Work Flow of Process

Stage	Description
Page request	The page request occurs before the page life cycle begins. When the page is requested by a user, ASP.NET determines whether the page needs to be parsed and compiled (therefore beginning the life of a page), or whether a cached version of the page can be sent in response without running the page
Start	In the start step, page properties such as Request and Response are set. At this stage, the page also determines whether the request is a postback or a new request and sets the IsPostBack property. Additionally, during the start step, the page's UICulture property is set.
Page initialization	During page initialization, controls on the page are available and each control's UniqueID property is set. Any themes are also applied to the page. If the current request is a postback, the postback data has not yet been loaded and control property values have not been restored to the values from view state.
Load	During load, if the current request is a postback, control properties are loaded with information recovered from view state and control state.
Validation	During validation, the Validate method of all validator controls is called, which sets the IsValid property of individual validator controls and of the page.
Postback event handling	If the request is a postback, any event handlers are called.
Rendering	Before rendering, view state is saved for the page and all controls. During the rendering phase, the page calls the Rendermethod for each control, providing a text writer that writes its output to the OutputStreamof the page's Response property.
Unload	Unload is called after the page has been fully rendered, sent to the client, and is ready to be discarded. At this point, page properties such as Response and Request are unloaded and any cleanup is performed.

IX. CONCLUSION

We have proposed a privacy setting system which makes users information safe and secure in social networking sites. Individual privilege setting prevents the illegal usage of posted contents. Time limit based uploads improves performance of the site by removing unwanted information. Automatic graph generation enables us to identify the strength of the group. Our experimental study proves that chain graph system is a practical tool that offers significant improvements over current approaches to privacy. We are hoping to keep investigating the topic to get a more detailed look of patterns across a longer time period, and perhaps in different cultures.

REFERENCES

[1] A. Acquisti and R. Gross. Imagined communities: Awareness, information sharing, and privacy on the facebook. In Privacy Enhancing Technologies Workshop, 2006.

[2] S. Ahern, D. Eckles, N. S. Good, S. King, M. Naaman, and R. Nair. Over-exposed?: privacy patterns and considerations in online and mobile photo sharing. In Conference on Human factors in computing systems, pages 357–366. ACM, 2007.

[3] M. Ames and M. Naaman. Why we tag: motivations for annotation in mobile and online media. In Conference on Human factors in computing systems, CHI' 07, pages 971–980. ACM, 2007.

[4] H. Lipford, A. Besmer, and J. Watson. Understanding privacy settings in facebook with an audience view. In Proceedings of the Conference on Usability, Psychology, and Security, 2008.

[5] H.-M. Chen, M.-H. Chang, P.-C. Chang, M.-C. Tien, W. H. Hsu, and J.-L. Wu. Sheepdog: group and tag recommendation for flickr photos by automatic search-based learning. In 16th ACM international conference on Multimedia, pages 737–740. ACM, 2008.

[6] A. K. Fabeah Adu-Oppong, Casey Gardiner and P. Tsang. Social circles: Tackling privacy in social networks. In Symposium On Usable Privacy and Security, 2008.

[7] K. Strater and H. Lipford. Strategies and struggles with privacy in an online social networking community. In British Computer Society Conference on Human-Computer Interaction, 2008.

[8] L. Church, J. Anderson, J. Bonneau, and F. Stajano. Privacy stories: Confidence on privacy behaviors through end user programming. In Symposium on Usable Privacy and Security (SOUPS), 2009.

[9] C. A. Yeung, L. Kagal, N. Gibbins, and N. Shadbolt. Providing access control to online photo albums based on tags and linked data. In Social Semantic Web: Where Web 2.0 Meets Web 3.0 at the AAAI 2009 Symposium, 2009.

[10] E. M. Maximilien, T. Grandison, T. Sun, D. Richardson, S. Guo, and K. Liu. Privacy-as-a-service: Models, algorithms, and results on the Facebook platform. Web 2.0 Security and Privacy workshop, 2009.

[11] D. Liu, X.-S. Hua, M. Wang, and H.-J. Zhang. Retagging social images based on visual and semantic consistency. In 19th interna- tional conference on World wide web, WWW '10, pages 1149–1150. ACM, 2010.

[12] Y. Liu, K. P. Gummadi, B. Krishnamurthy, and A. Mislove. Analyz- ing facebook privacy settings: user expectations vs. reality. In 2011 ACM SIGCOMM conference on Internet measurement conference, IMC '11, pages 61–70. ACM, 2011.

[13] P. Klemperer, Y. Liang, M. Mazurek, M. Sleeper, B. Ur, L. Bauer, L. F. Cranor, N. Gupta, and M. Reiter. Tag, you can see it!: using tags for access control in photo sharing. In ACM annual conference on Human Factors in Computing Systems, CHI '12, pages 377–386. ACM, 2012.

[14] A. Mazzia, K. LeFevre, and A. E. The pviz comprehension tool for social network privacy settings. In Symposium on Usable Privacy and Security (SOUPS), 2012.