How do Facebookers use Friendlists

Yousra Javed, Mohamed Shehab
Department of Software and Information Systems
University of North Carolina at Charlotte
Charlotte, NC 28233, USA
Email: {yjaved,mshehab}@uncc.edu

***Abstract*—Facebook friendlists are used to classify friends into groups and assist users in controlling access to their information. In this paper, we study the effectiveness of Facebook friendlists from two aspects: *Friend Management* and *Policy Patterns* by examining how users build friendlists and to what extent they use them in their policy templates. We have collected real Facebook profile information and photo privacy policies of 222 participants, through their consent in our Facebook survey application posted on Mechanical Turk. Our data analysis shows that users' customized friendlists are less frequently created and have fewer overlaps as compared to Facebook created friendlists. Also, users do not place all of their friends into lists. Moreover, friends in more than one friendlists have higher values of node betweenness and outgoing to incoming edge ratio values among all the friends of a particular user. Last but not the least, friendlist and user based exceptions are less frequently used in policies as compared to allowing all friends, friends of friends and everyone to view photos.**

***Index Terms*—Policy, Access Control, Grouping, Privacy, Social Network**

I. INTRODUCTION

Facebook (FB) is the largest online social network used today with over 900 million active users. The friendlist (FL) feature was introduced in 2007, in order to help FB users in organizing a large friend network into groups [2]. The FB privacy controls allow users to use the FLs for customizing their sharing preferences. Recently, FB improved the FL feature by standardizing lists into the following three categories:

• *Close Friends:* The user can put his top priority friends in this list. Mostly, these are the friends with whom the user interacts the most.

• *Acquaintances:* The friends in this list are the ones that user keeps with a mute button pressed. Their updates hardly appear in the homepage news feed.

• *Smart Lists:* These are lists that appear with lightening icon and are automatically created and populated for each new workplace, city or school that the user adds to his profile.

People can now use the FB standardized lists in addition to their own customized lists. But, research shows that creating FLs is a secondary task completed by only a few FB users. An average FB user has about 120 friends [9], which makes the process of friend grouping quite tedious. Most users are either not aware of the FL feature or consider managing lists a difficult task which requires remembering who is in which list [12]. In a small thesis study, it was found that none of the 10 study participants used FLs to control their privacy settings [13]. Strater et al. [14] studied 18 FB users and found that 72% of them either made their profile completely public or restricted it to their friends only. Only 5 users used customized settings to incorporate FLs. These studies lead to the following questions about the effectiveness of the current FL feature on FB:

• Do users use the FL feature frequently?
• How do users build FLs?
• How are FLs used to compose policy patterns?

In this paper, we try to answer the above questions by studying *real user profile data* and *photo privacy policies*. We collected the policy data of 222 Facebookers through our FB survey application distributed as Human Intelligent Task (HIT) on Mechanical Turks. Using this data, we analyze the effectiveness of FLs from two aspects: 1)Friend management, by finding out the number and size of FB and user created lists, percentage of users who do not fall in any lists, overlaps between FLs and their comparison with various network metrics and 2)Usage in policy patterns for setting exceptions.

The rest of the paper is organized as follows: In Section II, we discuss the related work on studying FB FLs. Section III explains how we collected the user data for our analysis. Section IV details the various statistics and metrics that we have used in our analysis. Section V presents and discusses the results. Finally, we wrap up the paper with our conclusions.

II. RELATED WORK

Kelley et al. [7] have done preliminary work towards investigating how users create friend groups in FB. They have examined four different methods of friend grouping and their results show that the type of mechanism used, affects the groups created. Their study shows that 30% of the users had FLs out of which 40% did not use them to control privacy settings. Those who had FLs never updated them.

Recently, researchers have developed tools in order to assist a user in grouping his/her friends efficiently and simultaneously enabling him to create better privacy policies and introduce exceptions. Adu-Oppong et al. [1] have proposed partitioning a user's friends into lists based on communities extracted automatically from the network, as a way to simplify the specification of privacy policies. Jones and O'Neill [6] created an automatic method to group FB users' friends using the SCAN clustering algorithm and compared these groups against the user created groups, achieving 70% accuracy. Mazzia et al. [11] built a policy visualization tool that extracts and presents the user's communities to help him in managing his group based privacy policies. Egelman et al. [4] built a Venn diagram based interface to cope up with semantic errors that the current users make in their access control settings resulting in over-sharing or under-sharing of information.

III. DATA COLLECTION

In order to collect real profile data and privacy policies, we developed a FB survey application using the FB APIs. The survey comprised of questions for gathering users' privacy concerns. The user consent was requested and users were informed of the collected data. To recruit participants, we published our survey application as an HIT on Amazon Mechanical Turk. Amazon Mechanical Turk is a crowd sourcing marketplace that pairs Requesters of work and Workers. Requesters formulate work into HITs which are individual tasks that workers complete. A total of 222 participants' profile and privacy policy data was collected. 173 out of these were male and 49 were females. 57% of them had ages in the range 15-25, 37% in the range 25-35 and 6% in the range 35-70. 17.5% had grad school as their highest education. 10% had college and 62.6% had high school as their highest education.

IV. DATA ANALYSIS

In this section, we describe the various metrics that were used to analyze the collected data.

A. *Friend Management*

In rest of the paper, we refer to FB's standardized list categories i.e., acquaintances, close friends, work, education and current city etc as *FB created friendlists (FCFLs)* and the user's customized FLs as *User created friendlists (UCFLs)*. For example, Alice has 200 friends. She creates three FLs with names *Family and relatives*, *High school friends* and *Undergrad friends*. After she updates her work with WellsFargo, FB automatically creates a list with the name of this organization and populates it. Therefore, the three FLs that she created herself fall into *UCFLs* while WellsFargo falls into *FCFLs*. To study how well users manage their friends through FLs, we set out to extract the following statistics and determine how users build their FLs:

1) FL creation statistics

• ***Frequency w.r.t FL type:*** We calculate the average number of FCFLs and UCFLs.

• ***FL size:*** We measure the average list size.

• ***Friend coverage:*** This metric specifies how many of the user's friends fall in at least one FL and is defined as $\frac{No.\ of\ friends\ in\ FLs}{Total\ No.\ of\ friends}$.

2) Grouping strategy statistics

→ ***Friends in more than one FLs:*** Sometimes, a user wants some of his/her friends in one list to have access to additional information as compared to the other members in this list. This can be done by placing this friend in another list which has access to this additional information, resulting in FL overlap. FL overlap is therefore, an interesting metric to investigate the number of roles a friend can have and to understand what type of friends fall in more than one lists.

In order to find out specific features of these common friends, we measure two network metrics:

→ ***Node Betweenness:*** This metric measures how central a node is to the network by calculating the number of shortest paths in the network that contain this node [10].

→ ***Node outgoing to incoming edge ratio:*** Node outgoing to incoming edge ratio is another metric to analyze the type of friends that belong to more than one FLs. This is calculated with respect to the cluster in which a friend lies. We use the Clauset Newman Moore (CNM) network clustering algorithm [3] to cluster a network and then find out the ratio between the number of connections of a node with nodes in other clusters and the number of connections that this node has with other nodes in its cluster.

• ***Comparison of FLs with CNM clustering:*** We study the correlation between UCFLs and FCFLs *and* the clusters generated by CNM network clustering algorithm [3]. For this purpose, we use the Adjusted Rand Index to measure the alignment between CNM clusters and user defined FLs [5]. The Adjusted Rand Index compares the predicted labels (CNM clusters) with the actual labels (user defined FLs) and produces an index between 0 and 1. To cater for the FL label of friends falling in more than one UCFL and FCFL, we picked a random label for our comparison.

B. *Policy Patterns*

A policy pattern represents a combination of allow and deny rules for access to information by friends. It can range from being public *to* moderately private i.e., allowing all friends or denying specific FLs *to* extremely private i.e., allowing or denying specific users only. The collected privacy policies and FL membership information was used to extract the policy patterns. We divide these policy patterns into two categories :

• ***Custom:*** These consist of FL and user based exceptions on FB objects. E.g., Allowing only Family FL and Bob.

• ***Default:*** These are the default options provided by FB. E.g., Allowing only Friends, Friends of friends or only me.

We analyze these policy patterns with respect to Westin′s privacy index based subject classification [8] and investigate the frequency of FL usage in policies by the three classes of users.

V. Results and Discussion

This section discusses the results of our analysis for each of the metrics described in the previous section.

A. *Friend Management*

Figures 1(a) and 1(b) show the number of FCFLs and UCFLs, and the percentage of users who created them. It can be seen from the figure, that the UCFLs are less frequent as compared to the FCFLs. This is indicated in Figure 1(a) and 1(b), as the percentage of users who did not create any UCFL is a lot higher than the percentage of users who did not create any FCFL. The average number of members in FCFLS and UCFLs was found out to be 45 and 32 respectively whereas the average number of UCFLs and FCFLs was equal to 0 and 4 respectively. Figure 1(d) shows the coverage of friends in the FLs. It is to be noted that a very small number of users (coverage value equal to 1) put all their friends into FLs.

Figure 1(c) represents the FL overlap by showing the number of FLs in which friends fall. The FCFLs overlap more as compared to the UCFLs. The figure indicates that the users do not place a lot of friends in more than two UCFLs. On the other hand, FCFLs, specifically smart lists are automatically created and populated based on the information in the user's profile which can be an explanation of why these lists have higher overlaps.

Using node betweenness, we have calculated that what percentage of friends in more than one FLs have the highest node betweenness values and outgoing to incoming edge ratios among all the friends in a users network. Figure 2 shows the percentage of friends in 2,3,4 and 5 FLs that have the top betweenness values. Overall, the trend shows that the percentage of these friends decreases as the betweenness values decrease, i.e., the betweenness values for a higher proportion of these friends lie in the top 10%. This implies that the friends with higher betweenness values or more connections have higher probability of being placed in more than one FLs.

Similarly, the friends in more than one FLs were ranked according to their outgoing to incoming edge ratio values. Figure 3 shows the percentage of friends in 2,3,4 and 5 FLs against the ratio values they have. The curves are not strictly similar to that for node betweenness i.e., the highest percentage of these friends do not have the ratio values in the top 10%, still, overall they follow the same trend; the curves are lying on the left side.

Figure 6 shows the statistics of adjusted rand index for UCFLs and FCFLs. The index values are comparatively higher for FCFLs as compared to the UCFLs.

B. *Policy Patterns*

We observed that the users in our study set the following access control policies over their photo albums:

• ***Custom***

1) *Allow* some friendlists and *deny* none
2) *Allow* some friendlists and *deny* some friends
3) *Allow* some friendlists and *deny* some friendlists
4) *Allow* all friends and *deny* some friends
5) *Allow* all friends and *deny* some friendlists
6) *Allow* some friends and *deny* none
7) *Allow* some friends and *deny* some friends

• ***Default***

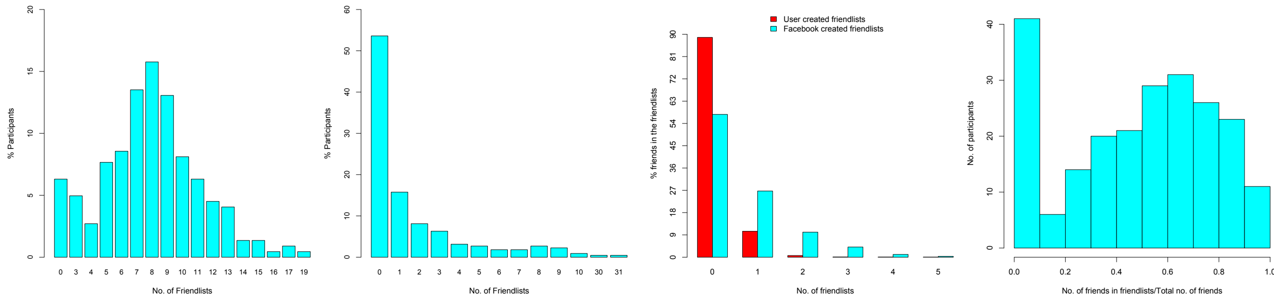
1) *Allow* me only and *deny* none
2) *Allow* everyone (Public) and *deny* none
3) *Allow* friends only and *deny* none
4) *Allow* friends of friends (FoF) and *deny* none
5) *Allow* friends and networks and *deny* none

To investigate the policy templates created by our study participants, we classified the users according to Westin's privacy index [8]. Figures 4 and 5 display the frequency of use of each policy pattern with respect to unconcerned, pragmatists and fundamentalists who were 5%,70.3% and 24.7% of the participant population respectively. Majority of the patterns were used less than 5 times. FL and user based exceptions were less frequently used in policies and users mostly allowed all friends, friends of friends or even everyone to view their photo albums.

The unconcerned participants used the least number of FL and user based exceptions in their policies and had more open policies. Pragmatists and Fundamentalists behaviors on the other hand, did not follow any specific trend and used almost all the policy patterns in both the categories. However, the percentage of users using default policies provided by FB was greater than those using FL based exceptions.

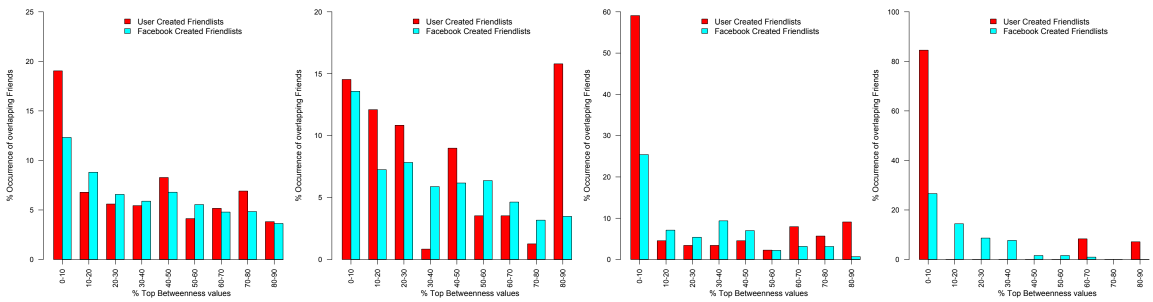
An important conclusion from the results in Figures 4 and 5 is that users do not frequently build custom policies that involve allowing and denying combinations of FLs and individual friends. They tend to either use the allow field or the deny field. Allow field is usually populated when a user wants to give access to specific lists or friends. In that case, he does not use the *Hide this from* field. Deny field, on the other hand, is mostly used in combination with allowing all friends, because, the user wants all his friends to view the specific information except some specific people. Therefore, in this case, he leaves the allow field empty.

We have also looked at how large the exception lists in the user policies are, by calculating the number of individual users and FLs in the allow and deny exceptions. Figure 4(h) shows the average exception list size for each of the three classes of users. The small allow list size of pragmatists and the small deny list size of the fundamentalists suggests that these users



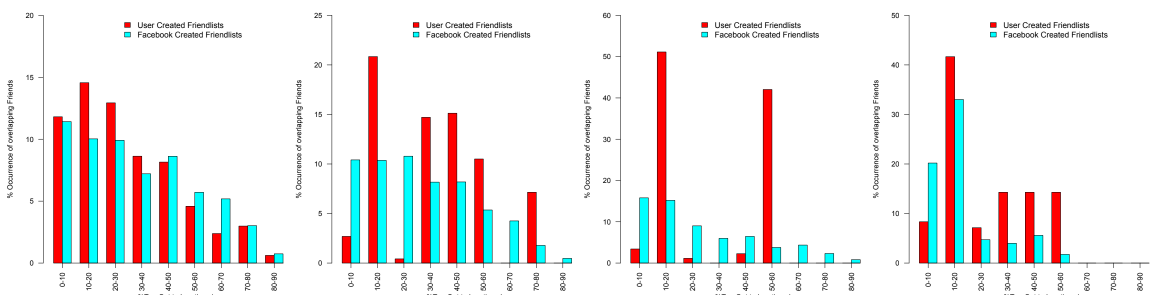
(a) No. of FCFLs (b) No. of UCFLs (c) Overlap in Friendlists (d) Friend coverage in friendlists

Fig. 1. Friendlist creation statistics



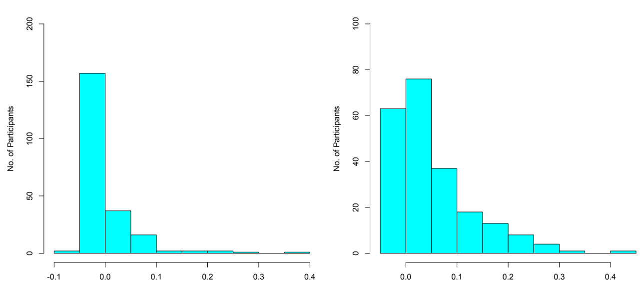
(a) Two Friendlists (b) Three Friendlists (c) Four Friendlists (d) Five Friendlists

Fig. 2. Node betweenness distribution of friends in more than one friendlist



(a) Two Friendlists (b) Three Friendlists (c) Four Friendlists (d) Five Friendlists

Fig. 3. Outgoing to incoming edge ratio distribution of friends in more than one friendlist



(a) UCFLs (b) FCFLs

Fig. 6. Comparison of Friendlists with CNM clustering using Adjusted Rand Index

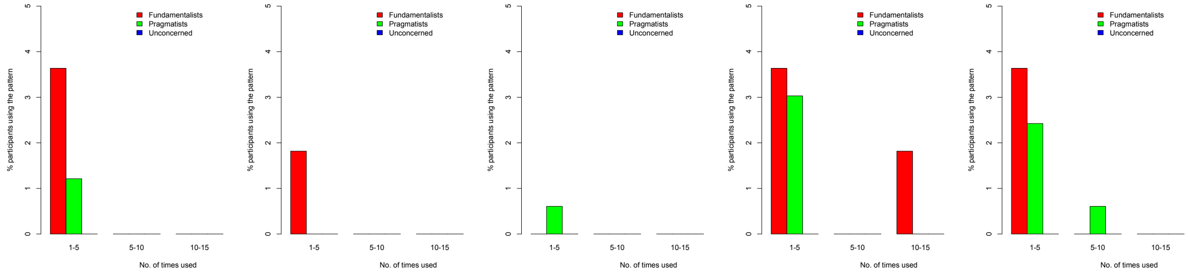
place all the to-be allowed or to-be denied friends in one list in order to assist them in managing access control.

VI. Conclusion

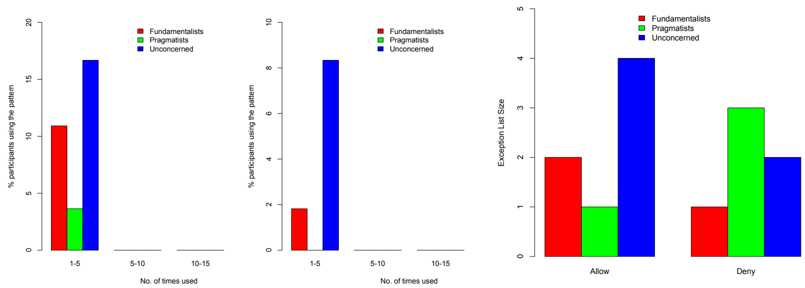
In this paper, we have studied Facebook friendlists through the collection and analysis of Facebook users's real profile information and photo privacy policies. The effectiveness of friendlist feature was analyzed from two aspects: 1) Organizing friends and 2) Setting exceptions in policies.

Our statistics regarding friend management show that: 1)Users own customized friendlists are less frequently created and have fewer overlaps as compared to Facebook created friendlists; and 2)Users do not place all their friends in lists; and 3)Friends in more than one friendlists have higher values of node betweenness and outgoing to incoming edge ratio values among all the friends of a particular user.

Our analysis of the 12 extracted policy patterns with respect to

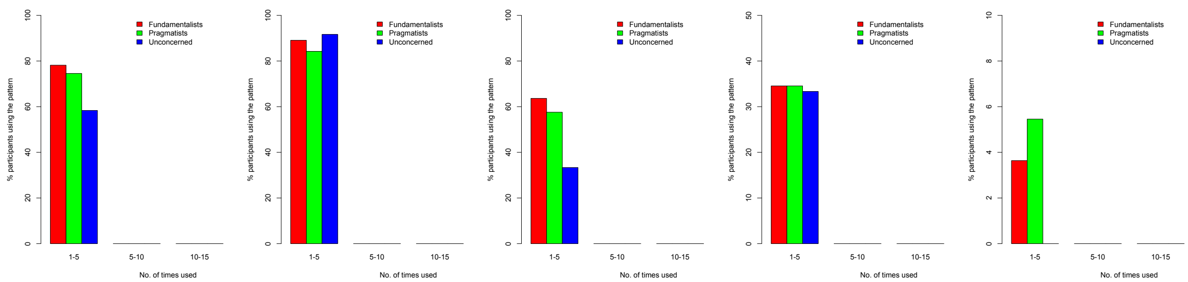


(a) Pattern 1 (b) Pattern 2 (c) Pattern 3 (d) Pattern 4 (e) Pattern 5



(f) Pattern 6 (g) Pattern 7 (h) User Exception list Size

Fig. 4. Frequency of use of custom policy patterns on Facebook and exception list size



(a) Pattern1 (b) Pattern 2 (c) Pattern 3 (d) Pattern 4 (e) Pattern 5

Fig. 5. Frequency of use of default policy patterns on Facebook

Westin's subject classification criteria shows that only few of the created friendlists are used in policies. Also, the friendlist and user based exceptions are less frequently used in policies as compared to allowing all friends, friends of friends and everyone to view photos.

REFERENCES

[1] F. Adu-Oppong, C. K. Gardiner, A. Kapadia, and P. P. Tsang. Social circles: Tackling privacy in social networks. In *SOUPS*, 2008.

[2] A. Agarwal. Facebook friendlist feature. http://www.labnol.org/internet/favorites/facebook-friends-groups/1956/, Dec 2007.

[3] A. Clauset, M. E. J. Newman, and C. Moore. Finding community structure in very large networks. *Physical Review E*, pages 1– 6, 2004.

[4] S. Egelman, A. Oate, and S. Krishnamurthi. Oops, i did it again: Mitigating repeated access control errors on facebook. In *ACM SIGCHI*, pages 2295–2304, 2011.

[5] L. Hubert and P. Arabie. Comparing partitions. *Journal of classification*, 2(1):193–218, 1985.

[6] S. Jones and E. O'Neill. Feasibility of structural network clustering for group-based privacy control in social networks. In *SOUPS*, 2010.

[7] P. G. Kelley, R. Brewer, Y. Mayer, L. F. Cranor, and N. Sadeh. An investigation into facebook friend grouping. In *INTERACT*, 2011.

[8] P. Kumaraguru and L. F. Cranor. Privacy indexes: A survey of westin's studies. *ISRI Tech. Report*, 2005.

[9] C. Marlow. Facebook statistcs. https://www.facebook.com/note.php?note_id=55257228858/, March 2009.

[10] C. Marlow. Node betweenness. http://en.wikipedia.org/wiki/Centrality,.

[11] A. Mazzia, K. LeFevre, and E. Adar. The PViz Comprehension Tool for Social Network Privacy Settings. Technical Report CSE-TR-570-11, University of Michigan, April 2011.

[12] M. M. Skeel and J. Grudin. When social networks cross boundaries: A case study of workplace use of facebook and linkedin. In *ACM GROUP*, pages 95–104, 2009.

[13] D. Smart. How do people limit their behaviour in response to their perception of potential social tension? a study of facebook. Technical report, Manchester Business School, May 2009.

[14] K. Strater and H. R. Lipford. Strategies and struggles with privacy in an online social networking community. In *Proceedings of the 22nd British HCI Group Annual Conference on People and Computers: Culture, Creativity, Interaction - Volume 1*, BCS-HCI '08, pages 111–119, Swinton, UK, UK, 2008. British Computer Society.