Access Control Policy Misconfiguration Detection in Online Social Networks

Yousra Javed
College of Computing and Informatics
University of North Carolina at Charlotte
Charlotte, NC 28223, USA
Email: yjaved@uncc.edu

Mohamed Shehab
College of Computing and Informatics
University of North Carolina at Charlotte
Charlotte, NC 28223, USA
Email: mshehab@uncc.edu

ABSTRACT

The ability to stay connected with friends online and share information, has accounted for the popularity of online social networking websites. However, the overwhelming task of access control policy management for information shared on these websites has resulted in various mental models of sharing with a false sense of privacy. The misalignment between a user's intended and actual privacy settings causes access control misconfigurations, raising the risk of unintentional privacy leaks. In this paper, we propose a scheme to extract the user's mental model of sharing, and enhance this model using information learned from their existing policies, in order to enable them to compose misconfiguration free policies. We present the possible misconfiguration patterns based on which we scan the Facebook user's access control policies. We implemented a prototype Facebook application of our scheme and conducted a pilot study using Amazon Mechanical Turk. Our preliminary results show that the users' intended policies were significantly different than their actual policies. Our scheme was able to detect the misconfiguration patterns in album policies. However, the reduction in the number of misconfigurations after using our approach was not significant. Participants' perceptions of our proposed policy misconfiguration patterns and the usability of our scheme was positive.

I INTRODUCTION

Online social networks have attracted a large user base over the recent years. Due to the vast amount of information being shared on these websites daily, effective data privacy management by users is a major concern on these websites [1]. To enable the customization of access control policies on user data, most social networks provide a privacy settings interface to manage the privacy of various profile items [2]. An *access control policy* represents the permissions set by a user to allow or deny access to a particular item. End-users are inexperienced in authoring access control policies, and therefore, struggle to express and maintain fine-grained policies for different data items [3] [4] [5] [6] [7]. Recently, tools to improve the usability of current interface *and* aid the users in understanding how their information is visible to their friends on Facebook style social networking websites, have been proposed [8] [9] [10].

The biggest challenge in effective data privacy management on social networks is how to effectively capture a user's sharing mental model, and how to enhance this mental model in order to reduce access control policy misconfigurations [11]. For example, Alice intends to allow access to her Facebook albums to only those friends who studied with her in college *and* are close to her. Therefore, she includes the two friend lists, namely, *Close friends* and *College friends* in her access control policy. This gives Alice a false perception of sharing the albums with the intended audience since in Facebook, access is allowed/denied to the union (and not intersection) of friends in the allow/deny fields [12]. Thus, leading to unintended sharing of information. It is therefore, essential to collect user intentions first, in order to help them reduce misconfigurations in their policies.

Existing social network data privacy checking tools focus on scanning of user's profile, mainly by searching for the visibility of user's data outside their friend network, and providing recommendations to the users for limiting access on privacy sensitive items such as location, friend lists and relationship status [13] [14] [15]. However, to the best of our knowledge, there is little work w.r.t capturing user mental model of sharing in order to detect and resolve misconfigurations. Madejski et al. have made an effort to collect user's sharing intentions [16]. However, their approach requires extensive user input in the form of intended audience for each profile item category. Secondly, they have not evaluated their scheme for privacy leaks. Also, the approach does not cater data objects such as albums; the privacy settings for each album can be drastically different and therefore, can not be handled as a single category.

In this paper, we propose a scheme to capture the

user's mental model of sharing Facebook albums, and enhance this mental model through data learned from their existing policies, in order to reduce the policy misconfigurations. The possible misconfiguration patterns in Facebook users' access control policies were formulated through the analysis of Facebook privacy settings interface and users' policy patterns. We focus on Facebook, since it is one of the most popular online social networking websites today with over 955 million active users [17]. Through the Facebook API, applications can access the user's privacy settings, which enabled us to extract and analyze the user's real privacy policies. Currently, our approach only focuses on photo albums, which can be easily extended to other types of user objects. Due to the large number of albums per user, they are vulnerable to access control policy misconfigurations. Our scheme involves the user in the process of misconfiguration detection in order to increase their understanding and awareness of the detected misconfigurations.

Following are the main contributions of this paper:

- Scheme to capture users' mental model of sharing and reducing misconfigurations in their policies.
- Discussion of the possible misconfiguration patterns in Facebook users' access control policies.
- Implementation of a prototype Facebook application and a pilot study to evaluate the proposed scheme.

The rest of the paper is organized as follows: In Section II, we discuss access control and policy composition in online social networks. Section III, explains our access control policy misconfiguration detection framework in detail. The pilot study design to test our proposed scheme is described in Section IV and the results are detailed in Section V. We discuss the related work in Section VI. Finally, we conclude the paper and describe our future work.

II ACCESS CONTROL IN ONLINE SOCIAL NETWORKS

Access control enables a user to define how other users will access the information items they share. In online social networks, this is achieved using the access control settings interface provided on their websites. When a user shares an information object, they define the privacy settings consisting of the friends who should be allowed and denied access to it.

In order to assist the users in managing a large number of friends, users are provided with mechanisms to categorize their friends based on relationships. This enables the users to organize their friends based on their roles. These friend groups can be used in the policies similar to the role based access control approach [18]. As a friend grouping mechanism, Facebook introduced friend lists, and Google+ introduced the google circles feature.

The objects in social networks represent various information items that the users share. These objects consist of user's personal information i.e., profile, and posts such as status, photos and videos. There are two types of permissions on objects in social networks, namely read, and write. The *read* permission enables the user to view an object's content. If access to objects under the profile category is allowed, the allowed friends can view but can not like/comment on the object. The *write* permission enables the user to like and comment on the object. If access to objects other than those under the profile category is allowed, the corresponding friends can view the content and like/comment on them.

Facebook provides a user interface to compose and edit the privacy settings of an information object. Figure 1, shows the different Facebook privacy settings interfaces. The user can either choose from a list of default policies, which are more generic policies (See Figure 1(a)), or create custom access control policies by specifying both the specific users or groups of users who should be allowed, and the users who should be denied; exceptions (See Figure 1(b)). Friends can be organized using the Facebook friend list feature. The user can also create, update and delete the lists or groups of users (See Figure 1(c)). Currently, three types of friend lists are supported in Facebook:

1. *Custom friend lists:* These are created and populated by the user at their will. The user controls the addition and deletion of friends from the list.
2. *Default friend lists:* They are present by default when the user creates their Facebook account. The user can control the friends inside these lists. The default friend lists consist of close friends, family and acquaintances.
3. *Smart lists:* These lists are automatically created when a user updates their home, work or education. Moreover, they populate themselves without user interaction. For example, if a user

(a) Default (b) Custom (c) Friendlists

Figure 1: Facebook privacy settings interface

adds University of North Carolina at Charlotte to their Education, a smart list for this education category will be created and the user's friends who listed this school under their education will be added to the list.

In Figure 1(c), Family, UNCC Work and Purdue University are examples of default, custom and smart lists respectively.

Information object categories in Facebook include wall posts, profile information, photos and videos. Each of these categories is further sub-divided into object types. For example, wall posts can be a status update, check-in, photo, video or a life event. Similarly, profile information consists of basic information, home, work, education and interests etc. Although Facebook allows fine-grained access control on objects, there are a few limitations. For example, currently, there is no mechanism to group similar albums. Also, for hierarchical objects such as albums, it is not possible to set policy on individual pictures (excluding profile pictures).

From our past Facebook user study, we observed that Facebook users' access control policies follow the below mentioned patterns:

• **Default:** These consist of a static list of policies to choose from. Each policy has a different visibility level as shown in Table 1.

Table 1: Facebook's default policies and custom policy patterns

#	Allowed users	Denied users
Default		
1	Everyone (within and outside the friend network)	None
2	Friends	People outside the friend network
3	Friends	Acquaintances
4	User only	All friends
Custom		
1	Some Friend lists	Some Friends
2	Some Friend lists	None
3	Some Friend lists	Some Friend lists
4	All Friends	Some Friends
5	All Friends	Some Friend lists
6	Some Friends	None
7	Some Friends	Some Friends

• **Custom:** These are the user's self composed rules involving the addition of individual friends and friend lists in the set of allowed and denied users. User exceptions are possible, enabling the user to grant access to a whole friend list except a few members by inserting their names in the set of denied users. Table 1 shows the custom policy patterns.

III MISCONFIGURATION DETECTION SCHEME

In this section, we explain our approach for the detection of policy misconfigurations in user albums.

Detecting policy misconfigurations in social networks is a hard problem because 1) we do not know what an

end-user considers a misconfiguration, and what are their sharing intentions 2) it is not guaranteed that the user understands the purpose and context of the detected misconfigurations, and is likely to ignore or forget them. Therefore, we approach this problem by involving the user in the process of capturing their mental model of sharing, and guiding them in order to compose better policies.

1 SHARING INTENTION COLLECTION

The first module of our scheme focuses on capturing the user's sharing intentions for their information items i.e., the photo albums. Our sharing intention collection approach is a three step process through which the users express their sharing intentions for each album.

1. Album grouping

An typical Facebook user has approximately 10 albums on average, many of which are shared with the same audience depending on the events related to the photos. Grouping these similar albums can therefore, reduce the number of objects that the user has to focus on, by treating the albums within a group as one object. Although this step can be automated using data mining and clustering schemes, we choose manual grouping, in order to maintain accuracy. The user is asked to group their albums based on sensitivity; albums to which they would like to assign the same permissions. They can create any number of groups as they require. Figure 2 shows the album grouping interface in our prototype. The user drags each album into a particular group container. In order to have one policy per album, each album must be placed into one group only. To help the user in grouping, a tooltip containing the album's information is displayed when the user rolls the mouse over an album icon. The album information in the tooltip includes its name, privacy settings, number of access control misconfigurations related to the album's policy, and the number of photos.

2. Metadata extraction

The number of album groups created in the first step implies the number of different policies the user has in mind. We attempt to enhance this sharing intention model such that it results in secure policies. For this purpose, we extract the following additional information from user's existing policies:

Frequency of use: Number of times a particular policy was set by the user

Misconfigurations: Number of misconfiguration patterns detected in the policy

This information is presented to the user in the next step in order to influence their decisions.

3. Album policy composition

The next step in intention collection process is policy composition for the user created album groups. The users express the access control criteria based on which they grouped their albums together, by setting permissions for each album group. Instead of using Facebook's existing privacy settings interface for policy composition, we present the users with their existing policies to choose from, based on the following propositions:

- These policies are representative of the set of friends with whom the user usually shares their items. Hence, the user is most likely to use a combination of the same friends in their new album policies.

- These policies can be complemented with the extracted metadata to influence the decisions.

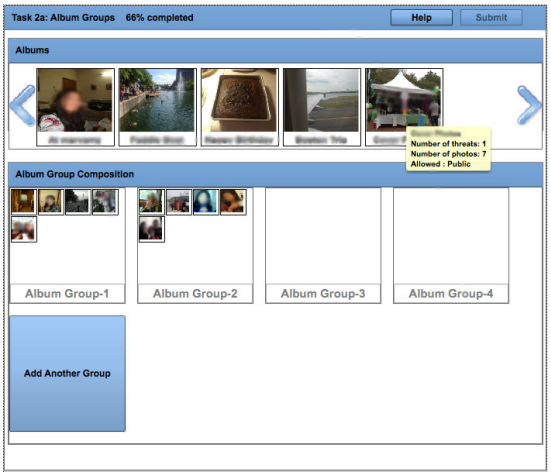


Figure 2: Album Grouping

In order to avoid the textual policies from looking verbose and unreadable, we present the user's existing policies in a visually appealing manner. We use Tag cloud visualization for this purpose. Tag cloud [19] is a visual representation of a set of words related to a particular topic. The attributes of text such as size, weight, or color are used to represent features, such as frequency of the associated terms. Tag cloud has been used by researchers for various purposes. Kuo et al. [20] have used Tag cloud to summarize descriptive web search results. Hearst et al. [21] state that the main role of Tag cloud is as a social signaler to attract peoples' attention rapidly. Eda et al. [22] have proposed an entropy based scheme to increase emphasis on emotional tags.

We split each policy into its allowed and denied components to extract tags. For example, if an album's privacy settings allow two friend lists and deny a particular friend, then this album's policy comprises of three tags.

We customize the user's existing policies using the extracted metadata and create two types of Tag clouds. In the first Tag cloud, we include the frequency of use of each policy along with its tag. However, policy usage frequency can bias the user to select a policy containing misconfigurations. For example, if most of the user's albums are public, then the size of tag related to public policy will be big, attracting the user attention towards it. Therefore, instead of increasing the tag size based on usage frequency, we keep the tag size fixed and combine the tag usage frequency with its label in order for the user to differentiate between the tags. This is shown in Figure 3(a). For the second Tag cloud, we incorporate the misconfigurations related to each tag using the equation 1. In this Tag cloud, the tags of an album policy, which are causing the access control misconfigurations are decreased in size according to the cumulative sensitivities of all the misconfigurations related to the respective tags. Each misconfiguration is assigned a sensitivity weight on a scale from 0 to 1 based on the extent of privacy leak that it can cause. The access control misconfigurations caused by each tag are calculated by scanning the user's existing album policies based on our misconfiguration patterns. For example, suppose a user has three albums with the following policies: *allow friends*, *allow public* and *allow only me* respectively. Since, the access control misconfigurations corresponding to "public" tag has the highest privacy leak, this tag will be very small in size as compared to the other two tags. Figure 3(b) shows the visual representation of album policy tags using Equation 1.

$$Tag_i = f_{max} - (f_{max} - f_{min}) * \frac{\sum_{j=1}^{No.\ MC} MCSen_j}{MCSenMax} \quad (1)$$

Where, f_{max} = Maximum font size of a tag
f_{min} = Minimum font size of a tag
$MCSen_j$ =Sensitivity of $Misconfiguration_j$
$MCSenMax$ = Misconfiguration with highest sensitivity amongst those caused by Tag_i

The Tag clouds for the allowed and denied part of the policies are calculated separately. The user composes the policy for each album group by dragging the respective tags into the *allowed* and *denied* fields. We incorporate both Tag clouds in our prototype to compare their effectiveness. The number of tags shown in a Tag cloud is also varied to study whether presenting tags only from the current album group's existing policies is better than presenting tags from all albums' existing policies.

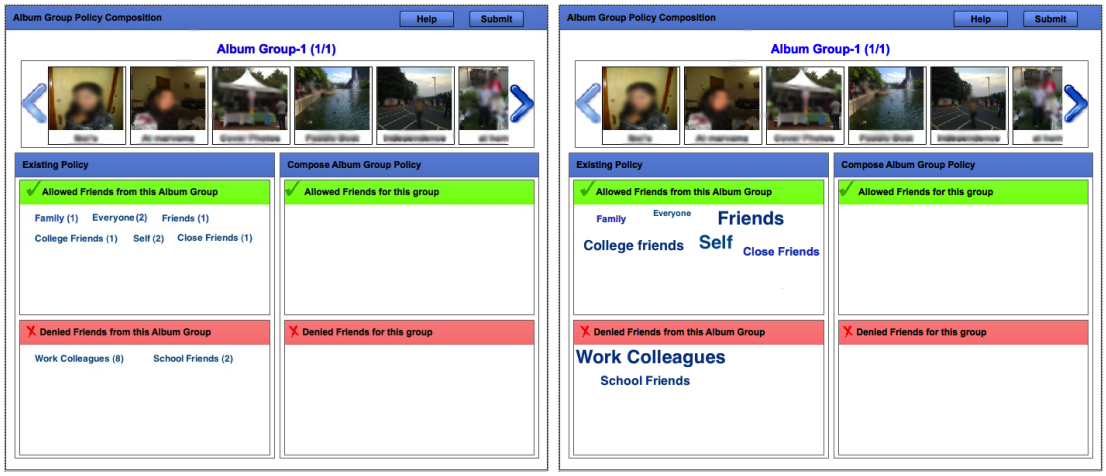
After the user composes the album group policies, we compare the original policies of the albums in a group with their respective album group policy. The user is presented with the old and new policies of each album within an album group. The matching items in the policies are shown in green color, while the mismatching items in the policies are shown in red color. If the user agrees to the new policy despite these dissimilarities, the new policy is adopted for this album, otherwise, its previous policy is adopted.

2 MISCONFIGURATION SCANNING

After acquiring the user's sharing intentions in the form of album policies, we scan them based on our misconfiguration patterns and present the detected misconfigurations to the user. Our analysis of Facebook users' policy patterns (described in Section II) revealed the following possible access control misconfiguration patterns.

P1 ***The information item is visible to people outside the friend network:*** Over the years, there has been a tremendous increase in the amount of a Facebook user's information that is public by default [23]. Due to the changes in interface, the default privacy settings of certain items have been changed. For example, recently, the profile pictures were made public and now each profile picture's settings has to be customized individually. There is a high probability that most users, who initially restricted the profile pictures' access to their friends or a subset of their friends, are unaware of this change, and each new picture they will share now will have public access. Moreover, users are unaware of the actual audience represented by Facebook's "Public" setting. Therefore, this is a potential misconfiguration leading to sharing of information with unintended audience. This scenario is possible if the chosen privacy settings are Friends of Friends, Friends and Networks, or Public.

P2 ***A friend has been explicitly denied access to this information item, but is allowed access to other information items:*** This misconfiguration can arise due to the existence



(a) Frequency of use

(b) Misconfigurations

Figure 3: Tag cloud visualization of existing album policies for album group policy composition

of a friend who has been given access to most albums but is denied access to a specific album by adding their name in the denied set of users. The user can have a false perception that this friend has been denied access to all the other albums as well. This is particularly possible if this friend is part of a smart list which has been allowed access to most albums; leading to unintended sharing of these albums with this friend. We believe that the use of friend exception in a policy indicates that the corresponding album is extremely private and should be hidden from specific users.

P3 A Facebook smart list, which updates without user concern, has been used: If a user grants access to a smart list, there is a potential chance of sharing information with unintended audience because the user does not control the list.

P4 There are common friends between friend lists: Facebook gives access to the union of all the friends/friend lists included in "Allowed" field. However, during policy composition, the user might want to allow access to only those friends who exist in all the friend lists in the "Allowed" field, leading to over-sharing of information.

P5 An empty friend list has been allowed or denied: This pattern informs the user about under-sharing of information and can arise if a friend list having no friends has been allowed access. For example, If the user allows access to a smart list which contained friends initially, but became empty over the course of time when the respective friends changed their work, education or hometown.

P6 One or more friends exist in both the allowed and denied fields: In Facebook access control, the *deny* permission takes precedence over the *allow* permission. Therefore, if a user has common friends between an allowed friend list and a denied friend list, these friends will be denied access. The user however, might have intended to allow access to this friend, with the perception that the friend only exists in the allowed friend list. This misconfiguration intends to notify the user of unintended denial of access to a friend.

P7 The information item is empty: We observed that users have empty albums with public access. For example, mobile uploads and timeline photo albums, which are created by Facebook and are "public" by default. This misconfiguration pattern informs the user about over-shared empty albums, and to be careful when adding photos to these albums.

In order to study how meaningful the above misconfiguration patterns and their privacy leaks are to the users, we ask them to rate each detected misconfiguration. See Figure 4.

3 PROTOTYPE ARCHITECTURE

We implemented a prototype of our proposed access control misconfiguration detection scheme. The prototype was built as a Facebook application, AlbumPrivacyScanner[1]. The application is hosted on our server and the back-end is based on PHP and MySQL. The client-side was implemented using Adobe Flex as a flash application. Upon installing the application, REST like Facebook APIs and Facebook Query Language are used to retrieve the user's album data, privacy settings and social connections. The collected data is transmitted over secure HTTPS based APIs to our server and stored in a MySQL database.

Figure 4: Policy Misconfiguration Rating

IV PILOT STUDY

In designing our user study[2], we set out to answer the following questions:

Q1 What are the participants' sharing intentions for albums?

Q2 How effective is our misconfiguration detection scheme w.r.t reducing policy misconfigurations and policy authoring time?

Q3 What are the participants' perceptions of our proposed access control misconfiguration patterns?

1 DESIGN

In order to answer our research questions, we built three tasks into our prototype application. The first two tasks were designed to evaluate our sharing intention collection approach and Tag cloud visualizations w.r.t misconfigurations and policy authoring time. The other task and survey was used to get participant perceptions of our misconfiguration patterns and usability of our misconfiguration detection scheme. Table 2 enumerates the user tasks.

Table 2: User Study Tasks

Sharing Intention Collection	
Task 1	Create album groups
Task 2	Compose album group policies
Access Control Misconfiguration Perceptions	
Task 3	Rate the detected misconfigurations
Survey 1	Demographics and usability of the approach

For the first two tasks, we divided the participants into four groups. Group 1 was shown the frequency of policy use based Tag cloud, constructed from the policies of albums within an album group, to serve as an indicator for helping them in policy selection. Group 2 was shown the frequency of policy use based Tag cloud constructed from the policies of all the user albums. Group 3 was shown misconfiguration based Tag cloud constructed from the policies of albums within an album group while Group 4 was shown misconfiguration based Tag cloud constructed from the policies of all the user albums. Task 1 and 2 involved only a subset of the participant's albums (for the purpose of the study). This subset was limited to 15 and included all the albums with access control misconfigurations (so that we can later compare the two Tag cloud approaches w.r.t reducing misconfigurations). These two tasks were intended to collect user's sharing intentions for album policies. Each participant was asked to group these albums according to the privacy settings that they wanted to assign to each group. In Task 2, the participants set permissions for the first album group using one of the assigned policy based Tag cloud. Then the participant was presented with the differences between the old and new policies of albums within this group, based on which they determined whether they want to adopt the new policy for that album or not. Task 2 was repeated for all the album groups. In Task 3, the participant's original album policies were scanned on our access control misconfiguration patterns described in Section III and they were asked to review and rate the detected misconfigurations. For each detected misconfiguration, the participant gave a rating by specifying whether they considered it a misconfiguration for that particular album or not and how serious it

[1] https://apps.facebook.com/albumprivacyscanner/
[2] *Approved IRB Protocol # 11-08-01*

was. The seriousness responses were collected on a Likert-scale from 1(Low Seriousness) to 7(High Seriousness). Upon completion of the four tasks, the participant was asked to complete a short survey. First half of the survey comprised of demographic questions while the second half was focused on usability of our tool. Each question was designed to capture the participant's perceptions in the following areas:

Ease of Use: The participant should be able to detect the misconfigurations in their album policies easily and intuitively. Otherwise, the scheme will lose its significance.

Readability: In addition to being easy to use, it should be understandable. An average user should be able to comprehend the involved tasks. The misconfigurations presented should be readable in order for the user to take appropriate action.

2 PARTICIPANTS

We recruited our participants from Amazon Mechanical Turk[3]. Amazon Mechanical Turk is a crowd sourcing marketplace which pairs requesters of work and workers. Requesters formulate work into Human Intelligent Tasks (HIT) which are individual tasks that workers complete. We set up our prototype Facebook application as a HIT. It included three tasks and a survey, as described in Section IV. To better control the quality of the recruited participants, we mandated that each worker have a 95% HIT approval rating, or better. The HIT took approximately 10-15 minutes to complete, for which each worker was paid a fee of $0.50. A total of 96 participants successfully completed the pilot study, 49 male and 47 female. Most of our participants were young, fairly well educated and active Facebook users who were members for more than 2 years. 42.7% were between the ages of 18 to 25. 43.75% were between 26-39 and 13.54% were aged 40 and above. 78% had between two and four years of college education. In addition, as part of the demographics section of our survey, we collected Westin privacy sentiment information summarized below with definitions of Unconcerned, Pragmatist and Fundamentalist provided by [24]:

Unconcerned: 4.1% of our user study population. *This group does not know what the "privacy fuss" is all about, supports the benefits of most organizational programs over warnings about privacy abuse, has little problem with supplying their personal information to government authorities or businesses.*

Pragmatists: 56.7% of our user study population. *This group weighs the value to them and society of various business or government programs calling for personal information, examines the relevance and social propriety of the information sought, wants to know the potential risks to privacy or security of their information, looks to see whether fair information practices are being widely enough observed, and then decides whether they will agree or disagree with specific information activities.*

Fundamentalists: 39.17% of our user study population. *This group sees privacy as an especially high value, rejects the claims of many organizations to need or be entitled to get personal information for their business or governmental programs, and favors enactment of strong federal and state laws to secure privacy rights and control organizational discretion.*

V STUDY RESULTS

This section discusses our pilot study results.

1 PARTICIPANTS' SHARING INTENTIONS

We calculated the following metrics to evaluate our intention collection approach:
Number and size of album groups: As described in Section III, our sharing intention collection approach was based on the assumption that the users tend to have albums with similar privacy settings. These albums can therefore be grouped together. For this purpose, we calculated the number of album groups created by a participant and the number of albums were placed in one group.
Number of album policies: In order to demonstrate that the user's sharing intentions are different than their actual policies, we calculated the total number of album policies after using our approach and compared it with the total number of previous album policies.

Table 3 shows the album grouping statistics of our participants. Fundamentalists and pragmatists created the same number of album groups i.e., around 3, demonstrating that there were at-least 3 privacy settings re-used on multiple albums. Unconcerned participants created 4 album groups on average. Fundamentalists and pragmatists had more than 4 albums

[3]https://www.mturk.com/

in one album group. Therefore, at-least 4 of the participant albums had the similar privacy settings.

The change in the number of policies is shown in Table 4, demonstrating that the intended album policies were different than the actual album policies of the underlying participants. Dependent t-tests showed that there is a significant difference between the number of policies the participants had before and after using our scheme with a p-value of 0.004. One factor could be the participant's memory; they did not remember their actual policies and were of the opinion that the intended audience for the album is the same as its actual audience.

Table 3: Participants' Album Grouping Statistics

Participants	**No. of album groups (μ)**	**Album group size (μ)**
Fundamentalist	3.55	4.96
Pragmatist	3.46	4.18
Unconcerned	4	2.54
All	3.52	4.42

Table 4: Number of album policies

Participants	**No. of Policies**		
	Before	After	p-value
Fundamentalist	2.15	1.71	0.08
Pragmatist	2.22	1.85	0.08
Unconcerned	3	1.25	0.06
All	2.22	1.77	0.004

2 TAG CLOUD VISUALIZATION EVALUATION

In this section, we discuss the effectiveness of our learned data based Tag cloud visualization, in enhancing the participants' sharing intention model. We used the following evaluation criteria:

Misconfiguration change: The decrease in the number of misconfigurations, calculated as: *Number of misconfigurations detected in the original policies - Number of misconfigurations detected after composing policies using Tag cloud*. This metric indicates whether the Tag cloud visualization enabled the participant to compose secure policies.

Policy authoring time: The time taken to compose an album group policy using the Tag cloud visualization. This metric indicates which of the four Tag cloud visualizations results in least policy authoring time.

To determine the effect of Tag cloud on misconfigurations, we calculated the number of misconfigurations detected in participants' policies with and without using Tag cloud visualization. Table 5 shows the evaluation of Tag cloud visualizations with varying number of tags. The fields for unconcerned participants have been omitted, since the number of observations were not sufficient to conduct statistical t tests. Pairwise t tests on the number of misconfigurations before and after using Tag cloud showed that our scheme enabled the participants to decrease the number of misconfigurations in their policies. The difference in misconfigurations (before - after) is positive. However, One way ANOVA test to compare the misconfiguration change among the four Tag cloud visualizations revealed no significant difference between the decrease in number of misconfigurations according to the frequency of policy use based Tag cloud with tags per album group, frequency of policy use based Tag cloud with tags of all albums, misconfiguration based Tag cloud with tags per album group, misconfiguration based Tag cloud with tags of all albums.

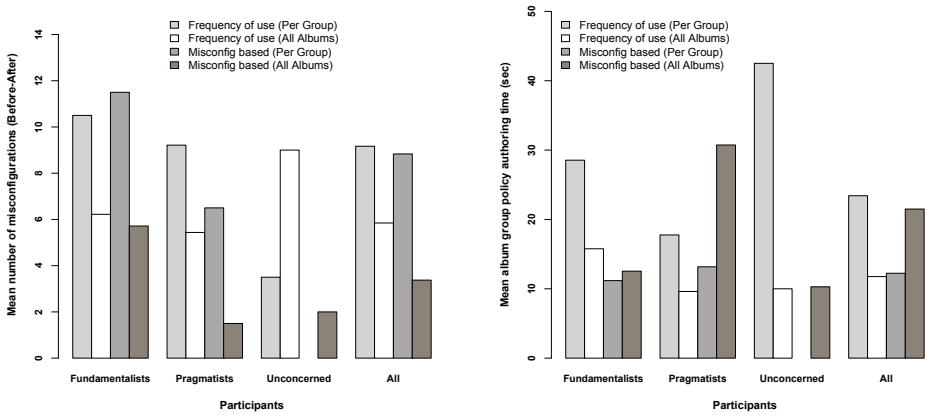
Next, we compared the average amount of time (sec) taken by a participant to author an album group policy. Table 5 shows that there was a significant difference in the policy authoring time of pragmatists. Pairwise comparison t-tests between policy authoring time of the four Tag cloud visualizations for pragmatists showed that using frequency of policy use based Tag cloud, it took significantly less time to author a policy (9.6 sec) as compared to using misconfiguration based Tag cloud (30.7 sec) when shown tags from all the albums, with (p-value 0.04). However, when shown Tag cloud based on the tags of the current album group only, misconfiguration based Tag cloud (13.1 sec) resulted in lesser policy authoring time compared to frequency of policy use based Tag cloud (17.7 sec).

3 PARTICIPANTS' PERCEPTIONS

The participant perceptions of our access control misconfiguration patterns were gathered from their misconfiguration ratings, using the following criteria:

Misconfiguration Votes: The number of participants who considered a misconfiguration pattern important, out of the total number of participants who got that misconfiguration in one of their album policies.

Misconfiguration Seriousness: The participant rating of how serious a privacy threat is posed by the detected misconfiguration. This metric was mea-



(a) Misconfiguration change

(b) Policy authoring time

Figure 5: Comparison of Tag cloud visualizations

Table 5: Comparison of Tag cloud visualizations

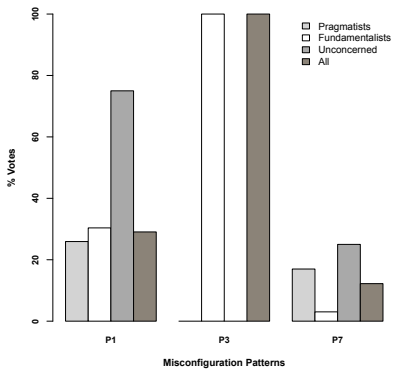
Participants	**Frequency based (Per Group)** (μ)	**Frequency based (All Albums)** (μ)	**Misconfig based (Per Group)** (μ)	**Misconfig based (All Albums)** (μ)	**Effect Size** r^2	**F-value**	**p-value**
Misconfigurations (before - after)							
Fundamentalist	10.5	6.22	11.64	5.71	0.04	0.48	0.69
Pragmatist	9.21	7.43	7.25	5.50	0.08	1.55	0.21
All	9.16	7.07	9.30	5.37	0.04	1.45	0.23
Avg. album group policy authoring time (sec)							
Fundamentalist	28.54	15.77	11.17	12.54	0.16	2.13	0.11
Pragmatist	17.76	9.61	13.17	30.73	0.15	2.85	0.04
All	23.41	11.75	12.23	21.49	0.08	2.81	0.04

sured on a Likert scale (1-7), where 1 indicates low seriousness and 7 indicates high seriousness.

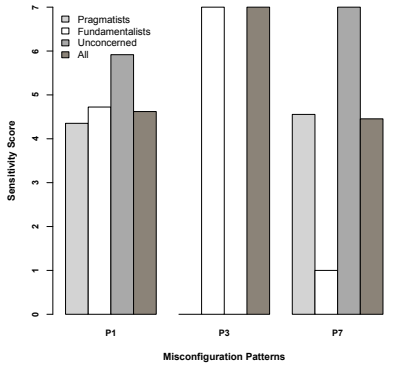
Amongst our 7 misconfiguration patterns, we observed that only 3 were detected in the participants' album policies. Figures 6(a) and 6(b) show the participant perceptions of our misconfiguration patterns. Pattern P1(Album's visibility outside the friend network) was the most commonly detected misconfiguration pattern. Despite being the misconfiguration with highest privacy leak, less than 50% of fundamentalists and pragmatists considered it important, with seriousness scores of 4.72 and 4.35 respectively. Only unconcerned participants gave it 75% votes with a seriousness score of 5.91. It is possible that the fundamentalist and pragmatist participants, who did not vote for it were aware of the audience represented by "public" setting and only gave public access to insensitive information. Pattern P3 (smart list usage) was only detected in the policies of fundamentalists, and was considered important, receiving 100% votes and a high seriousness score of 7. This shows that the participants are unaware of the smart lists and their difference from the other friend lists. Pattern P7 (Empty album) was considered the least important and received the lowest number of votes. Overall, it received 12.22% votes with seriousness score of 4.45, demonstrating its importance to the participants who considered misconfiguration pattern P1 meaningful. Other misconfigurations involving more than one friend lists and user exceptions could not be detected, since most of our study participants did not

have custom policies involving friend lists and exceptions. The average number of friend lists created per participant was 2 and the average number of smart lists per participant was 7. However, less than 10% of a participant's policies involved friend lists and user exceptions.

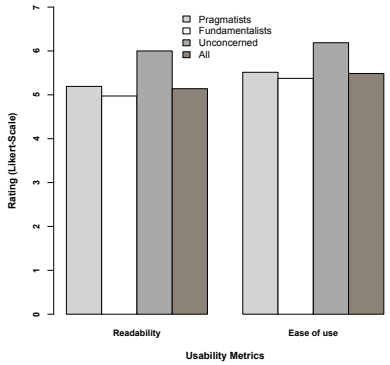
Secondly, we performed a qualitative analysis of our misconfiguration detection application prototype, using the participant responses to our survey questions. The two usability metrics measured were, ease of use and readability, as discussed in Section IV. Figure 6(c) shows that unconcerned participants gave the highest ratings, however, fundamentalists and pragmatists had slightly lower usability rating. Overall, the average rating for readability and ease of use was greater than 5. This demonstrates that the participants easily understood the tasks involved in misconfiguration detection scheme and found the application's interface usable. The average time taken per participant to complete all the tasks was only 12.16 minutes.



(a) % of votes received by misconfiguration patterns



(b) Seriousness score of misconfiguration patterns



(c) Usability of misconfiguration detection scheme

Figure 6: Participant Perceptions

VI RELATED WORK

Recently, there has been a surge on creating awareness among the social network users about privacy issues in their existing policies, through interactive applications. One category of such applications tests the user's knowledge of how the social networking website handles privacy for various data items. PrivacyIQ [25] is a Facebook application in which the user is asked a set of questions to test their knowledge of privacy leaks regarding friends, content, Apps and check-ins, and is provided with a score based on correct answers. The other category scans the user's Facebook profile and calculates a privacy score based on how much data is exposed to people outside their friend network. ProfileWatch [13], PrivacyCheck [14] and Secure.me [15] fall into this category. Secure.me even highlights questionable posts that contain taboo words. However, these schemes are not very deep in their privacy analysis and do not take the user's intentions into account.

In parallel, efforts are being made to improve the end-user's ability to compose better access control policies. Egelman et al. [26] proposed a Venn diagram based privacy settings interface to cater for access control scenarios that lead to user errors. The Facebook's current privacy interface does not handle such scenarios. Their user study shows that these scenarios lead to access control errors by users and result in over-sharing of the data without their knowledge. We leverage their scenarios related to common friends between friend lists, in our misconfiguration patterns. Yuksel et al. [27] proposed an API to auto-

matically cluster the users' friends in order to assist them in managing privacy on social networks. Similarly, Jones et al. [28] have analyzed and identified 6 criteria of how users group their friends for privacy management. Recently, a general privacy wizard for social networks has been proposed to eliminate the burden of fine-grained policy specification from the users' shoulders [10]. The design is based on the fact that users have an inherent set of rules based on which they set their privacy preferences. The wizard extracts these rules by getting the user's input on a small subset of their friends and using their friends' visible data. A classifier is then trained on this data to specify privacy preferences for the rest of his friends. However, since the wizard requires user input, it is not possible to completely remove the policy specification task from the user.

Mazzia et al. [9] have proposed PViz, a policy visualization tool for social networks which aims to align with the users' mental model and helps them understand their policies quickly. The basis of their approach is that users conceive their networks in terms of communities and therefore want to see how a particular data item is visible to the friends in that community. While this is an interesting visualization scheme, they do not dig deeper into a Facebook item category e.g., photo albums, and treat it as a single item. However, the privacy settings of each album can be very different. Moreover, their scheme leaves it up to the user to discover misconfigurations. Our scheme detects the misconfigurations in a user's policies for Facebook items such as albums, which cannot be considered a single category. We also engage the user in the process to increase their understanding. Anwar et al. [8] state that Facebook style social network users compose and continuously adjust their policies according to the impression they want to present to different friends. They have developed a reflective policy assessment tool which enables the user to view his profile impression as it appears to a particular friend. However, they use graphs in their user interface, which are difficult for an average user to understand. Besmer et al. [29] have studied how the information of users in the community impact their policy decisions on social networks. Lipford et al. [30] evaluate two policy presentation interfaces i.e, audience view and expandable grids and propose a combination of the two interface to get the best of the both worlds.

VII CONCLUSIONS AND FUTURE WORK

In this paper, we proposed a scheme to detect misconfigurations in access control policies on social networks. Our misconfiguration detection scheme focused on Facebook album policies. The user's sharing intentions model was collected with the help of user interaction in the form of album group policies. Tag cloud visualizations were used for policy composition; the existing policies were presented to the user together with extracted metadata, in order to guide their sharing intention model. The resulting album policies were scanned on our proposed misconfiguration patterns. The users' intended policies were found out to be significantly different than their actual policies. Our scheme was able to detect the misconfiguration patterns in album policies. However, the reduction in the number of misconfigurations after using our approach was not significant. User perceptions of our misconfiguration patterns were collected in order to study the seriousness of the privacy threats they possessed. The qualitative analysis of our overall scheme demonstrated its ease of use and readability.

Currently, we are planning to conduct a large scale user study and address the following limitations in our current work: 1) Having a different criteria for handling the denied policy related tags in the Tag cloud visualization, in order to prevent the user from allowing the respective users in the new policies 2) The scalability of our approach, i.e., how useful is our approach if the user has hundreds of albums. 3) The extension of our approach to other profile items.

VIII ACKNOWLEDGEMENTS

This research was partially supported by grants from the National Science Foundation (NSF-CNS-0831360, NSF-CNS-1117411) and a Google Research Award. We would like to thank Adharsh Desikan and Rahul Ramkumar for their help with the development and implementation of the Tag could policy application.

References

[1] I.-F. Lam, K.-T. Chen, and L.-J. Chen, "Involuntary information leakage in social network services," in *Proceedings of IWSEC 2008*, 2008.

[2] N. Y. Times, "Facebook privacy: A bewildering tangle of options," http://www.nytimes.com/interactive/2010/05/12/business/facebook-privacy.html/, 2010.

[3] A. Acquisti and R. Gross, "Imagined communi-

ties: Awareness, information sharing, and privacy on the facebook," in *Privacy Enhancing Technologies*, 2006, pp. 36–58.

[4] L. Church, J. Anderson, J. Bonneau, and F. Stajano, "Privacy stories: confidence in privacy behaviors through end user programming," in *Proceedings of the 5th Symposium on Usable Privacy and Security*, ser. SOUPS '09. New York, NY, USA: ACM, 2009, pp. 20:1–20:1. [Online]. Available: http://doi.acm.org/10.1145/1572532.1572559

[5] R. Gross and A. Acquisti, "Information revelation and privacy in online social networks," in *Proceedings of the 2005 ACM workshop on Privacy in the electronic society*. ACM, 2005, pp. 71–80.

[6] H. Lipford, A. Besmer, and J. Watson, "Understanding privacy settings in facebook with an audience view," in *Proceedings of the 1st Conference on Usability, Psychology, and Security*. USENIX Association Berkeley, CA, USA, 2008, pp. 1–8.

[7] 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*, ser. BCS-HCI '08. Swinton, UK, UK: British Computer Society, 2008, pp. 111–119. [Online]. Available: http://portal.acm.org/citation.cfm?id=1531514.1531530

[8] M. M. Anwar and P. W. L. Fong, "A visualization tool for evaluating access control policies in facebook-style social network systems," in *Symposium On Applied Computing*, ser. SAC, 2012, pp. 1443–1450.

[9] A. Mazzia, K. LeFevre, and E. Adar, "The pviz comprehension tool for social network privacy settings," in *Symposium on Usable Privacy and Security*, ser. SOUPS, 2012, p. 13.

[10] L. Fang and K. LeFevre, "Privacy wizards for social networking sites," in *Proceedings of the 19th international conference on World Wide Web*, 2010, pp. 351–360.

[11] Y. Liu, K. Gummadi, B. Krishnamurthy, and A. Mislove, "Analyzing facebook privacy settings: User expectations vs. reality," in *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*. ACM, 2011, pp. 61–70.

[12] M. Madejski, M. Johnson, and S. M. Bellovin, "The failure of online social network privacy settings," Tech. Rep., 2011.

[13] "Profile watch," http://www.profilewatch.org/.

[14] "Privacy check," http://www.rabidgremlin.com/fbprivacy/.

[15] "Secure.me," https://apps.facebook.com/secure-me/.

[16] M. Madejski, M. Johnson, and S. Bellovin, "A study of privacy settings errors in an online social network," in *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2012 IEEE International Conference on*. IEEE, 2012, pp. 340–345.

[17] "Facebook statistics," http://newsroom.fb.com/Key-Facts.

[18] D. Ferraiolo and R. Kuhn, "Role-based access control," in *In 15th NIST-NCSC National Computer Security Conference*, 1992, pp. 554–563.

[19] A. W. Rivadeneira, D. M. Gruen, M. J. Muller, and D. R. Millen, "Getting our head in the clouds: toward evaluation studies of tagclouds," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '07. New York, NY, USA: ACM, 2007, pp. 995–998. [Online]. Available: http://doi.acm.org/10.1145/1240624.1240775

[20] B. Y.-L. Kuo, T. Hentrich, B. M. . Good, and M. D. Wilkinson, "Tag clouds for summarizing web search results," in *Proceedings of the 16th international conference on World Wide Web*, ser. WWW '07. New York, NY, USA: ACM, 2007, pp. 1203–1204. [Online]. Available: http://doi.acm.org/10.1145/1242572.1242766

[21] M. A. Hearst and D. Rosner, "Tag clouds: Data analysis tool or social signaller?" in *Proceedings of the Proceedings of the 41st Annual Hawaii International Conference on System Sciences*, ser. HICSS '08. Washington, DC, USA: IEEE Computer Society, 2008, pp. 160–. [Online]. Available: http://dx.doi.org/10.1109/HICSS.2008.422

[22] T. Eda, T. Uchiyama, T. Uchiyama, and M. Yoshikawa, "Signaling emotion in tagclouds," in *Proceedings of the 18th international conference on World wide web*, ser. WWW '09. New York, NY, USA: ACM, 2009, pp. 1199–1200. [Online]. Available: http://doi.acm.org/10.1145/1526709.1526927

[23] "Facebook privacy," http://mattmckeon.com/facebook-privacy/.

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

[25] "Privacy iq," https://apps.facebook.com/privacy_iq/.

[26] S. Egelman, A. Oates, and S. Krishnamurthi, "Oops, i did it again: mitigating repeated access control errors on facebook," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '11. New York, NY, USA: ACM, 2011, pp. 2295–2304. [Online]. Available: http://doi.acm.org/10.1145/1978942.1979280

[27] A. S. Yuksel, M. E. Yuksel, and A. H. Zaim, "An approach for protecting privacy on social networks," in *Proceedings of 5th International Conference on Systems and Networks Communications*. Washington, DC, USA: IEEE Computer Society, 2010. [Online]. Available: http://dx.doi.org/10.1109/ICSNC.2010.30

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

[29] A. Besmer, J. Watson, and H. R. Lipford, "The impact of social navigation on privacy policy configuration," in *Symposium on Usable Privacy and Security*, ser. SOUPS, 2010.

[30] H. R. Lipford, J. Watson, M. Whitney, K. Froiland, and R. W. Reeder, "Visual vs. compact: a comparison of privacy policy interfaces," in *CHI*, 2010.