Social Access Control

Andreas C. Sonnenbichler

Abstract Facebook is one of the most important social networking sites used by over a billion of people. Facebook offers a specialized and rather limited approach to decide upon content privacy for it's users. In this paper we analyze the content protection features Facebook offers. We suggest four classes of Facebook users ranging from consumers with very limited content privacy requirements to consumers interested in fine-granular content restrictions. We want to empower the customer to choose on the access control model meeting their specific requirements. The access control model shall be customer-based and not modeled on a general Facebook-wide level. We show how such a flexible approach can be introduced into Facebook by the usage of the Access Definition and Query Language.

1 Introduction: Where Facebook Fails

In 2002, nobody knew Facebook. In 2012, Facebook got more than 1 billion users (Vance, 2012). Launched in 2004 (Phillips, 2007), Facebook today is *the* social networking platform. Probably anybody using a computer and the Internet has at least heard of Facebook. People use Facebook to post status messages, share images and videos, chat and com-

Andreas C. Sonnenbichler
Karlsruhe Institute of Technology
✉ andreas.sonnenbichler@kit.edu

Customer & Service Systems
KIT Scientific Publishing
Vol. 1, No. 1, S. 133–145, 2014

DOI 10.5445/KSP/1000038784/15
ISSN 2198-8005

municate. They stay in contact with their acquaintances, share content with friends, organize parties, share travel pictures and videos, and much more. Facebook is seen to have so much influence on peoples' life, that even the term "generation Facebook" has been used (cf. Kord, 2008).

However, Facebook is not uncriticized. One of the main criticisms is that Facebook offers very limited functionality in protecting the content of users.

In this paper, we analyze the options Facebook provides to protect user data. We show that the underlying access control model is a simplified access control list (ACL). ACLs are only one way to protect data and may not be the appropriate way for every Facebook users: There may be users (e.g. companies) who require very limited content protection as every content they provide can be accessed freely. There may be security un-aware users who require only basic access control. There may be security-aware users who want to specifically decide which of their posts can be seen by whom. There even may be "paranoid" users who exactly want to define which of their content elements can be seen by which users for a specific time period. It is our target to empower Facebook customers, so that the customer can decide upon the way how his data is protected.

First, we analyze the features Facebook currently offers to protect customer content. Second, we formalize this model. Third, we discuss some options for different approaches which might be adequate for certain customer needs. Fourth, we show what is required to empower the customer to choose his preferred way of data protection.

2 Analyzing Facebook's Data Privacy Options

Which options does Facebook offer to the customer to decide upon protecting his data? When this paper was written, Facebook offered a specific way of information protection for it's users.

The menu point "menu" allows to change general account settings like the customer's name, his email address, password, and language settings. The information here is not directly related to content privacy.

The second menu point "security" offers

1. Settings allowing to change the security question a user must answer when he has forgotten his password;

Table 1 Facebook's privacy settings menu as effective on 2013-01-06 depicted as table for better readability

Privacy Settings and Tools

Who can see my stuff?	**Who can see your future posts?** You can manage the privacy of things you share by using the audience selector **right where you post**. This control remembers your selection so future posts will be shared with the same audience unless you change it.		Close
	Review all your posts and things you're tagged in		Use Activity Log
	Limit the audience for posts you've shared with friends of friends or Public?		Limit Past Posts
Who can look me up?	Who can look you up using the email address or phone number you provided?	Friends of Friends	Edit
	Do you want other search engines to link to your timeline?	Off	Edit

2. Notification settings in the case a user connects from a device he has previously not used used to connect to Facebook;
3. Settings for making mandatory use of the https protocol instead of the unencrypted http protocol.

Again, these settings provide no usage for data privacy.

The third menu option "privacy" is depicted in figure 1. The option "who can see my stuff" let the customer change the default settings who receives status updates of the customer. E.g. when this option is changed to "friends", all Facebook friends of the user receive status updates of the user. However, this option only changes the default setting. A customer may change the target group of a status update for each update specificly. This option is related to data privacy, as it allows to set a kind of protection level for a status update.

The option "review all posts and things you're tagged in" allows to review the "activity log" of the user. The "activity log" includes all status updates, content sharing and change of information in the profile of the user, e.g. if a user made new friends, posted a video, commented something, pressed the "I-like" button and so on. The offered option "review all posts and things you're tagged in" lists the activity log and let the user decide if the specific entry shall be part of his "Facebook timeline" or shall be excluded from it. The "Facebook timeline" is the set of all activities of the user ordered by their creation time. This option is related

to data privacy and enables the customer to hide activities from his timeline.

The option “limit the audience for old posts on your timeline” allows a customer to restrict all content of the customer’s timeline to be visible only for friends. This feature is a macro allowing to reset the protection level of old data to “friends”. Actually, this is not an additional feature for data protection: The customer can manually update all old activity entries and restrict it to “friends”. The feature is a comfort function allowing to modify a lot of data with only one click.

The two menu points “who can look me up” allow the customer to decide who (friends, friends of friends, public) may lookup his e-mail address and phone number. Further, the consumer can decide whether Facebook allows search engines like Google to lookup the activity data of the customer. The latter functionality only applies to content shared with “public”. Thus, it is not an override to make all protected content available on a general level.

Summarizing the data protection functionality of this menu, it allows a customer to set the protection level of user activity (per default and per update), to exclude updates from the customer’s timeline and to limit search functionality for the e-mail address, telephone numbers, and public content for search engines.

The menu “timeline and tagging” is depicted in figure 2. “Who can add things to my timeline” allows the customer to define which users may post content on his activity log (friends or no one) and if these posts must be reviewed by the customer before they appear in his activity log.

The menu point “review what other people see on your timeline” presents the activity log simulating being a public or a specific user. This function is very useful in verifying the security settings. However, it is not used to define security settings.

The menu points “who can see posts you’ve been tagged in on your timeline” and “who can see what others post on your timeline” allow a customer to limit the tagging of others users, e.g. when a friend marks the customer on an image, or posts of other users on the customer’s timeline.

The menu option “review tags people add to your own posts before the tags appear on Facebook” allows the customer to decide if tags, a user assigns to his posts, e.g. comments, tags on images, have to be reviewed before they appear. The option “when you’re tagged in a post, who do you want to add to the audience if they aren’t already in it” has to

Table 2 Facebook's timeline and tagging menu as effective on 2013-01-06 depicted as table for better readability

Timeline and Tagging Settings			
Who can add things to my timeline?	Who can post on your timeline?	Friends	Edit
	Review posts friends tag you in before they appear on your timeline?	Off	Edit
Who can see things on my timeline?	Review what other people see on your timeline		View As
	Who can see posts you've been tagged in on your timeline?	Friends	Edit
	Who can see what others post on your timeline?	Friends	Edit
How can I manage tags people add and tagging suggestions?	Review tags people add to your own posts before the tags appear on Facebook?	Off	Edit
	When you're tagged in a post, who do you want to add to the audience if they aren't already in it?	Friends	Edit
	Who sees tag suggestions when photos that look like you are uploaded? (this is not yet available to you)	Unavailable	

be explained in detail: Let a user share content. "Our" customer is then marked by somebody (e.g. the sharing user, himself or a third user), e.g. on a picture the customer is said to be someone depicted. Then, Facebook adapts the users being able to see this content depending on the menu option. It offers the settings "friends", "only me", and "customs". The latter allows to include or exclude specific users. The privacy functionality offered here is an adaption of the people being able to see content, if the customer is tagged on the content.

The last option, "who sees tag suggestions when photos that look like you are uploaded" is currently not implemented by Facebook.

Summarizing the timeline and tagging menu, it offers the following data protection functionality:

1. restrict comments and tags of other users to be reviewed by the customer,
2. define who sees tagging and posts from others in the customer's timeline,

3. define who is added to the list of users able to see content when the customer is tagged on content.

The menu "blocking" (not depicted here) allows to restrict the viewing rights of certain friends. Concerning content, when entered in the list of "restricted", these users are not perceived as friends but public users. If a user is added to this block list, Facebook handles him the same way as if he has not been defined as a friend of "our" customer.

All other menu points of Facebook's security menu (not depicted here) are not related to content privacy settings.

3 Formalizing Facebook's Security Model

In the last section we analyzed the security options Facebook offers concerning data privacy. In this section we want to formalize the access control model used by Facebook.

The access control model used by Facebook can be described by a simplified access control list (ACL). Access control lists allow to assign each object a list of users combined with their access right.

ACLs belong to discretionary access control models and are based on the famous HRU-model (Harrison et al, 1976; Harrison and Ruzzo, 1978). For background information for this kind of model type we refer to Lampson (1971, 1974), Graham and Denning (1972), and Benantar (2006).

Basically, in an ACL for each privacy protected object a list of subjects exists. Each subject is associated with a set of access rights. Objects in access control are entities to be protected (e.g. files, posts, activity list entries) and subjects are entities accessing those objects (e.g. Facebook users).

Facebook's access control model simplifies this approach in two ways: (1) It omits a general model of access rights. Only the access right "read" or "view" is modeled. (2) It assigns subject classes to users. These classes are "public", "friends of friends", "friends" and "myself". Each subject is assigned to such a class. In contrast to many other access control models, these subject classes are consumer-centered. The class "friends" differs for each subject, e.g. the class "friends" of consumer A is different from the class "friends" of user B.

An object modeled in Facebook's access control model is an activity list entry, e.g. a status update, an image, a video, a comment, and so on. A subject is a user identified by her e-mail address.

4 Empowering Facebook Customers By Choosing Customer-Based Access Control Models

We said, that we want to let the customer decide how his data is protected. Let us first define, what we exactly mean: We define a customer as a Facebook user. Technically, a customer is represented by an account identified by an unique e-mail address. We refer to a customer in the context of access control as "subject". The data of the customer consists of all content and information the customer provides, thus each activity log entry. This includes e.g. status messages ("posts"), images, videos, "I-likes", and comments on content of other people. Further more, "data" includes all profile information, e.g. his name, address, phone number, geo-location, e-mail address and so on. Also included are the friends of a user, his groups, calendar, and his "Facebook applications". To each such data entity we refer as "object".

Let us briefly suggest four Facebook user classes with different security requirements for their content:

1. There may be subjects representing companies who require limited content protection: The content they provide can be accessed freely (public access, even for non-Facebook users) or publicly to all Facebook users.
2. There may be privacy-unaware users who require only basic access control. Their content can be shared by everyone or with their friends only.
3. There may be privacy-aware users who want to specifically decide which of their content can be seen by whom. The content protection is based on a groups, thus content can be offered to close-friends, friends, acquaintances, or publicly.
4. There may be "paranoid" users who exactly want to define which of their content elements can be seen by which users for a specific time period. Paranoid consumers may want to make this decision not on a group level ("this picture may be seen by my friends") but on a very

detailed level (”this picture may be seen only by users A and B for the next two weeks”).

We want to empower the customer to choose the access control model he requires. Currently, Facebook offers only one hard-coded access control model. We want to enable Facebook to offer unlimited access control models which can be defined by their customers:

One choice can be the simplified ACL model which Facebook currently uses. This is an adequate choice for the privacy-unaware user.

Another choice can be a “Bell-LaPadula”-like access control model (cf. Bell and LaPadula, 1973, 1975): Each subject is assigned a security level, e.g. “close friends”, “friends”, “acquaintances”, “public”. Each object is assigned a protection level, e.g. “close friends”, “friends”, “acquaintances”, “public”. Both lists represent a hierarchy. A subject may see content, if the subject’s security level is at least as high as the object’s protection level. Such a model is appropriate for a security-aware consumer as it allows him to classify his content.

A third option for the access control model chosen by the customer is a binary model: Subjects are grouped into “friends” and “public”. The group assignment is done automatically by making a user a Facebook friend (or not). Each object is then assigned a flag “friend” allowing only Facebook friends to view the item or “public” making it available for anyone (even non-Facebook users). Such a model can be appropriate for a company offering most of it’s content freely.

We can think of several more access control model options, e.g. a power-user model, where the model is specifically designed by and for a specific customer (for our “paranoid” user class). We do not want to deepen the possible models here, but continue with the steps required to provide a free choice for the access control model for every customer.

5 What A Customer-Based Access Control Model Requires

To empower the customer to choose his preferred access control model, the following steps have to be undertaken:

First, the access control component must be modularized. The access checks performed by Facebook’s application must not assume a specific access control model and realize this through hard-coded security com-

ponents. Instead, it has to be externalized to a separate software component. This requirement is basically a system-design feature which must be simply realized by Facebook's software designers. Instead of hard-coding the access control model, each request is delegated to an external component.

Second, this external access control component must be based on a meta-model for access control. This allows the component to be defined to work in a specific manner. This "specific manner" is defined by the access control model used by a specific user. This second requirement can for example be realized by the Access Definition and Query Language (ADQL) (Sonnenbichler and Geyer-Schulz, 2012; Sonnenbichler, 2013). ADQL is a software service allowing to define access control models, policies, facts and queries for access requests. The definition of the used access control model is written in a formal language, ADQL. It allows to model all previously mentioned access control models user-specifically and realize user-specific groups. Further more, existing access policies and facts can be modeled, and of course, queries can be issued and are answered if access is granted based on the current model of the user, facts, and policies.

As a short demonstration of the usage of ADQL we provide the definition of the current access control model of Facebook in ADQL:

```
# Define users, activity log entries and ACL entries
CREATE CONTAINERS users, act;
CREATE CONTAINERS acls: {public, fof, friends, hidden};

# Define content owners, friends, and activity ACLs assignments
CREATE RELATIONS owners (act, users);
CREATE RELATIONS friends (users, users);
CREATE RELATIONS acl (act, acls);

# Define policies
# The owner may always see all her content
CREATE POLICY p_owner: { (owners([act],.),[users]) };
# Public content is freely accessible
CREATE POLICY p_public: { (acl([act],.),{public}) };
# Content for friends accessible for friends
CREATE POLICY p_friends: { (acl([act],.),{friends}), (owners([act],.),
    friends([users],.)) };
# Content for friends-of-friends accessible for friends-of-friends
CREATE POLICY p_fof: { (acl([act],.),{fof}), (owners([act],.),friends(
    friends([users],.),.)) };
```

For a comprehensive description of the syntax and semantics of ADQL as well as examples of additional access control models we refer the reader to (Sonnenbichler and Geyer-Schulz, 2012; Sonnenbichler, 2013).

In the following we provide short explanations for each line of the above code:

- Line 2 defines containers which are collectors for similar entities, here for users and activities.
- Line 3 defines a container for access control lists “acls” and assigns to it entities for public, friend-of-a-friend (fof), direct friends, and hidden.
- The idea is, that an activity “act” can be linked to to an entry of the access control list. This relationship is named “acl” (please note the missing trailing ’s’), linking activities from “act” to an access control list classifier from “acls” (line 8).
- Consequently, line 6 defines a relation named “owners” between activities and users,
- line 7 defines a relation named “friends” between users and users.
- Lines 1-8 define the access control model, the lines 10-18 define some example access policies:
- Line 12 creates a policy named “p_owner” allowing an owner of an activity access to the activity. The expression utilizies so-called ADQL one-filtered projections, which we do not introduce here.
- Line 14 allows access for any user to all content classified as public.
- Line 16 defined a policy granting access to all direct friends of the activity owner in the case the activity has been classified in the acl category “friends”.
- Line 18 grants access to all friends-of-friends including direct friends of the activity owner, if the activity has been classified “fof”.

Third, some choices for access control models shall be defined and offered for the users. A simple model shall be chosen as default. This allows in-experienced users to make use of fail-safe defaults while allowing experienced users to use exactly the model appropriate for their requirements. The third requirement should be undertaken by analyzing access control requirements of different Facebook user classes, e.g. as suggested here, companies, privacy-unaware users, privacy-aware users, and paranoid users. Specialized models for these user groups can be defined and offered.

6 Conclusion

Facebook offers basic functionality to let a customer protect his data, e.g. status updates, videos, images, comments and so on. We analyzed the access control model offered by Facebook. We came to the result, that this access control model can be implemented by a simplified access control list (ACL) model: Access rights are omitted and represented only by "read" or "view". Subjects (users) are assigned to the user-specific subject classes "public", "friends of friends", "friends" and "myself".

Security requirements of Facebook may differ a lot based on the type of user. We suggested four user classes:

1. Companies with limited access control requirements offering their content for public access or Facebook "friends".
2. Privacy-unaware consumers who are interested in a simple protection allowing them to share content with everyone or with their friends only.
3. Privacy-aware consumers who want to decide for each of their content elements who shall receive it. The decision should be made on a group level, thus content can be shared for "close friends", "friends", "acquaintances", "public".
4. Paranoid consumers who want to make this decision not on a group level ("this picture may be seen by my friends") but on a detailed level ("this picture may be seen only by users A and B for the next two weeks").

To empower the consumer to let him choose his preferred access control model, three steps have to be undertaken:

1. The access control model must be modularized and not hard-coded.
2. A software component allowing to define the user-specific access control model must be used.
3. Some choices for access control models must be designed and offered for the user.

We sketched how step 2 can be implemented by the Access Definition and Query Language (ADQL) for the current access control model of Facebook.

References

Bell DE, LaPadula LJ (1973) Secure computer systems: Mathematical foundations and model. Mitre Corporation, Bedford, MA, USA 1(M74-244):42, DOI 10.1016/0169-7552(95)00081-X

Bell DE, LaPadula LJ (1975) Secure Computer Systems: Mathematical Foundations and Model. M74-244, Mitre Corporation, Bedford, MA, USA

Benantar M (2006) Access Control Systems. Springer, New York

Graham GS, Denning PJ (1972) Protection: Principles and practice. In: Proceedings of the May 16-18, 1972, Spring Joint Computer Conference, ACM, New York, AFIPS '72 (Spring), pp 417–429, DOI 10.1145/1478873.1478928

Harrison MA, Ruzzo WL, Ullman JD (1976) Protection in operating systems. Communications of the ACM 19(8):461–471, DOI 10.1145/360303.360333

Harrison MH, Ruzzo WL (1978) Monotonic protection systems. In: Demilo R (ed) Foundations of Secure Computations, Academic Press, pp 337–365

Kord JI (2008) Understanding the Facebook Generation: A Study of the Relationship Between Online Social Networking and Academic and Social Integration and Intentions to Re-enroll. ProQuest Information and Learning Company, Ann Arbor, Michigan, USA

Lampson BW (1971) Protection. In: Proceedings of the Fifth Princeton Symposium on Information Sciences and Systems, Princeton University, pp 437–443

Lampson BW (1974) Protection (reprint). ACM SIGOPS Operating Systems Review 8:18–24, DOI 10.1145/775265.775268

Phillips S (2007) A brief history of facebook. The Guardian 2007-07-25

Sonnenbichler A (2013) An Access Definition and Query Language: Towards a Unified Access Control Model. KIT Scientific Publishing, Karlsruhe, Germany

Sonnenbichler AC, Geyer-Schulz A (2012) ADQL: A Flexible Access Definition and Query Language to Define Access Control Models. In: Samarati P (ed) Proceedings of the International Conference on Security and Cryptography 2012, The Institute for Systems and Technologies of Information, Control and Communication (INSTICC), Rome

Vance A (2012) Facebook: The making of 1 billion users. Bloomberg Businessweek 2012-10-04