Visualizing Privacy Implications of Access Control Policies in Social Network Systems

Mohd Anwar[1], Philip W.L. Fong[1], Xue-Dong Yang[2], and Howard Hamilton[2]

[1] Department of Computer Science, University of Calgary, Alberta, Canada
{manwar,pwlfong}@ucalgary.ca
[2] Department of Computer Science, University of Regina, Saskatchewan, Canada
{yang,hamilton}@cs.uregina.ca

Abstract. We hypothesize that, in a Facebook-style social network system, proper visualization of one's extended neighbourhood could help the user understand the privacy implications of her access control policies. However, an unrestricted view of one's extended neighbourhood may compromise the privacy of others. To address this dilemma, we propose a privacy-enhanced visualization tool, which approximates the extended neighbourhood of a user in such a way that policy assessment can still be conducted in a meaningful manner, while the privacy of other users is preserved.

1 Introduction

One of the main purposes of privacy preservation is impression management [1,2]. This is particularly true in the context of social network systems. A profile owner selectively grants a profile viewer access to her profile items in accordance with the impression she wants to convey. For example, say Jill is a friend of Alice, and Bob is a friend of Jill. For proper impression management, Alice may grant Jill, but not Bob, access to her sorority photo album. To check whether her policy allows her to convey the desired impression, Alice may want to look at her profile from the lenses of Bob and Jill, to find out what Bob as well as Jill can see. In our everyday life, we look into a mirror to get a sense of what others see when they look at us. We use the term ***reflective policy assessment*** to refer to this process of assuming the position of a potential accessor for the sake of assessing the privacy implications of access control policies.

Authorization in a social network system is primarily based on the topology of the social graph, which is co-constructed by all the users of the system. It is therefore difficult for a user to mentally keep track of the topology of her constantly changing social network. Furthermore, one's needs for privacy is constantly changing, requiring a user to constantly perform policy assessment. As a result, reflective policy assessment is a nontrivial undertaking. Tool support is definitely desirable.

Unfortunately, a privacy dilemma is inherent in reflective policy assessment. To assess policies reflectively, a user must begin with identifying a potential accessor who is of interest to her. This, however, could lead to breaching the privacy of the potential accessor, as the latter may not want her identity to be disclosed to the user conducting the policy assessment. Suppose the running example is situated in Facebook. If Bob

J. Garcia-Alfaro et al. (Eds.): DPM 2009 and SETOP 2009, LNCS 5939, pp. 106–120, 2010.

adopts a privacy setting that allows his identity to be revealed only to friends but not friends of friends, then Alice will not be able to conduct reflective policy assessment against Bob without breaching his privacy.

This privacy dilemma is not specific to just Facebook. Fong et al. proposed an access control model to delineate the design space of privacy preservation mechanisms in Facebook-style social network systems [3]. In this model, policies such as "only friends" and "friends of friends" are but examples of more general ***topology-based policies***, whereby accessibility is determined by the present topology of the social graph. For example, Alice may adopt the policy that grants access to her sorority photo album only if the accessor shares three common friends with her. With these policies, it would even be more important to have access to one's extended neighbourhood in addition to her immediate friends for the purpose of policy assessment.

This dilemma is rooted in the asymmetric nature of trust. In the process of reflective policy assessment, a resource owner (e.g., Alice) conceptualizes the level of trust she is willing to invest in a potential accessor (e.g., Bob). Yet, this endeavor is possible only if the identity of the potential accessor is known to the resource owner, the feasibility of which may not always be possible because the potential accessor may not trust the resource owner.

This paper is about the design of a privacy enhanced visualization tool for Facebook-style social network systems (FSNSs) to facilitate reflective policy assessment while preserving the privacy of potential accessors. The visualization tool helps a user assess her access control policies by: (a) visually depicting the extended neighbourhood of her social graph and (b) allowing her to inspect her profile from the view point of a potential accessor at her extended neighbourhood. Our contributions are the following:

1. We introduce the notion of reflective policy assessment, which helps a user assess the privacy implications of her policies by positioning herself as a potential accessor. We also discover and address an inherent privacy dilemma of reflective policy assessment.
2. We translate the concept of reflective policy assessment into a concrete visualization tool for policy assessment. Since this tool would not require the knowledge of access control policies of all the users of the system, it can be implemented on the client side (e.g., as a third-party Facebook application).
3. At the core of our visualization technique is a visual representation of a user's extended neighbourhood. We establish graph-theoretic properties common to the social graphs of FSNSs. Based on these properties, we devise an algorithm to generate a surrogate of a user's extended neighbourhood. This surrogate can be examined for reflective policy assessment without violating the privacy of other users.

The organization of this paper is as follows. Sect. 2 describes an access control model for FSNSs. In Sect. 3, we present the main idea of assessing policies through visualization. In Sect. 4, we present an algorithm for generating a surrogate of a user's extended neighbourhood for policy assessment. Sect. 5 discusses subtle issues in our visualization approach. Sect. 6 presents some open questions on how to evaluate the proposed visualization technique. Sect. 7 surveys related literature, and Sect. 8 describes conclusion and future work.

2 An Access Control Model for FSNSs

In this work, we study reflective policy assessment for the family of FSNSs proposed in [3]. Specifically, [3] defines an access control model for social network systems, of which Facebook is but one instantiation. The model generalizes the authorization scheme of Facebook, allowing a more expressive policy vocabulary (see below). We argued in [3, Sect. 5] that careful instantiations of the model can serve as the access control infrastructure of information sharing systems. This section briefly outlines the FSNS access control model so as to anchor the discussion in the sequel. Formal details of this model can be found in [3].

Profile and Profile Items. An FSNS allows each user to construct a representation of him- or herself in the form of a ***profile***. A profile displays such ***profile items*** as personal information, multimedia contents, activity logs, or other user-authored contents. Users may grant one another access to their profile items.

Search Listings. Access to profile items is authorized in two stages (See Fig. 1). In ***Stage I***, the accessor must ***reach*** the ***search listing*** of the profile owner. Then in ***Stage II***, the accessor requests access to the profile, and profile items are selectively displayed. The search listing of a user could be seen as a "capability" [4,5] of the user in the system, through which access is mediated. There are two means by which a profile can be reached in Stage I: ***global name search*** and ***social graph traversal***.

Global Name Search. The first means to reach a search listing is to conduct a global name search. A successful search would produce for the accessor the search listing of the target user. A profile owner may specify a ***search policy*** to allow only a subset of users to be able to reach her search listing through a global name search.

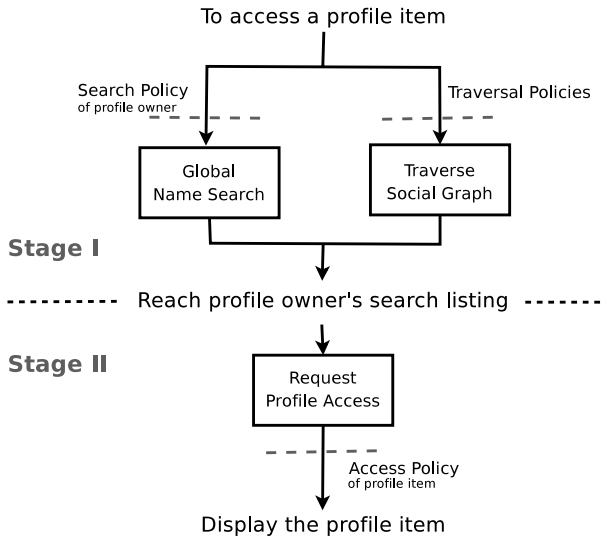

Fig. 1. Authorization procedure for FSNSs

Social Graph Traversal. A second means to reach a search listing is by traversing the ***social graph***. Users can articulate their relationships with one another through the construction of ***friend lists***. Every user may specify a set of other users as her ***friends***. This induces a simple graph in which users are nodes and relationships are edges. A user may traverse this graph by examining the friend lists of other users. More specifically, the friend list of a user is essentially the set of search listings of her friends. A user may restrict traversal by specifying a ***traversal policy***, which specifies the set of users who are allowed to examine her friend list once her search listing is reached.

Profile Access. Once the search listing of a profile owner is reached, the accessor may choose to access the profile, and thereby, initiate Stage II of authorization. Since a profile owner may assign an ***access policy*** to each profile item, not every accessor sees the same profile items when a profile is accessed.

Friendship Articulation. Articulating friendship involves a consent protocol, whereby users interact with one another via a fixed set of ***communication primitives*** (e.g., friendship invitation, accepting an invitation, etc). Once a mutual consent is reached, that friendship is recognized by the FSNS. When a sender initiates a communication primitive against a receiver, the search listing of the latter must be reached before the communication primitive can be initiated. A user can prevent others from initiating a certain communication primitive against her by assigning a ***communication policy*** to that primitive.

Topology-Based Policies. User activities are controlled by user-specified policies (i.e., search, traversal, access and communication policies). Each FSNS offers a fixed policy vocabulary, so that users may adopt policies from the vocabulary to identify sets of privileged users. Since there is no global name space of users, these predefined policies identify user sets by an intensional specification[1]. For example, one may specify that a certain profile item is accessible only by members of the "University of Calgary" network. In [3], we examined a family of intensionally-specified policies known as ***topology-based policies***, which identify privileged users solely in terms of the current topology of the social graph. For instance, one may mandate that a certain profile item is visible only to "friends of friends". We proposed in [3] a number of topology-based policies that are not currently supported by Facebook, but nevertheless possess rich social significance. A sample of these topology-based policies are shown in Fig. 2.

Policy predicate:	*When is access allowed*
$\mathsf{distance}_k$:	distance between owner and accessor is no more than k
clique_k:	owner and accessor belong to the same k-clique (i.e., they belong to the same close-knit group)
$\mathsf{common\text{-}friends}_k$:	owner and accessor share k common friends (i.e., accessor is a known quantity)

Fig. 2. A sample of topology-based policies

[1] An extensional definition specifies a set by enumerating its members (e.g., $S = \{0, 1, 2\}$). An intensional definition specifies a set by stating the characteristic property of its members (e.g., $S = \{x \in \mathbb{N} \mid x < 3\}$).

As we mentioned before, it is cognitively challenging for an FSNS user to understand the privacy implications of adopting a certain topology-based policy. The next section presents a visualization technique that supports reflective policy assessment in the presence of topology-based policies.

3 A Privacy-Enhanced Visualization Technique

A Mirror-based Visualization Technique. Our visualization technique seeks to provide a mirror-like affordance to users in FSNSs. To create a desired impression, we repeatedly look into the mirror and adjust our getup until we are satisfied. A mirror allows us to see what others see when they look at us. The process of formulating access control policies is similar to what it takes to create a desired look. With our ever changing social network and ever changing desire for privacy, a user needs to repeatedly assess and adjust their policies. We propose a mirror-like tool to help a user visualize what others see when they look at her.

Our proposed visualization tool offers the following functionalities to a profile owner.

1. The tool provides a visual representation of an extended neighbourhood of a profile owner in the social graph. The profile owner may specify the size of her extended neighbourhood.
2. This tool allows the profile owner to point to any user in the extended neighbourhood as a potential accessor of her profile. This action signals to the tool that the profile owner intends to position herself as the selected user and examine her profile from the view point of that user.
3. The tool displays a succinct representation of the profile, as seen from the eyes of the potential accessor.
4. The tool suggests potential accessors representing interesting access scenarios (see Sect. 5.2).

This tool contributes to policy assessment in the following ways:

What-if Analysis: It allows a profile owner to perform "what-if" analysis on her access policies. More specifically, it allows her to assess the adequacy of her access policies in concrete access scenarios, and to evaluate the effect of adopting these policies when her extended neighbourhood possess a certain topological structure.

Targeted Effort: As the tool displays how other users are topologically related to a profile owner, it helps her identify topologically interesting nodes in the extended neighbourhood, thereby allowing her to properly target her policy assessment effort. For example, in Fig. 4, the node *FOF* corresponds to an interesting access scenario when the profile owner *Me* attempts to assess a "friends of friends" policy.

Visualizing without Breaching Privacy. The visual representation of the extended neighbourhood must be generated in such a way that the privacy of a potential accessor is preserved. To see this, recall in Sect. 2 that not every potential accessor is reachable from the profile owner, even if there is a path between them. This scenario may arise if at least one of the intermediate nodes along the path has a traversal policy that prevents the profile owner from examining the friend list of that intermediate

node. Consequently, depicting the extended neighbourhood in full accuracy compromises privacy. Fortunately, an accurate rendering of the extended neighbourhood is not necessary for reflective policy assessment. Rather, an approximate rendering that exhibits the topology typical of social networks should suffice. Therefore our approach is to approximate the unreachable region of the extended neighbourhood by generating synthetic nodes and edges in a way that preserves such properties of social networks as power law vertex degree distribution [6] and small-world characteristic [7]. Details of the graph generation algorithm can be found in Sect. 4.

Mockup. In Fig. 3, we show a mockup of our visualization tool. Here, the black node is the profile owner (*Me*). White nodes (e.g., *Jay*) and solid edges (e.g., *Jay-Doe*) depict the interior of the profile owner's reachable region in the social graph. Grey nodes (e.g., *Doe*) mark the boundary (inclusive) of the reachable region. The dotted nodes and dotted edges are generated to approximate the unreachable region of the profile owner. The double-circled dotted or solid nodes are the potential accessors representing interesting access scenarios (as suggested by our tool, see Sect. 5.2 for details). As the profile owner selects a potential accessor by pointing her cursor over the latter, an information box pops up. The information box displays what profile items of the profile owner that the selected user can see as a result of the profile owner's current policies. Specifically, the information box displays three categories of information: (i) the profile

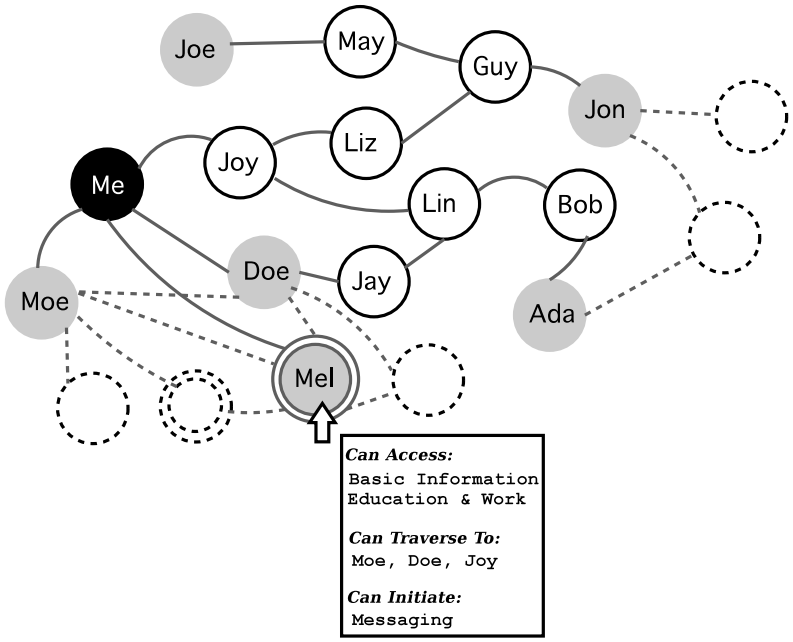

Fig. 3. A prototypical visualization tool to facilitate reflective policy assessment. The black node represents the profile owner. The double-circled node depicts a potential accessor representing an interesting access scenario. When the profile owner points to the potential accessor, Mel, the pop-up box displays a configuration of the profile owner's profile that Mel sees.

items of the profile owner that the selected user can access, (ii) a list of the profile owner's friends that the selected user can reach through the profile owner, and (iii) a list of communication primitives that the selected user can initiate against the profile owner.

Section (i) of the information box is a "reflection" of the profile under assessment. This section supports the assessment of access policies. Section (ii) of the information box supports the assessment of traversal policies. A user's traversal policy has privacy implications not only on the user, but also on her friends. Specifically, an overly relaxed traversal policy will expose one's friends to unwanted accessors. In a similar vein, section (iii) of the information box supports the assessment of one's communication policies.

As an example, in Fig. 3, when the profile owner *Me* points to *Mel*, the tool displays the following: (i) *Mel* can access two profile items of the profile owner: "*Basic Information*" and "*Education and Work*"; (ii) *Mel* can reach *Moe*, *Doe* and *Joe* through *Me*; (iii) *Mel* can send a message to *Me*, but cannot invite *Me* to be a friend.

Assessing Topology-based Policies. A critical reader may question why it is necessary to consider unreachable nodes in the process of reflective policy assessment. We illustrate the utility of this practice by giving some examples. Consider the extended neighbourhood of user *Me* in Fig. 4. We show how various topology-based policies need to be evaluated from the view point of unreachable nodes.

distance$_k$: Suppose user *Me* adopts distance$_5$ as the access policy for her wedding video, thereby granting access to anyone within a distance of five. Let us suppose further that *Jon* is at distance four, whose traversal policy does not allow *Me* to traverse to *Jon*'s friends, including, for example, *D5*. However, user *Me* may precisely want to examine her profile from the perspective of *D5*, which is at distance five from *Me*, in order to evaluate her distance$_5$ policy.

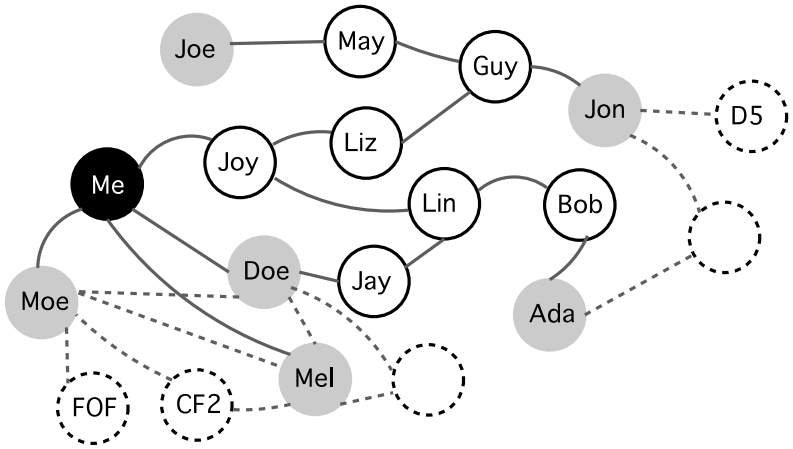

Fig. 4. A visual representation of a profile owner's extended neighbourhood. The black node depicts the profile owner. The grey nodes mark the boundary of the reachable region. The dotted nodes and dotted edges depict the unreachable region of the profile owner's extended neighbourhood.

common-friends$_k$: Suppose the profile owner *Me* specifies common-friends$_3$ as the access policy of her "Contact Information", so that the latter is accessible to those users sharing three common friends with *Me*. According to Fig. 4, users *Me* and *CF2* have only two common friends (*Moe* and *Mel*). It is to the interest of user *Me* to assess her policies reflectively from node *CF2*. Lets suppose Moe and Mel do not allow someone to look at their friends list. Therefore, rendering the node *CF2* would be a breach of *Moe*'s and *Mel*'s privacy. Furthermore, by breaching *Moe*'s and *Mel*'s traversal policy, *CF2*'s privacy is also breached since *CF2* delegates its reachability to *Moe* and *Mel*.

clique$_k$: Suppose user *Me* specifies an access policy, clique$_4$, for her "Status". That is, access is granted to her friends who belong to the same 4-clique as she does. In Fig. 4, users *Me*, *Moe*, *Doe* and *Mel* belong to the same 4-clique. Even though user *Me* needs to confirm that *Moe* and *Doe*, *Doe* and *Mel*, and *Mel* and *Moe* are friends in order to assess her clique$_4$ policy, the traversal policies of *Doe*, *Moe* and *Mel* do not allow the *Me* to discover these relationships.

4 Constructing a Social Graph for Policy Assessment

This section describes an algorithm for generating a visual representation of the social graph for policy assessment. We set the stage by describing some graph-theoretic properties of FSNS social graphs (Sect. 4.1), and then apply the properties to devise the algorithm and establish its correctness (Sect. 4.2).

4.1 Properties of Social Graphs

A node v is u-traversable if the traversal policy of v allows u to examine the friend list of v. If there is a uv-path $uv_1 \ldots v_n v$ in the social graph such that every v_i is u-traversable, then we say user v is ***u*-reachable**. Otherwise, v is ***u*-unreachable**. A u-reachable node is a ***u*-interior node** if it is u-traversable, and a ***u*-fringe node** otherwise. An edge is ***u*-visible** if one of its ends is a u-interior node, otherwise it is ***u*-hidden**. The node u in the above definitions is called the ***origin***. We drop the "u-" prefix when the origin is clear from the context.

Property 1. Given an origin, every neighbour of an interior node is reachable, and thus, no hidden edge can have an interior node as an end.

Property 2. Suppose an origin is given. By definition, at least one end of each visible edge is an interior node. Therefore, no visible edge can join two fringe nodes.

4.2 A Graph Generation Algorithm

We present an algorithm for generating a graph to approximate an extended neighbourhood of a user u in the social graph. The generated graph is composed of two regions. The first region is made up of the reachable nodes and the visible edges. The second region is randomly generated to approximate the unreachable nodes and the hidden

edges of the social graph. To ensure that the randomly generated region reflects the topological structure of a typical social graph, we employ the R-MAT [8] algorithm, which randomly generates graphs exhibiting statistical properties of a real-world social network. (Other appropriate graph generation algorithms can also be used.)

Algorithm. $A(u, M, N)$

1. Using u as the origin, construct a graph consisting of all reachable nodes and visible edges.
2. Temporarily remove all interior nodes and visible edges, leaving only the fringe nodes.
3. Add N "synthetic nodes".
4. Use R-MAT to randomly generate M "synthetic edges".
5. Add back the interior nodes and visible edges removed in step 2, and return the resulting graph.

Algorithm A has three parameters: the origin u, the number N of synthetic nodes to be generated, and the number M of synthetic edges to be added into the graph. We plan to decide on the default value of N and M heuristically based on our forthcoming user study. Step 1 can be achieved by an elementary third-party Facebook application that performs a breadth-first search[2]. This means the algorithm can be executed on the client side, execising no more privileges than the profile owner conducting policy assessment.

The correctness of algorithm A on generating an approximated extended neighbourhood hinges on two conditions. The first correctness condition is that, because synthetic edges are surrogates of hidden edges, the former should only be generated where the latter may occur. By **Property 1**, no hidden edge can have an interior node as an end, and thus synthetic edges should only be generated among fringe nodes and sythetic nodes. This is guaranteed by the removal of interior nodes from consideration in Step 2.

A second correctness condition is that the invocation of R-MAT in Step 4 must begin with an empty graph, so that the statistical properties of R-MAT is preserved. (This condition is not specific to R-MAT, and is necessary even if other graph generation algorithms are used.) By **Property 2**, no visible edge can join two fringe nodes, and thus Step 4 always starts with an empty graph.

5 Issues and Discussion

5.1 Information Leakage

Displaying the profile of a user from the perspective of an accessor may allow the profile owner to infer information about the accessor that is otherwise inaccessible, thereby violating the privacy of the latter. To make the objection concrete, consider the following "attack": Suppose a user u imposes an access policy on a certain profile item o, so that o is visible to someone who belongs to the "University of Calgary" network. Suppose further that user v is a member of that network, but she sets up her access

[2] For example, the third-party Facebook application TouchGraph performs a similar search.

policies so that this fact is not accessible by u. Now, by performing reflective policy assessment from the view point of v, and observing that o is visible to v, u can infer that v belongs to the said network. Thus the privacy of v is breached.

It turns out that information leakage can be prevented by adopting topology-based policies (see Sect. 2 or [3, Sect. 4.2]), so that reflective policy assessment does not leak information that is not already accessible by the user conducting the assessment. With topology-based policies, accessibility is determined solely by the current topology of the social graph. For example, the policies in Fig. 2 are all topology-based. If the FSNS offers only topology-based policies in its policy vocabulary, then mirror-based visualization reveals no other information than the current topology of the social graph. The question then is, does reflective policy assessment disclose topological information that a user does not already possess? The answer is negative. Recall in Sect. 4 that visible edges are already accessible by the profile owner. Hidden edges do not take part in reflective policy assessment. Instead synthesized edges are randomly generated surrogates of hidden edges in reflective policy assessment. Therefore, the topological information that is revealed by reflective policy assessment is either already available (visible edges) or anonymized (synthesized edges). Topological information induced by hidden edges is not revealed at all.

5.2 Recommending Access Scenarios

A feature of our visualization technique is to recommend nodes (potential accessors) that represent interesting access scenarios by highlighting such nodes so that a profile owner can target her policy assessment effort against these potential accessors. In the following we elaborate on what we mean by "interesting access scenarios", and provide additional justifications of our approach.

Once the visualization tool has generated an extended neighbourhood of the profile owner, some nodes are indistinguishable from an access control point of view. More specifically, the appearances of the owner's profile as accessible from the view points of these nodes may be identical. Consequently, there is no need for the profile owner to conduct reflective policy assessment against more than one of these nodes. In short, the various profile appearances partition the nodes into equivalence classes. Each equivalent class represents a distinct ***access scenario***. An access scenario is interesting if it has not been encountered before.

If k distinct topology-based policies are assigned to the items in the profile[3], then there is at most 2^k distinct profile appearances, and thus the same number of distinct access scenarios. To see this, note that what induces a specific profile appearance is the satisfiability of the k policy predicates when a node is given. Consider the following example. Suppose P_1 and P_2 are the policy predicates assigned to the various profile items. Two nodes that both satisfy P_1 but violate P_2 are going to produce the same profile appearance, and thus belong to the same access scenario[4].

[3] The same policy can be assigned to multiple profile items, while certain policies in the policy vocabulary may not be assigned to any profile item at all. Therefore, we do not concern ourselves with the number of profile items or the size of the profile vocabulary.

[4] Note that 2^k is only an upper bound, because some profile appearances are not feasible. For example, no node can violate $\mathsf{distance}_2$ but satisfy clique_4.

Therefore, a tool that supports reflective policy assessment should: (a) help the profile owner identify an enough number of distinct access scenarios (cf. Sect. 6.3) so that the profile owner can have confidence of its privacy settings, and (b) provide a means to describe the individual access scenarios to the profile owner. We intend our visualization tool to track the access scenarios that the profile owner has encountered within a policy assessment session. The tool will selectively highlight a node if it corresponds to a novel access scenario. Multiple extended neighbourhoods can then be generated to help cover commonly occurring access scenarios. Requirement (a) is thus addressed. We also anticipate that the visual depiction of the extended neighbourhood provides an efficient and comprehensible description of access scenarios, thereby addressing requirement (b).

It may appear that since we already know the 2^k access scenarios, there is no need to randomly sample extended neighbourhoods of the profile owner. All we need is to enumerate the 2^k access scenarios, and display the corresponding profile appearances. Unfortunately, this hypothetical solution does not address requirement (b). Recall that a description of the access scenario must be conveyed to the profile owner. We believe that a visual depiction is more effective than a verbal summary of potential access scenarios, such as "$\mathsf{common\text{-}friends}_4$ but not clique_4". Now the question arises whether we should systematically construct visual representation of access scenarios for the policy space of the profile owner. We desire our tool to be indifferent to the specific choice of policy vocabulary. If we are to enumerate all access scenarios, our tool has to do an exhaustive search in the space of all possible social graphs, resulting in exponential time complexity. Instead, our approach of randomly generating the extended neighbourhood can be seen as a Monte Carlo strategy to cope with the intractability of enumerating arbitrary graph-theoretic access scenarios.

6 Open Questions

Our proposal motivates a number of open questions.

6.1 To What Extent Does Our Visualization Technique Facilitate the Assessment of Access Control Policies in FSNSs?

If a tool is effective in supporting policy assessment, we should observe that privacy-aware users tend to formulate a different set of policies after adopting the tool. An empirical user study will help us test if this is indeed the case for our visualization technique. Such a user study shall compare the policies formulated by the user in at least three configurations: (i) no visualization is available, (ii) mirror-based visualization with the rendering of reachable nodes only, (ii) mirror-based visualization with the rendering of both reachable and unreachable nodes.

6.2 How Do We Build a Testbed to Run the Proposed User Study?

A deployed FSNS, such as Facebook, would have been a convenient environment to conduct the proposed user study. There are, however, two problems with this approach.

First, not all topology-based policies are supported in Facebook. As a result, the effectiveness of reflective policy assessment against advanced topology-based policies cannot be gauged. Second, such a study will harvest information of users located in the reachable region of a participant. This setup thus requires consent from a population much larger than the participating group. Even if this aggressive experimental design is approved by the institutional research ethics committee, successfully obtaining consent from such a large population is not likely. We anticipate that the resolution of this problem will involve a clever design of a simulated environment that addresses these privacy challenges.

6.3 To What Extent Are the Randomly Generated Graphs (Sect. 4.2) Useful Approximations of the Unreachable Region of One's Extended Neighbourhood?

We hypothesize that the graphs generated by algorithm A cover topologically interesting access scenarios needed by the profile owner for conducting reflective policy assessment against unreachable nodes. Intuitively, repeated policy assessment on multiple generated graphs should increase the coverage of topologically interesting access scenarios. A natural research question is thus the following: "*how many graphs does one need to generate in order to gain enough confidence on the policies under assessment?*" A probabilistic analysis of this problem is in order.

6.4 How Well Does Our Visualization Tool Facilitate Reflective Policy Assessment in a Very Large Extended Neighbourhood?

It would be burdensome for a user with a very large extended neighbourhood to assess her policies against every access scenario. However the profile owner needs not conduct reflective policy assessment on every node since some of these nodes have the same access privilege to the profile. Our tool groups access scenarios into equivalent classess, and thereby, suggests a distinct access scenario per equivalent class. Additionally, we can apply *focus + context* technique on a hyperbolic plane [9] to effectively render a large neighbourhhod for reflective policy assessment. Using this technique, we want to assign more display space to some interesting access scenarios (to render greater focus), while still embedding the focused access scenarios into the context of entire neighbourhood. A profile owner can easily move her mouse pointer to focus on a different part of her extended neighbourhood and perform policy assessment against different access scenarios.

7 Related Works

Assessing the security implications of access control policies traditionally lies in the domain of safety analysis [10,11], or, more recently, security analysis [12,13]. When the projection of security implications becomes a challenging computational problem, safety or security analyses are indispensable. While appreciating the scope and analytical rigor of such approaches, this paper seeks to address the *cognitive challenges* of

users in the projection of the *privacy implications* of their access control policies. A visualization tool can reduce the cognitive load of users in policy assessment. It is also a better fit with the requirements of impression management.

Visualization techniques have long been used in social network analysis [14]. With the soaring popularity of online social networks, visualization techniques are widely used to empower users of such networks. For example, Heer & boyd employed visualization techniques for exploration and navigation of large-scale online social networks [15]. Facebook offers a profile owner to see how a friend sees her profile[5]. Reeder et al. proposed a visualization technique to support authoring of security policies [16], whereby the access control matrix is rendered as an expandable grid representation. Ours and Reeder et al.'s work share a common underpinning of visualizing authorization decisions under the assumption of some security policies. In our work, when a profile is displayed from the view point of a potential accessor, we are essentially rendering a segment of the row in access control matrix corresponding to that accessor. Our work is distinct from their work on two counts: (i) our work is tailored for the assessment of topology-based access control policies in the context of social network systems, and (ii) we are concerned with preserving the privacy of potential accessors. To the best of our knowledge, we are the first to propose visualization of social network for access control policy assessment. Our proposed visualization technique supports impression management for a family of FSNSs. This family was defined by Fong et al. [3], who formally specify an access control model that delineates the design space of social network systems employing the same access control paradigm as Facebook. A distinctive feature of FSNSs is that no global name space is available for identifying users, and thus access control policies are specified in terms of the present topology of the social graph. This element of distributed access control causes policy assessment to be a nontrivial undertaking, thereby necessitating our visualization technique. Furthermore, Fong et al. formulated some policies that are purely based on topological information: e.g., Degree of Separation, Known Quantity, Clique, etc.

A number of recent proposals attempt to advance beyond the access control mechanisms found in commercial social network systems. A notable example is that of Carminati et al., in which a decentralized social network system with relationship types, trust metrics and degree-of-separation policies is developed [17,18,19,20,21]. An interesting research issue is to design tools that support reflective policy assessment in these next-generation social network systems.

8 Conclusion and Future Work

We anticipate that our visualization technique can reduce users' cognitive load in understanding the privacy implications of their access control policies in a FSNS. Specifically, this visualization technique helps a profile owner assess her policies by displaying how potential accessors are topologically related to her in an extended neighbourhood, and allowing her to visually assess her policies via a mirror-like facility from the perspective of a potential accessor of her choice. This technique supports the reflective assessment

[5] http://www.facebook.com/privacy/?view=profile

of access, traversal and communication policies in FSNSs. We plan to conduct an empirical user study to gauge the effectiveness of this visualization technique. We also plan to address the theoritical question of figuring out the number of graph samples needed for inducing confidence on reflective policy assessment.

References

1. Goffman, E.: The Presentation of Self in Everyday Life. Anchor-Doubleday, New York (1961)
2. Patil, S., Kobsa, A.: Privacy as impression management. Technical Report UCI-ISR-03-13, Institute for Software Research, University of California - Irvine, Irvine, CA, USA (December 2003)
3. Fong, P.W.L., Anwar, M., Zhao, Z.: A privacy preservation model for Facebook-style social network systems. In: Proceedings of the 14th European Symposium on Research in Computer Security (ESORICS 2009), Saint Malo, France (September 2009)
4. Dennis, J.B., van Horn, E.C.: Programming semantics for multiprogrammed computations. Communications of the ACM 9(3), 143–155 (1966)
5. Miller, M.S., Yee, K.P., Shapiro, J.: Capability myths demolished. Technical Report SRL2003-02, System Research Lab, Department of Computer Science, The John Hopkins University, Baltimore, Maryland, USA (2003)
6. Faloutsos, M., Faloutsos, P., Faloutsos, C.: On power-law relationships of the internet topology. In: Proceedings of ACM Special Interest Group on Data Communications (SIGCOMM 1999), pp. 251–262 (1999)
7. Milgram, S.: The small world problem. Psychology Today 1, 60–67 (1967)
8. Chakrabarti, D., Faloutsos, C., Zhan, Y.: Visualization of large networks with min-cut plots, A-plots and R-MAT. International Journal of Human-Computer Studies 65, 434–445 (2007)
9. Lamping, J., Rao, R.: The hyperbolic browser: A focus+context technique for visualizing large hierarchies. Journal of Visual Languages and Computing 7(1), 33–35 (1996)
10. Harrison, M.A., Ruzzo, W.L., Ullman, J.D.: Protection in operating systems. Communications of the ACM 19(8), 461–471 (1976)
11. Lipton, R.J., Snyder, L.: A linear time algorithm for deciding subject security. Journal of the ACM 24(3), 455–464 (1977)
12. Li, N., Winsborough, W.H., Mitchell, J.C.: Beyond proof-of-compliance: Safety and availability analysis in trust management. In: Proceedings of the 2003 IEEE Symposium on Security and Privacy, pp. 123–139 (2003)
13. Li, N., Tripunitara, M.V.: Security analysis in role-based access control. In: Ninth ACM Symposium on Access Control Models and Technologies (SACMAT 2004), pp. 126–135 (2004)
14. Freeman, L.C.: Visualizing social networks. Journal of Social Structure 1(1) (2000)
15. Heer, J., boyd, d.: Vizster: Visualizing online social networks. In: Proceeding of IEEE Symposium on Information Visualization, pp. 33–40 (2005)
16. Reeder, R.W., Bauer, L., Cranor, L.F., Reiter, M.K., Bacon, K., How, K., Strong, H.: Expandable grids for visualizing and authoring computer security policies. In: Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems (CHI 2008), pp. 1473–1482. ACM, New York (2008)
17. Carminati, B., Ferrari, E., Perego, A.: Rule-based access control for social networks. In: Meersman, R., Tari, Z., Herrero, P. (eds.) OTM 2006 Workshops. LNCS, vol. 4278, pp. 1734–1744. Springer, Heidelberg (2006)

18. Carminati, B., Ferrari, E., Perego, A.: Private relationships in social networks. In: Proceedings of Workshops in Conjunction with the International Conference on Data Engineering – ICDE 2007, Istanbul, Turkey, pp. 163–171. Springer, Heidelberg (2007)
19. Carminati, B., Ferrari, E.: Privacy-aware collaborative access control in web-based social networks. In: Atluri, V. (ed.) DAS 2008. LNCS, vol. 5094, pp. 81–96. Springer, Heidelberg (2008)
20. Carminati, B., Ferrari, E., Perego, A.: Enforcing access control in web-based social networks. ACM Transactions on Information and System Security (to appear, 2009)
21. Carminati, B., Ferrari, E., Heatherly, R., Kantarcioglu, M., Thuraisingham, B.: A semantic web based framework for social network access control. In: SACMAT 2009: Proceedings of the 14th ACM symposium on Access control models and technologies, pp. 177–186. ACM, New York (2009)