Inferring Privacy Policies for Social Networking Services

(Position Paper)

George Danezis
Microsoft Research, Cambridge
7 J J Thomson Avenue, Cambridge, CB3 0FB, U.K.
gdane@microsoft.com

ABSTRACT

Social networking sites have come under criticism for their poor privacy protection track record. Yet, there is an inherent difficulty in deciding which principals should have access to user's information or actions, without requiring them to constantly manage their privacy settings. We propose to extract automatically such privacy settings, based on the policy that information produced within a social context should remain in that social context, both to ensure privacy as well as maximising utility. A machine learning approach is used to extract automatically such social contexts, as well as a tentative evaluation.

Categories and Subject Descriptors: K.4.1 [Computers and Society]: Public policy issues – privacy;

General Terms: Security

Keywords: Social networks, privacy, cohesive groups

1. INTRODUCTION

Social networking sites have recently become very popular. Facebook[1], the largest one, has more than 100 million user accounts registered. Yet, social networking services are increasingly criticised for failing to adequately protect their users' privacy [8].

These privacy failures can be attributed to incidental and intrinsic reasons. Social networking services have been very fast growing, and their limited engineering resources have focused on scaling their platforms, instead of adapting them to the changing usage patterns. For examples, the default privacy policy on facebook that restricted who can view a user's profile by network, made sense when users were mostly within universities. It fails once anyone can join regional networks, and users accept contacts from all walks of life. Privacy bugs, that for examples, extracted deleted photos, also reflect rushed engineering, with scalability being the key priority.

[1] `http://www.facebook.com`

AISec'09, November 9, 2009, Chicago, Illinois, USA.

Besides these incidental problems ensuring a privacy friendly experience for social networking sites would require setting a default policy, about who can see what, which is compatible with users' expectations. It has been argued before [6] that simply providing fine grained controls that allow users to set their preferences is not sufficient to support privacy. Users find the task of specifying who should access each new piece of content tiresome in practice. That is true even for users that declare they would spend the time – as the investment people are ready to make in practice for privacy is minimal [1]. Even if users were willing to put the effort to specify their policies, it is not clear that they would be able to do so safely. Creating security and privacy policies is an end-user programming exercise [9], that requires specialists to apply tools close to programming languages like SecPAL [4].

As a result, a recent proposal [6] is for pre-packaged privacy policies, designed by experts, to be made available, for users to chose whichever suits them best. Following this trend, our position is that *specifying privacy policies has to be sensitive to the social context in which content is generated, and furthermore this context has to be automatically inferred by the privacy policy mechanism.* We provide a short overview on privacy and context; then we argue that identity management approaches proposed in the literature are too static and cumbersome to really help users protect their privacy; finally, we propose a specific privacy mechanism for social networking based on context inference through social network analysis, that we evaluate briefly.

2. PRIVACY, CONTEXT AND THE CHALLENGE OF SOCIAL NETWORKING

The idea that privacy is maintained when information stays within the proper context is an old one. The EU data protection framework [12], that forms the basis for the national data protection legislation throughout the EU member states, puts forward eight key principles to keep personal information within a proper context. In particular it specifies that personal information collected should be processed for a specific purpose to which the user has consented, and only for that purpose. This principle makes it clear that sharing information does not mean that it becomes available to everyone and for any purpose, clearly bounding the context in which it can be used lawfully.

Similar ideas were developed by the philosopher Helen Nissenbaum, that introduced the concept of *contextual integrity* as the essence of privacy [3]. The privacy mechanism derived from this concept, ensure that all personal informa-

tion is labeled with a specific context and the access control logic ensures that personal information never crosses contexts in an identifiable form, and never loses its context. The idea of Hippocratic databases push this approach to its logical extreme [2] with particular applications to the health care sector.

These approaches are important, but apply to the relationship of a user as a data subject, with an institution or otherwise large entity that has a particular need for personal information to provide a service. These institutions are in a very good position to define proper privacy policies for their services, and regulations like HIPPA or the EU data protection regimes require them to do so. Yet, these rigid approaches fail in social networking sites, since the context of interactions is not explicit, and multiple contexts may coexist within the same system. It is very common for users to create contacts within those sites that are friends, family, colleagues, or people within shared communities of interest. These contexts are independent from the social network service provider (who might also have certain contexts, like advertisers), and the service has a-priori no idea what context user-generated information belongs to, or even what contexts exist within the system.

User-centric identity management [13] approaches were proposed partly to solve this problem. They give the user control over what information is shared with other parties, and allow the user to make decisions about the context of each transaction. Special software ensures that credentials with personal information are only provided to those that are authorised. Users are free to negotiate privacy preferences [5], and may be able to chose services according to the information they demand, or how they process it. Despite the promise of flexibility, such systems place a burden on the users: they have to make judgements for each interaction, and constantly make decisions about the context of each transaction and the items of information they should be divulging. The user experience of interacting with others would be severely degraded if such an approach was to mediate all exchanges of information within social networking sites[2].

To summarise, current approaches require at best personal information, principals, and interactions to be manually labeled with a context, or at worse require the user to make an ad-hoc access control decision for every interaction. This might be tolerable for extremely formalised interactions, with the government or a bank, where the privacy policies remain relatively static and the contexts are well delineated. In the context of social networking sites, where context is implicit, ever changing, and not a-priori known to the service provider, requiring the user to manually label all interactions or authorise them individually becomes a usability nightmare. As a result even if privacy controls exist in this form, they are unlikely to be used.

3. SOCIAL NETWORKING IN CONTEXT

We assume a model of a social networking site in which the service provider is trusted. As we will see this is not a major restriction since our approach is applicable to peer-to-peer social networking, as well as single provider centralised services. Our concern is how to decide which of the users of the system is allowed to see information generated as part of an interaction of other users. For example, if Alice comments on an item that has been published by Bob; who is allowed to see this comment? Similarly, if Charlie "tags" Diane in a photograph he published; who is allowed to see the photograph and the "tag"?

Current approaches to solving this question are rather coarse grained: users can usually make interactions private, restrict them to all their contacts, or their full network, or make them public. This involves respectively no-one, hundreds, thousands or millions of people being able to observe an action. Any finer grained control requires manual setup of an access control list per item or interaction.

Our approach aims to support users when they wish to restrict the visibility of their interactions to a smaller subset of their contacts. For most interactions we aim to infer the context within which the interaction should stay, and prevent users that are outside that context from seeing it. Concretely we partly rely on inference algorithms to solve the following tasks:

1. **Infer user contexts.** For each user, and based on the social graph around them only, we infer a set of possible contexts within which actions may occur. We classify the contacts of each user as belonging to one or more contexts, vis-a-vis the user.

2. **Context assignment.** Each interaction has to be assigned to one or multiple extracted contexts automatically, or with minimal help from the user. Hints, such as the two parties interacting, or the people that are "tagged" as part of an action can be used to facilitate this.

3. **Privacy policy per context.** Once information resulting from interactions has been assigned a context, the default visibility for that context can be assigned to it. A safe default would be only for contacts of users in the context of an action to be able to see the resulting information.

The idea of inferring access control groups, which is related to our first inference task, has already been applied to extract roles for Role Based Access Control policies [14]. The algorithms proposed for role mining involve some supervised learning, where sets of permissions-users assignments are provided, and roles are extracted. In the context of social networking we cannot assume the user will have the patience to provide examples of context-user-interaction assignments, and thus we have to restrict ourselves to unsupervised learning approaches to extract those contexts.

The principle that contexts, and thus privacy sensitive decisions, are extracted automatically might be seen as contrary to the concept of informed consent that is at the heart of privacy technology. As we have argued, informed consent implemented by harassing the user and forcing them to micro-manage their preferences is ineffective. Our inference approach can be compatible with the principle of consent, by ensuring that the contexts and access decisions inferred are transparent, and modifiable by the users. Both the notification and the ability to tweak the inferred context should be unobtrusive, and should not interrupt the task of the

[2]The reaction to the Microsoft Vista User Account Control mechanism, that prompts for authorisation and confirmation, shows that users have limited patience for setting security policies, particularly when action is required with every interaction.

user – a break from current privacy practices. While neither inference tasks need to be perfect, they should be reliable enough for users not to have to tweak the contexts manually most of the time.

3.1 Extracting contexts

The key philosophical question we need to address is "what constitutes a privacy context?" This is a burning question, since we need to automatically infer a set of contexts in which actions are performed, without any labeled user decisions. Since we are concerned with privacy we define a context as *a set of contacts of a user, that are closely related to each other, in such a way, that one would expect information about the user's interactions with one of them to become known to the others, independently of the social networking site.* Conversely, information about actions is less likely to propagate between contexts outside the social networking system. Intuitively, we aim to make the policy within the social networking service reflect whichever social reality users face about their private information in the "real world", outside the service.

A number of sources of information could be used to infer those contexts. For example we could use explicit information flows, or any other proxy for real-world social interaction to detect how likely information transfer is outside the system, and cluster contacts in groups accordingly. Since we are designing a privacy policy engine, we cannot assume that those rich flows of information are available to the user and the inference algorithm to infer contexts. In particular to infer flows of information within groups we need to have information about the interactions amongst third parties–which are exactly the interactions we may want to hide through the privacy policy. An unsatisfactory solution would require a trusted entity to compute the contexts of each user. We would then be concerned about inference control [11], and the amount of private information leaked.

To ensure that the context extraction is as privacy friendly as possible we use only information about the social graph directly around the target user whose contexts we infer. Specifically we require only the list of the user's contacts and information about which of these contacts consider each other a contact. We exclude any information about users that are not contacts of the target user, so we effectively limit our algorithm to the sub-graph of diameter between one and two around the target user. Formally, we use all loops of length three, starting and ending at the target user. Such information is available, or easy to extract from current social networking sites [7].

To support our context inference we assume that users forming links in that sub-graph is an indication that they have a real-world relationship, and that information would be likely to flow on those links independently of the service. Detecting contexts thus becomes a problem of inferring cohesive groups of users, i.e. groups that have many links within them, while having fewer links with non-members of the group. Social network analysis, a sub-field of sociology, provides many definitions for such cohesive subgroups, such as cliques and k-plexes [15]. Since those graph theoretic definition are a bit rigid we introduced the related concept of a (K, δ, γ)-group based on high-density sub-graphs:

Definition 1. Density of a sub-graph. The *density* of a sub-graph is the number of actual edges between the vertices belonging to the sub-graph, divided by the maximum number of distinct edges that could exist between all vertices (excluding self-loops). It is a normalised measure of the fraction of edges present in a sub-graph and takes values in $[0, 1]$.

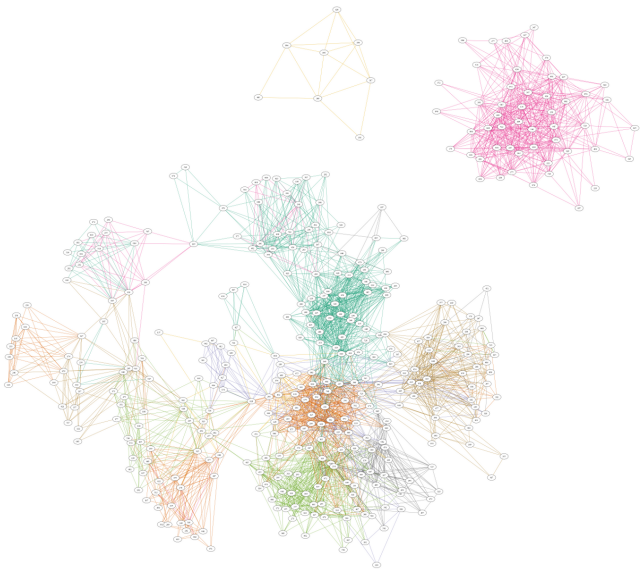

Figure 1: Inferred contexts as $(4, 0.95, 0.33)$-groups of a user. Out of 550 contacts 393 are classified in a context, indicated by different colors of links. (Only links within contexts are plotted.)

Definition 2. (K, γ, δ)-group. A (K, γ, δ)-group is a sub-graph A of density at least δ, composed of vertices each belonging to some sub-graph $B \subset A$, where B is of size at least K, and density at least γ. We require the densities of the sub-graphs B to be greater than the density of A, i.e. $\gamma > \delta$.

Our definition allows for large groups of a moderate density (δ), that are composed of smaller (at least size K) groups of a very high density (γ). In practice we use parameters like $K > 3$, $\gamma > 0.95$ and $\delta > 0.3$ meaning that we expect the majority of links in the context to be present, and groups to be composed of small sub-groups of extremely high density, where nearly all links are present.

We extract a set of (K, γ, δ)-groups to use as contexts through a greedy algorithm. First we visit all pairs of contacts that have a link between them and detect if they are part of a trivial small sub-group of size K and density γ. This simply involves intersecting their respective contact sets and checking the resulting sub-group's size and density. In a second phase we cluster these small, very cohesive, groups, into larger groups while ensuring that the target density of the larger groups never goes below δ. Finally, we check whether contacts that have not been assigned to a context can be merged into the larger groups, without violating the invariants.

Figure 1 illustrates the context extraction procedure using $(4, 0.95, 0.33)$-groups. The target user has 550 contacts, and 393 of them were assigned to at least one context (only 516

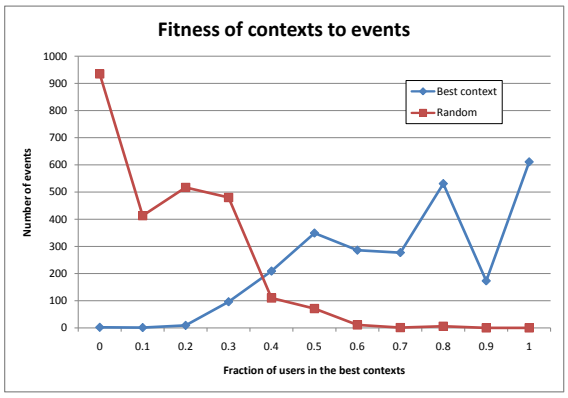

Figure 2: The fraction of event participants that are contained in the best context for the event, versus a random group of the same size. (2544 events were used.)

of the users had another contact in the region around the target user). In total 31 contexts are extracted, the largest one composed of 57 users, and the smallest of 7 users. On average each context contains about 26 users (median 22 users). We note that the density of the sub-graph composed of all contacts of the target user is 0.034, about an order of magnitude smaller than the density $\delta > 0.33$ we require to accept contexts.

3.2 Evaluating contexts

Evaluating the fitness of the contexts extracted is a hard problem. One approach would be to present the contexts to users, and require feedback about their quality. We used feedback from a few users to tune the parameters (K, γ, δ) used to extract the models. In these cases users were able to easily assign a semantic context to each group, an important consideration if such groups were to be used for access control. The extracted groups tend to be conservative, and they omit users "at the boundaries" between groups. This might be a usability problem, but it ensures that only users deeply embedded within a context receive information within that context. The key challenge, as for any privacy study, is that users may simply state that they would be happy with those contexts, but actually not like them and use them in practice. For this reason there is a need for a more rigorous evaluation of the fitness of the extracted contexts and potential parameters for their extraction.

To perform a more rigorous evaluation of the quality of the extracted contexts we used Facebook "events". These are not simply groups of interest, but correspond often to physical events that users can attend. Events can give us some insight about the social structure outside the social networking site, and help us evaluate whether extracted contexts match this real-world structure.

For the target user, we chose 2544 events at random that their or their contacts were subscribed to (we required at least four contacts of the target to be subscribed to each event). For each of these events, we extract the list of contacts of the target user that are subscribed. Our hypothesis is that if the contexts extracted are good, they will closely match the groups of people attending an event. Otherwise, if they are poor, they should match these events no better than a random group of a similar size. To evaluate our hypothesis we find the best context for each of the events, by selecting the one with the largest overlap, and observe the fraction of contacts in the event, that are within that context.

Figure 2 illustrates our results. The red line indicates the number of events for each fraction of overlap for random groups. As expected, the overlap is small, and usually no more than 50%, with the vast majority of overlap being equal or less than 30%. The blue line indicates the fraction of user contacts contained within the best context. As we observe, the overlap is significant: on average more than 70% – 80% of contacts are contained in the best context. More than 600 events contain participants that are totally covered by the appropriate best context. It is quite likely that aggregating the two best contexts would exceed this.

Given those encouraging results we are confident that the bulk of context extraction can be done automatically, and yield high quality inferences. Refining the definitions of contexts, under the light of social network analysis and cohesive sub-group definitions, to improve the quality of the inference is likely to continue.

3.3 Assigning contexts to interactions

Interactions within a social networking site naturally offer strong hints as to the context they belong. We have already examined Facebook events that have guest lists. A standard policy might be that any information concerning the user's participation to that event should only be shared within the context with the highest or a minimal overlap of participants.

Many other functions also provide natural groups of users that can be used to match them to a context: photo albums are tagged with people featured in photographs, notes and links have recipients, comments belong to a thread, notes on each other's spaces contain at least two participants. It is an open research problem how well each type of interaction can be automatically categorised. In case it cannot, a hint from the user, in the form of a couple of intended recipients would be sufficient to provide the missing context. There if no need, as it is traditional, to provide extensive access control lists built from scratch in an ad-hoc manner for each interaction.

Positive feedback mechanisms can also be provided to reinforce any algorithms ability to assign a context to interactions. Since contexts ideally delineate communities, and have some semantic meaning, they could be used as groups to support actions within the social networking service. For example, all contacts from a particular context may be invited to a work party, or sent a card for Christmas.

4. CONCLUSION

We have argued in this position paper, that traditional ways of conceiving privacy are inapplicable, cumbersome or unusable in the context of social networking sites, where interactions between people are frequent and context is not explicitly labelled to apply a policy. Instead, we propose to extract automatically social contexts within which personal information should reside, using only the social structure around each user. Given the minimal amount of informa-

tion required, the context inference can be performed by the users themselves in a peer-to-peer social network, or a centralised service. We evaluated a preliminary context extraction algorithm, based on highly cohesive social sub-groups, and argued it does a good job of describing the underlying real-world social groups.

Despite these results we expect the approach proposed to be controversial. Automatically extracting user's preferences when it comes to security and privacy could be seen as removing user's autonomy, and may open the system to privacy attacks. The problem of consent and autonomy can be managed through a high degree of transparency. The security of the scheme is based on the infiltration resistance of the units composing contexts: it would be expensive for an adversary to infiltrate a large context, since they would have to fool many of its participants to maintain a high density. A similar security argument was recently used as part of sybil defence schemes [10].

Ultimately, the user acceptance of such a scheme will depend on the quality of the inference of contexts and assignments to contexts. If users come to trust the automated schemes, as we trust today search engines to deliver the content we expect, this will be a significant step forward to achieve privacy on-line.

Acknowledgements. The authors would like to thank Joseph Bonneau and Luke Church, from the University of Cambridge Computer laboratory, for discussions as well as scientific and philosophical disagreements that motivated this position paper. Emilia Käsper and Parka Poser provided valuable feedback on the quality of the sub-group extraction.

5. REFERENCES

[1] Alessandro Acquisti and Ralph Gross. Imagined communities: Awareness, information sharing, and privacy on the facebook. In George Danezis and Philippe Golle, editors, *Privacy Enhancing Technologies*, volume 4258 of *Lecture Notes in Computer Science*, pages 36–58. Springer, 2006.

[2] Rakesh Agrawal, Roberto J. Bayardo Jr., Christos Faloutsos, Jerry Kiernan, Ralf Rantzau, and Ramakrishnan Srikant. Auditing compliance with a hippocratic database. In Mario A. Nascimento, M. Tamer Özsu, Donald Kossmann, Renée J. Miller, José A. Blakeley, and K. Bernhard Schiefer, editors, *VLDB*, pages 516–527. Morgan Kaufmann, 2004.

[3] Adam Barth, Anupam Datta, John C. Mitchell, and Helen Nissenbaum. Privacy and contextual integrity: Framework and applications. In *IEEE Symposium on Security and Privacy*, pages 184–198. IEEE Computer Society, 2006.

[4] Moritz Y. Becker, Cédric Fournet, and Andrew D. Gordon. Design and semantics of a decentralized authorization language. In *CSF*, pages 3–15. IEEE Computer Society, 2007.

[5] Abhilasha Bhargav-Spantzel, Anna Cinzia Squicciarini, and Elisa Bertino. Trust negotiation in identity management. *IEEE Security & Privacy*, 5(2):55–63, 2007.

[6] Joseph Bonneau, Jonathan Anderson, and Luke Church. Privacy suites: Shared privacy for social networks. In *SOUPS 2009: Symposium On Usable Privacy and Security*, Mountain View, CA, USA, July 15 2009.

[7] Joseph Bonneau, Jonathan Anderson, and George Danezis. Prying data out of a social network. In *ASONAM 2009: The 2009 International Conference on Social Network Analysis and Mining*, Athens, Greece, July 20 2009.

[8] Joseph Bonneau and Soren Preibusch. The privacy jungle: On the market for data protection in social networks. In *The Eighth Workshop on the Economics of Information Security (WEIS 2009)*, June 2009.

[9] Luke Church, Jonathan Anderson, Joseph Bonneau, and Frank Stajano. Privacy stories: Confidence in privacy behaviors through end user programming. In *SOUPS 2009: Symposium On Usable Privacy and Security*, Mountain View, CA, USA, July 15 2009.

[10] George Danezis and Prateek Mittal. Sybilinfer: Detecting sybil nodes using social networks. In *NDSS*. The Internet Society, 2009.

[11] S. C. Hansen and E. A. Unger. An extended memoryless inference control method: Accounting for dependence in table-level controls. In James Clifford and Roger King, editors, *Proceedings of the 1991 ACM SIGMOD International Conference on Management of Data, Denver, Colorado, May 29-31, 1991*, pages 348–356. ACM Press, 1991.

[12] Eleni Kosta, Martin Meints, Marit Hansen, and Mark Gasson. An analysis of security and privacy issues relating to rfid enabled epassports. In Hein S. Venter, Mariki M. Eloff, Les Labuschagne, Jan H. P. Eloff, and Rossouw von Solms, editors, *SEC*, volume 232 of *IFIP*, pages 467–472. Springer, 2007.

[13] Birgit Pfitzmann. Privacy in enterprise identity federation - policies for liberty single signon. In Roger Dingledine, editor, *Privacy Enhancing Technologies*, volume 2760 of *Lecture Notes in Computer Science*, pages 189–204. Springer, 2003.

[14] Jaideep Vaidya, Vijayalakshmi Atluri, and Qi Guo. The role mining problem: finding a minimal descriptive set of roles. In Volkmar Lotz and Bhavani M. Thuraisingham, editors, *SACMAT*, pages 175–184. ACM, 2007.

[15] Stanley Wasserman, Katherine Faust, and Dawn Iacobucci. *Social Network Analysis : Methods and Applications (Structural Analysis in the Social Sciences)*. Cambridge University Press, November 1994.