Trust and Privacy in the di.me Userware

Marcel Heupel, Mohamed Bourimi, and Doğan Kesdoğan

Institute for Information Systems, University of Siegen, Germany
heupel@wiwi.uni-siegen.de

Abstract. People in online social networks are constantly sharing information with different audiences and it becomes more and more difficult for them to keep track of distributed information. Further, due to the complexity of the digital landscape, it is a constant risk to unintentionally share information to the wrong people, possibly leading to a loss of reputation. The European research project di.me is concerned with the development of a userware, empowering end-users to keep track of their digital footprints, with an intelligent user interfaces (UI) and smart advisory. In this paper we present how we calculate persons trust and inform the privacy of resources shared among persons. We show the concepts for trust and privacy advisory in the di.me userware and address problems, we were confronted within the design and evaluation process and how we tackled them. In this respect we specifically address change requirements (i.e. trust model and UI improvements) we conducted after an evaluation and user trials with a first prototype.

Keywords: Trust, privacy, user interface, privacy advisory, online social networks, di.me.

1 Introduction

In todays online social life people are constantly sharing information. It is not easy for end-users to keep track of all their information, distributed over different online social networks (OSNs). When in the beginning of OSNs some years ago most information posted was public, the privacy awareness of people raised over the last years. Nowadays, for most applications it is common practice to configure the visibility of posted information with the help of security settings or privacy preferences in the user interface (UI).

The strong interplay of security and usability, more precisely, the fact that usability is an important prerequisite for secure systems, is getting more and more attention recently (see e.g., [1]) and it was already mentioned by Kerckhoff in 1883 [2] in his six design principles for military ciphers. This is even more crucial as user experience (UX) and usability have also consequences for privacy and trust, as it is well known that usability and UX are important factors for trust [3]. A recent study of Madejski et al. [4] reported, almost all people make errors when setting their privacy preferences in OSNs. This leads in the end to information being shared with wrong audiences without intention and can cause serious damage to the personal reputation in extreme cases. The European

M. Kurosu (Ed.): Human-Computer Interaction, Part III, HCII 2013, LNCS 8006, pp. 39–48, 2013.

research project di.me is developing a tool, integrating all information from connected OSNs in a single-user controlled information sphere. Key functionalities in this respect are the provision of an privacy-preserving intelligent UI and the provision of smart advisory.

In this paper we will present our approach providing smart trust and privacy advisory when sharing information in OSNs. The approach consists of a trust metric, bringing together the privacy of information and trust of persons, which was already presented in [5] and several UI concepts in order to adequately present the advisory (based on that metric) to the user in an intuitive and non-intrusive way. The first version of the metric, as well as corresponding UI concepts, has been evaluated in first prototypes of the di.me userware. Main focus of this paper is the presentation of the current status of work as well as addressed change requirements (CRs) we identified in the evaluation.

The paper is structured as follows: In Section 2 we elaborate key concepts for trust and privacy in di.me and put them in relation to related work. In Section 3 we provide some essential background information about the di.me userware. In Section 4 we present our identified CRs, which we address in our approach, discussed in Section 5. Finally we conclude the paper in Section 6.

2 Definition of Trust in di.me and Related Work

Trust and privacy advisory is one of the key functionalities of the di.me userware, so it was of major importance to find a common understanding for those terms, and to develop adequate design concepts for the UI to prevent misunderstandings on the side of the end-users. Since it is not easy to give a short definition of trust, it is even harder to present this in an intuitive way to the user. The concept of trust is difficult to perceive, and the vast amount of definitions make it even harder. Therefore, we will expound here the concept of trust as we use it in di.me. In the past, there has been a lot of research in the field of trust, and there are also a lot of different definitions. When talking about trust, many people think about trust establishment with unknown parties (e.g., [6] and many more). This is strongly related to reputation, since to the decision to trust an entity or not is based on the public opinion. There are a lot of papers concerned with trust in social networks. Usually they talk about trust in the OSN provider itself or about trust in unknown entities (e.g., when adding new contacts etc.), which is also more going in the direction of reputation (*to trust someome because of good reputation*, see also [7]).

The di.me userware deals with user-centric information, contact management, and it supports the user by interacting with already existing contacts in different OSNs. Since the establishment of new connections is of minor relevance in the context of online social networks[1]. Therefore, there is no concept of reputation in di.me, not even a concept for friends-of-friends [8]. Consequently the concept of trust in di.me differs from the previously mentioned definitions. In general the

[1] We assume most people do not consider reputation to decide if they connect to a person in a social network or not.

di.me concept of trust can be defined like the following, which is very similar to Josangs [7] definition of *Reliability trust*. To trust someone means that the other party behaves as expected with a specific probability. This is of course always connected to a certain context. In di.me we are more or less only moving in the context of information disclosure, which means basically that trusted persons will give private information with a very low probability to (untrusted) third parties. Trust in di.me tries to measure the personal direct trust of the user to each of his contacts. It is formed by the interaction between individuals (e.g., communication, sharing) and can consequently also be seen as measure for the strength of relationship. Sharing private information to a person is an expression of trust. As described in [5], we compute the trust based on previous interactions, especially the disclosure of information. The more private the disclosed information is, and the smaller the audience, the higher is the resulting trust value. Related to privacy in OSNs many works deal with supporting the user in privacy preferences (see e.g., [4,9,10]), which are rather static. Our approach, of giving dynamic trust based privacy advisory on runtime, is rather novel and, to our best knowledge, not covered by other works.

3 Background Information

3.1 The di.me Project

The di.me project targets the integration of personal information in a personal information sphere, in order to give users full control over their data and allow intelligent support and advisory [11]. The di.me userware operates on user-controlled servers, the so called *di.me Personal Servers* (PSs) and can be accessed either by a Web-based UI or an Android application. It aggregates information, gathered from connected services, like social networks or special services for e.g, processing location information. The PSs' form a decentralized network and also implement State-Of-The-Art network security and anonymity mechanisms, like e.g., Tor, StartTLS or OAuth in order to ensure maximum protection of users' privacy and their data.

Collected information is stored in a semantic repository enabling intelligent advisory and advanced techniques like e.g., semantic matching of profiles [12].

3.2 The di.me TrustEngine

The component in the di.me userware, processing trust and privacy in order to calculate respective advisory, is called *TrustEngine*. It is located on the PS and accesses information stored in the semantic core as well as in live context data in order to trigger warnings and advisory, shown in the UI. The TrustEngine is integrated in all interaction flows concerned with the manipulation and especially the disclosure of information (e.g., share a profile to a group, send a file to a contact, etc.). Figure 1 depicts a simplified architecture of the di.me userware in order to illustrate the integration of the TrustEngine into the core of the

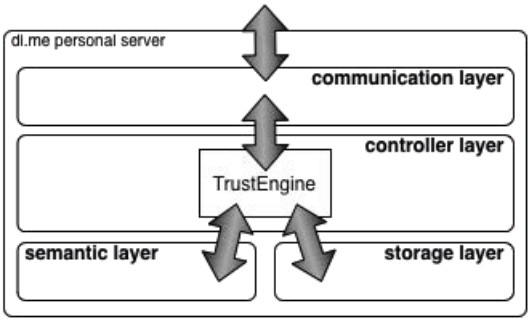


Fig. 1. The TrustEngine in the architecture of the di.me userware

system. The core concept to calculate trust and trigger privacy warnings and advisory was already presented in Heupel et al. 2012 [5]. The general idea is that all information stored within the PS should be classified regarding the privacy of the content. To classify information we us a scale from 0 to 1, where 0 is considered public information and 1 secret[2] The trust value for contacts can then be calculated based on the information shared with them. The disclosure of very private information to a person is an implicit expression of trust. This approach would of course require a short learning phase before it has enough confidence to detect privacy flaws. Once it is calibrated, it is easy to detect if private information is shared to untrusted persons. Further it can detect a possibly wrong classification of information when information marked as private, is shared to a lot of people, it is very likely that the classification is wrong. Besides the automatic calculation of trust values, it is also possible to set it manually in the UI, overriding the calculated value.

3.3 First Prototype and Evaluation

The first prototype was presented to the visitors of a summer school event in Segovia in July 2012. The visitors had the possibility to test the application on a booth or even download it to their own Android phones and were asked to fill out a questionnaire. Additionally a focus group was selected, which got a special presentation and tried the di.me userware on several provided devices. Figure 4 shows screenshots of the first prototype.Most interesting in the evaluation of the first prototype, concerning trust and privacy advisory, was to get an impression if the test users understood the general concept, if they thought it was something useful and if the concepts are presented in an intuitive and non-intrusive way in the UI[3]. Especially the members of the focus group (12 students with

[2] A value of 1 should here really be understood as top secret, e.g., like a private encryption key, which is usually never communicated to anyone.

[3] The interested reader can find more information about the overall UI concepts in [13].

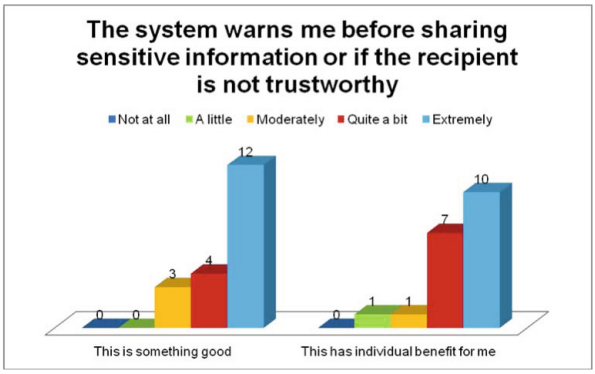


Fig. 2. Answers to one of the questions about privacy advisory in the questionnaire

technological majors) were really interested in the feature of privacy advisory and considered it an extremely useful feature (see also Fig. 2 for some of the evaluation results).

4 Requirements Gathering

Besides the general usefulness of the features, we asked also more concrete questions about selected features in the UI, if they are usable and understandable. By doing this we identified several issues in our concepts[4], from which we address the main critical ones in the context of this paper. In the following we will shortly discuss the change requirements (CR1-CR3) we deducted from those issues.

A major critical point of the first prototype was concerned with the general trust model. There is only one trust value for each person and only only privacy value for each sharable information item. This model turned out to be not covering all different use-cases. The di.me userware is intended to cover all different lifespheres of the user, this means business as well as private. We identified that trust and privacy can be interpreted slightly different, depending on the lifesphere[5]. Especially when dealing with intersection of different lifespheres (like e.g., business and private) this can lead to situation where an unintended information disclosure is not detected. An example for such a situation could be: The user shares accidentally a highly confidential document to his grandmother. Obviously this case will not be detected. Therefore we need to extend the model to detect information being shared to a context, it does not belong to (CR1).

The second important issue we identified, was concerned with the presentation of the trust and privacy values in the UI and how they can be manipulated. The presentation of trust and privacy values in the UI is no trivial task, since it can be easily misinterpreted. As shown in Fig. 4, the privacy value of files, as well

[4] Developed internally by involving selected di.me consortium partners.

[5] The privacy value of confidential business documents can mean something else than for private photos.

(a) Privacy slider (b) Trust slider

Fig. 3. Screenshots of trust and privacy slider in the first prototype

as the trust value for persons are represented as a continuous slider. The label next to the slider was showing the value as a percentage value, depending on the position. This turned out not to be very intuitive. The users were not sure what exactly the setting to a high or low trust means, respectively for the privacy value. It turned out to be easy to misunderstand, if a low trust in someone would mean just *no trust* (e.g., due to a lack of information) or *distrust* (meaning to expect someone to behave to the users' disadvantage). Therefore we needed to redesign the UI elements representing trust and privacy values, in order to make the concept really clear to the user (CR2).

The calculation of trust advisory takes place on the server and is triggered by all calls involving manipulation of data and access rights. Therefore, the TrustEngine is involved if information is shared to a person or group, but also when adding persons to a group or documents to a databox. For all those actions a HTTP request to the PS is necessary, which can produce a lot of communication overhead and lead to a bad UX in the end. Therefore the third identified requirement is to optimize the communication flow for trust advisory (CR3).

5 Approach

5.1 Extending the TrustEngine (CR1)

One of the main critical points that has been identified in the evaluation of the first prototype was, that the trust metric we used might not be applicable in all use-cases, especially when dealing with intersection of different lifespheres (e.g.,

business and private). To overcome this problem, we analyzed the problem and extended our approach from [5] respectively. Our solution envisages to use the groups[6] in the systems to identify lifesphere borders[7].

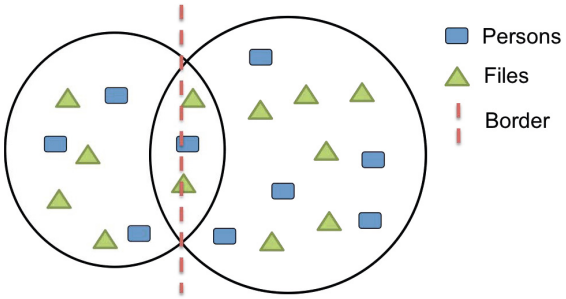


Fig. 4. Illustration of two overlapping groups with common files and persons

We assume that users usually tend to group the contacts and thereby automatically align them to their lifespheres. To foster this behavior we decided to have 3 default groups in the system, for family, friends and business contacts. On top of this, there are automatic created groups for contacts, imported from existing social networks (e.g., all LinkedIn contacts) and automatic ad-hoc groups that are created based on location context information, if contacts are nearby for a certain time (e.g., sitting in the same room during a meeting). In order to identify information passing over lifesphere borders without the users intention, the first step is the identification of such borders. Naturally not every group represents a separate lifesphere (Fig. 4 illustrates two overlapping groups with some common files and contacts). In oder to identify a possible threat, we compute the pairwise distance of two groups, the one the information is already related to[8] and the recipient group. Information becomes "attached" to a group by being shared to members of this group.

In order to calculate the distance between two groups, we consider the two most relevant factors: common contacts and common files. Since we are interested in the difference, and not what both groups have in common, we take the symmetric difference ($\triangle$), of both sets (defined for sets Y, Z in (1)). In a coordinate system both values are orthogonal to each other, so the distance between two groups (= distance to the origin) can be calculated like shown in (2).

$$Y \triangle Z = (Y \setminus Z) \cup (Y \setminus Z) \tag{1}$$

[6] A group in di.me is only for ordering contacts on user side, and should not be confused with "discussion groups". They can be compared to the "circles" of Google+ or the user defined friend lists of Facebook.

[7] A border in this context e.g., between business and private lifesphere is weak in most cases, since there can be colleagues who are also friends or family.

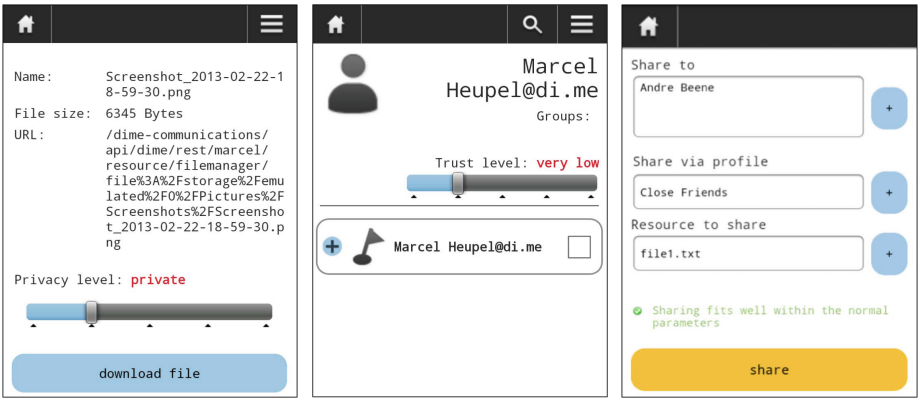
[8] Information becomes related to a group, by being shared to members of those groups before.

$$GD = \sqrt{\left(\frac{|A \triangle B|}{|A \cup B|}\right)^2 + \left(\frac{|C \triangle D|}{|C \cup D|}\right)^2} \tag{2}$$

With this mechanism we can calculate a distance between two groups, where $\sqrt{2}$ is resembling the maximum distance (both groups are completely disjunct). Combined with the privacy value, the new distance can now be used to trigger additional warnings, not covered by the previous approach.

5.2 Reworking UI Concepts (CR2)

As a representation of the trust value in the first prototype, we used a continuous slider and labels showing a percentage value depending on the position. This turned out not to be very intuitive. The test users were not sure what exactly the setting of a trust of e.g., 50% means and the same for privacy. Therefore we consolidated the user feedback and reworked our UI concepts together with UI design experts. Figure 5 (a, b) shows some screenshots of the improved UI for the Android prototype. We left the continuous slider, but the label is no longer showing a percentage value. Instead we use simple text labels like "private" or "public" for the privacy of files and "trusted", "untrusted" for trust. Further, the labels are changing their color depending on the restrictiveness (red→private/untrusted, yellow→medium trust/privacy, green→public/trusted). To reduce the arbitrariness of setted values (e.g., how should someone decide if a file is 64% or 67% private?), we introduced 3 fixed steps on each slider (left, middle, right) and thereby drastically reduced complexity and uncertainty of the user setting the values.



(a) Privacy slider (b) Trust slider (c) Trust awareness

Fig. 5. New trust and privacy concepts in the Android UI

5.3 Optimizing the Interaction Flow (CR3)

In the first version of the clients the TrustEngine was completely integrated in the server, checking all manipulations of data for possible trust and privacy issues. If such an issue was detected, a warning will be sent to the UI instead of an acknowledgment of a successful performed manipulation. The user had then different options to dissolve the issue (e.g., by removing untrusted persons from the recipients) or to ignore the warning and share anyway, which would lead to an adaption of trust values of the recipients (like described before). This approach did work well, but in practical tests we discovered some potential to be optimized as we identified in section 4. Therefore we changed the API a bit, and moved the logic of advisory calculation for file sharing to the client. By doing this we were able to improve the user experience due to a recognizable reduced communication overhead. We were also able to provide additional awareness features in the UI, like e.g., a realtime indicator showing if there is a possible trust issue when adding people to a group of recipients. Figure 5(c) shows two of the new colored textfields, indicating the trust status.

6 Conclusions

In this paper we presented how trust and privacy are addressed in di.me and discussed selected issues identified in first evaluations and user trials of the di.me userware related to them. We also presented improvements to the general model, the UI and the interaction design in order to solve those issues. Thereby we were able to improve the warning mechanism to identify information, shared with the wrong audience, made the UI more intuitive and increased the response time for showing privacy advisory in the client UI.

Since the di.me project follows an agile approach merging best-practices from the security, usability and HCI community [14], we will continuously conduct further evaluations and improve the system accordingly. Especially the new concepts introduced in this paper will be analyzed and evaluated again, with end-users and experts. Besides this, we will further extend the privacy advisory by including analysis of microposts with NLP and including live context information going a step further as we already proposed in [15]. Another target for future work is the improvement of initialization of trust and privacy values in order to reduce the learning phase as well as the need for manual settings to a minimum.

Acknowledgements. The work carried out in order to write this paper was supported by the Seventh Framework Program of the European Union, (FP7/ 2007- 2013), in the digital.me project under grant agreement no. 257787. Special thanks goes to our consortium partners from Fraunhofer IAO for their valuable contributions.

References

1. Cranor, L.F., Garfunkel, S.: Security and Usability: Designing Secure Systems That People Can Use. O'Reiley (2005)
2. Kerckhoffs, A.: La cryptographie militaire. Journal des Sciences Militaires IX, 5–38 (1883)
3. Corritore, C.L., Kracher, B., Wiedenbeck, S.: On-line trust: concepts, evolving themes, a model. International Journal of Human-Computer Studies 58(6), 737–758 (2003)
4. Madejski, M., Johnson, M., Bellovin, S.M.: A study of privacy settings errors in an online social network. In: International Conference on Pervasive Computing and Communications Workshops (PERCOM), pp. 340–345. IEEE (2012)
5. Heupel, M., Fischer, L., Kesdogan, D., Bourimi, M., Scerri, S., Hermann, F., Gimenez, R.: Context-Aware, Trust-Based Access Control for the di.me Userware. In: 5th International Conference on New Technologies, Mobility and Security (NTMS 2012), pp. 1–6 (2012)
6. Aberer, K., Despotovic, Z.: Managing trust in a peer-2-peer information system. In: Proceedings of the Tenth International Conference on Information and Knowledge Management, pp. 310–317. ACM (2001)
7. Jøsang, A., Ismail, R., Boyd, C.: A survey of trust and reputation systems for online service provision. Decision Support Systems 43(2), 618–644 (2007)
8. Heupel, M., Bourimi, M., Scerri, S., Kesdogan, D.: Privacy-preserving concepts for supporting recommendations in decentralized OSNs. In: Proceedings of the 4th International Workshop on Modeling Social Media, MSM 2013. ACM, New York (to appear, 2013)
9. Fang, L., Kim, H., LeFevre, K., Tami, A.: A privacy recommendation wizard for users of social networking sites. In: Proceedings of the 17th ACM Conference on Computer and Communications Security, pp. 630–632. ACM (2010)
10. Chaytor, R., Brown, E., Wareham, T.: Privacy advisors for personal information management. In: SIGIR Workshop on Personal Information Management, Seattle, Washington, pp. 28–31 (2006)
11. Thiel, S., Bourimi, M., Giménez, R., Scerri, S., Schuller, A., Valla, M., Wrobel, S., Frà, C., Herman, F.: A requirements-driven approach towards decentralized social networks. In: Proceedings of the International Workshop on Social Computing, Network, and Services (2012)
12. Cortis, K., Scerri, S., Rivera, I., Handschuh, S.: Discovering semantic equivalence of people behind online profiles. In: Proceedings of the 5th International Workshop on Resource Discovery, RED 2012 (2012)
13. Hermann, F., Schuller, A., Thiel, S., Knecht, C., Scerri, S.: The di.me user interface: Concepts for sharing personal information via multiple identities in a decentralized social network. In: Kurosu, M. (ed.) Human-Computer Interaction, Part I, HCII 2013. LNCS, vol. 8006, pp. 29–38. Springer, Heidelberg (2013)
14. Bourimi, M., Barth, T., Haake, J.M., Ueberschär, B., Kesdogan, D.: AFFINE for enforcing earlier consideration of nFRs and human factors when building socio-technical systems following agile methodologies. In: Forbrig, P. (ed.) HCSE 2010. LNCS, vol. 6409, pp. 182–189. Springer, Heidelberg (2010)
15. Bourimi, M., Rivera, I., Scerri, S., Heupel, M., Cortis, K., Thiel, S.: Integrating multi-source user data to enhance privacy in social interaction. In: Proceedings of the 13th International Conference on Interaccion Persona-Ordenador, INTERACCION 2012, pp. 51:1–51:7. ACM, New York (2012)