Privacy Issues with Sharing Reputation across Virtual Communities

Nurit Gal-Oz
Department of Computer Science
Ben-Gurion University of the Negev
Beer-Sheva, Israel

Tal Grinshpoun
Department of Software Engineering
SCE - Sami Shamoon College of Engineering
Beer-Sheva, Israel

Ehud Gudes
Department of Computer Science
Ben-Gurion University of the Negev
Beer-Sheva, Israel

ABSTRACT

This paper outlines the privacy concerns in the Cross-Community Reputation (CCR) model for sharing reputation knowledge across communities. These privacy concerns are discussed and modeled, and a policy-based approach that copes with them is presented.

1. INTRODUCTION

Information sharing is a key objective in the age of Internet and virtual communities. Considering reputation information as part of a user's identity makes it both a sensitive and a desired data for communities to share. A user may be a member of multiple communities. Sharing reputation that a user has gained in one community can leverage her state in new communities.

Several researchers have studied the issue of transferring reputation data between agents (and communities). Pinyol et al. [7] propose the use of a common ontology in order to exchange reputation between agents. Several preliminary ideas for translating recommendations are proposed in [3]. The exchange and translation of reputation data should not necessarily be bound to a pair of agents. Communities that employ trust and reputation systems gain knowledge about the reputation of their users. Exchange of such reputation is a valuable resource both for the users and for the communities. *Cross-Community Reputation* (CCR) can be achieved by sharing and combining reputation data from different communities [6, 4]. On the one hand, CCR provides many advantages and opens new opportunities for both users and communities. On the other hand, it raises several new privacy issues which are not present in single community domains. Our study is focused on the following issues:

- **Linkability.** In order to enable CCR, one must make sure that the user registered in the two (or more) communities is the same user. This must be done without compromising the user's anonymity in any of the communities and with the requirement of unlinkability between the communities. Standard identity providers (e.g., myOpenID [2]) enable the user to control the identity she provides to the communities and to the CCR provider. The user may wish to hide her identity in one community from other communities that she is a member of. Reputation sharing may lead to linkability which in turn can jeopardize the user's privacy.

- **Reputation dissemination.** Sharing reputation across virtual communities may result in an uncontrolled dissemination of reputation-related information such as the community in which it was originated and the exact attributes that it is composed of. The consent of both the user and the community to participate in the CCR service should be further empowered by the ability to control what reputation information is allowed to be exposed to each destination.

- **Privacy *vs.* trust.** The tradeoff between privacy and trust is well recognized [9]. In order to increase trust within a community one would like to import good reputation values and good credentials from other communities. However, these may expose the details of the reputation values and thus impair the user's privacy. In some instances a user or a community may be willing to report only the aggregated values of reputation. In other cases, users may be willing to disclose the data behind the aggregated values, such as individual ratings (e.g., most hotel recommendation sites disclose individual ratings).

The present paper outlines, discusses, and models the privacy issues in CCR systems. Section 2 provides a brief review of the CCR model. In section 3 we model the privacy concerns in CCR and extend our policy-based approach to cope with these concerns. Section 4 concludes the paper.

2. CROSS-COMMUNITY REPUTATION

The cross-community reputation (CCR) model is the basis for the privacy issues that are raised in the present paper.

PAIS 2011, March 25, 2011, Uppsala, Sweden.

Next, we describe the CCR scenario (See [4] for a detailed description of the CCR computation process).

The process begins when a community that wishes to receive CCR data regarding one of its users (i.e., requesting community), sends a request to relevant responding communities (either directly or through a trusted third party). Communities that have reputation data of the user and are willing to share the information, reply with the relevant reputation data. The received data is assembled into an object containing the CCR data of the user in the context of the requesting community. This process is illustrated in Figure 1: (1): A requesting community sends the CCR provider a request for the CCR of a community member; (2): The CCR provider (represented by TRIC in the figure) compiles a request and (3) submits it to all potential responding communities; (4): Responding communities submit a reputation object of the member at subject; (5): The CCR provider processes all reputation objects and compiles a CCR object; (6): The CCR provider sends the CCR object to the requesting community.

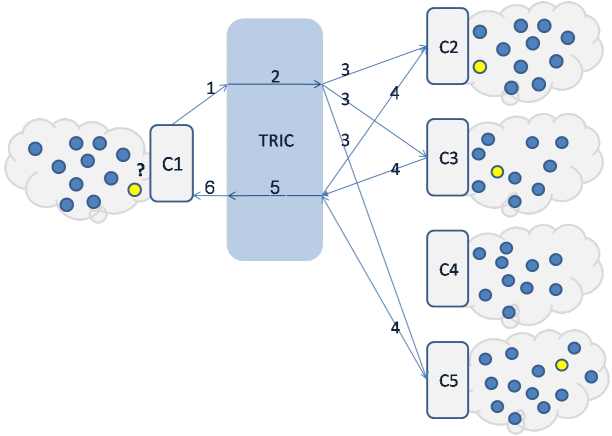

Figure 1: Request for CCR scenario

A major issue in computing CCR [4] is converting between the way reputation is computed and represented in different communities.

3. PRIVACY CONCERNS IN CCR

In his book "The future of reputation" [10] Solove discusses the question of why should we be able to control our reputation and raises the following conflict: *"...we want information to flow openly, for this is essential to a free society, yet we also want to have some control over the information that circulates about us, for this is essential to our freedom as well".*

This paradox is at the heart of the CCR model. We want the users to have control over their information, but we also want to encourage them to share it with others. From the viewpoint of the CCR model there should be some strong incentive to encourage people to comprehensively expose their reputation in the different communities.

Although reputation is treated as a private piece of data, it is hard to perceive it as private once it is publicly known in one community. Preserving user privacy in this respect concerns with allowing a person to use different pseudonyms in different communities while keeping in private the linkage between them. This unlinkability property is also essential for achieving another objective concerning a person's private information – control over reputation dissemination. We want to enable users to have control over who can approach their reputation-related data and to what extent.

Next we discuss privacy-related aspects in the CCR model and suggest means to cope with them within the CCR model and the TRIC framework [4].

3.1 Unlinkability

One of the major concerns of any application dealing with private data is the gradual accumulated information about a person. Two non-private pieces of evidence may turn into a private piece of evidence by a simple join. The requirement for unlinkability of two pseudonyms in the CCR scenario is motivated by two needs:

1. To allow exposure of different parts of one's personal information in different communities. For example, one may not want her travelers community to learn she was once hospitalized for trying "Magic Mushrooms" in Thailand, but she does want to share this information with the forum she is taking part in concerning young people trying drugs.

2. To prevent the identification of the real-world identity of a person based on the data accumulated within two communities. For example, a neurologist that lives in a small town may be anonymously active in both a neurology community and a community that deals with her hometown's municipal issues. A linkage between pseudonyms may render the disclosure of the user's identity, since she is the only neurologist in town.

Unlinkability is achieved in the CCR model by using a centralized architecture with a trusted party. In previous work we have presented a framework for computing and exchanging *Trust and Reputation In virtual Communities* (TRIC) [4]. The CCR model is a core component of this framework.

The TRIC framework assures that the reputation shared among communities does not cause a linkage between a user's identity in one community and that user's identity in another community. It does so by compelling a member of a community to explicitly register to the CCR service in order to share her reputation in the community with other communities she is active in. In the registration process the member provides a virtual identity that can be authenticated by an identity provider that is supported by the service (e.g., myOpenId [2]). From this point on the member should use this identity whenever she initiates registration to CCR services from other communities. However, this identity may be different than the one the member uses within a community. Moreover, the identity provider supported by the CCR service and the identity providers supported by a community may be completely separate entities.

The unlinkability requirement means that the CCR service cannot be aware of the user's identity in the community, and vice versa, the community cannot be aware of the user's identity in the CCR service. Nevertheless, it is mandatory that the CCR service and the community interact and refer to the same user. This issue is addressed at the user registration phase. Registration to the CCR service is initiated by the user from the community. The community then submits the registration request on behalf of the user. Finally,

the CCR service generates a pseudonym for the user and passes it to the community. From this point on, the CCR service and the community use that pseudonym to identify the user. As detailed in [4], pseudonym generation is done only after the user has approved the community's request to register to TRIC and has authorized the sharing of data.

3.2 Control over Reputation Dissemination

Within a single community the reputation of a member is considered public information that is known to all other members of the community. Introducing a member's reputation in another community could violate her privacy. It reveals not only the actual reputation of a member outside the community, but also the fact that she operates in that other community. One can further learn about this person from following her activities in the other community. In order to deal with this problem, a community can provide a member's reputation from other communities without specifying the origin communities and even by hiding possible identifying information. Blocking information such as the set of attributes, the set of categories, and statistical information, turns the responding communities into anonymous sources of CCR data. Control over dissemination of reputation information is done by the definition and enforcement of policies. This is discussed next.

An interesting question is who owns a member's reputation. Obviously the immediate answer to this question would be the member herself. The member was the one who gained reputation due to her honest or professional behavior within the community. Note that "owns" in this sense does not mean that the member can alter the data, but it can make choices on how, to whom, and to which extent the reputation data is shared. Another important player here is the community, which is the key enabler for computing and maintaining this reputation. It does so by collecting and managing all the needed evidences to compute and provide the reputation data. Thus, in some sense the community may also have some say concerning the dissemination of this information outside the community. A community may choose not to publish the components of the final reputation score (e.g., the attributes by which the rating was collected), to protect the privacy of its members. Consequently, we assume that the owners of a member's reputation are both the community and the member . Accordingly we suggest that they can each place their policies to control the dissemination of this valuable information.

CCR policies are concerned with two aspects of the CCR object, namely the CCR *computation* and *representation*. Computing the CCR score requires a reputation object from each of the responding communities. A reputation object may include not only the single computed reputation score but also the scores of the attributes, textual comments, and possibly statistical information. For example, a restrictive policy may disable the use of attributes scores for computation and allow only the use of a single computed reputation score. The motivation behind such a policy can be found in the following scenario: a user who has a reasonable overall reputation in some community but has a relatively low rank in one attribute that is compensated by other excellent attributes. The user may not wish to expose this fact to some other communities.

A CCR object consists of the single score values that result from aggregation of the attributes computed from each of the responding communities. In addition it may contain the score and certainty for each attribute. An even increased level of detail exposes the communities of origin (responding communities) and the scores from each community. However, even if we allow the use of all reputation information available during the CCR computation, we may still restrict the level of details presented when compiling the reputation object to be returned to the requesting community.

A CCR Policy defines which details of a reputation object are provided by a responding community with respect to a member and a requesting community. The policy also differentiate the details that can be used in order to compute the CCR score for this request from the details that can be compiled into the CCR object for the use of the requesting community.

DEFINITION 1. *A CCR Usage Permission p is a pair ($op \in OPS$, $res \in RES$) where op is the operation that a CCR engine can perform and res is the resource on which one the operation can be applied.*

Without loss of generality we define the set of operations $OPS = \{use, publish\}$ and the set of resources $RES = \{score, attributes, origin, text, statistics\}$. This set of resources conforms with the CCR object as designed in this work. Nevertheless, we aim towards a wider definition of CCR in which additional resources may be of interest. For example, one may think of real-life credential such as a *GrandMaster* title from the *World Chess Federation*, as another valuable resource. The set of permissions P in our model consists of the following permissions:

- $(use, score)$ – use the responding community's single reputation score in the CCR computation.
- $(use, attributes)$ – use the responding community's reputation attribute scores in the CCR computation.
- $(publish, origin)$ – publish the community of origin (responding community's name or URL) in the compiled CCR object.
- $(publish, score)$ – publish the responding community's single reputation score in the compiled CCR object.
- $(publish, attribute)$ – publish the responding community's reputation by attributes scores in the compiled CCR object.
- $(publish, text)$ – publish the responding community's reputation textual comments in the compiled CCR object.

DEFINITION 2. *A CCR Dissemination Control Policy ψ is a tuple of the form $\{M, C_{req}, C_{res}, P\}$, where M denotes the set of members at subject, C_{req} denotes the group of requesting communities, C_{res} denotes the group of responding communities to which this policy holds, and P specifies the set of permissions granted by C_{res} or M for compiling a CCR request initiated by C_{req} concerning M.*

The groups of communities are defined by three parameters – *Names*, *Categories*, and *Confidence*:

$$C_{req} = \{C | ((C \in Names) \vee (C \in Categories)) \wedge Confidence(C_{res}, C_{req}) \geq Confidence\} \quad (1)$$

- *Names* is a list of community names, e.g., $Names = \{experts.com, JavaCoders.com\}$ defines all communities that appear in the list. An empty list denotes no community and the set *All* denotes all communities.

- *Categories* is a list of category names, e.g., $Category = \{soccer, football\}$ defines all communities that belong to one or more categories that appear in the list. An empty list denotes no community and the set *All* denotes all communities.

- *Confidence* is a threshold value for the confidence level (see [4]), e.g., $Confidence = 0.5$ defines all communities towards which the confidence level is at least 0.5. When defining the set of requesting communities the confidence considered is from the responding community to the requesting community. When defining the set of responding communities the confidence considered is from the requesting community to the responding community. A zero threshold consists of all communities.

The set of members can be replaced by *All*, denoting all members. The set of permissions may consist of any subset of P. An empty set of permission denotes no permission.

Policies can be defined by either users or communities or by some third party acting as the CCR service. A member m may only define policies in which $M = \{m\}$. A community c may only define policies in which $C_{res} = (Names = \{c\})$ and $M = All$, to prevent members discrimination. Several policies concerning a member can be defined by the different communities she is active in. Moreover, a member and a community may each define a policy concerning the access allowed to the reputation of the member in that community. In these cases the intersection of all permissions yield the valid permission in consistence with the least privileged principle. The valid policy for a CCR request is demonstrated by a short example in the next subsection.

Policies are enforced by the CCR service. Since a responding community has no knowledge about the requesting community, it provides the complete data to the CCR service. In turn, the CCR service resolves the valid policy for the request at subject and performs the computation and compilation of the CCR object accordingly. The compiled CCR object can be composed of partial published data, for example if one community allowed publishing of its origin and attributes and other communities did not.

3.3 Tradeoff between Reputation and Privacy

The policies described in section 3.2 enable control over the dissemination of reputation-related information by each member and community that are the owners of the information. The least privileged rule assures that all policies are enforced and that the valid policy obeys the restrictions. However, in some cases harsher restrictions than required are imposed, and it is the interest of the communities as a group to encourage as much openness as possible in order to assure clear results and prevent data manipulation. For example, if all policies lead to the basic permission $\{use, score\}$ only, the computation will be less accurate. Restricting the set of responding communities may indicate an attempt to hide bad reputation information or the fact that a member is participating in a disreputable community. It can also hint that a community wishes to hide the way it computes reputation. One may think of hiding parts of the details as lack of transparency. The 2010 Edelman Trust Barometer [1] shows that trust and transparency are as important to corporate reputation as the quality of products and services. In [8], Rawlins came with the observation that trust and transparency are significantly and strongly correlated and as organizations become more transparent they will also become more trusted. Accordingly, we assert that transparency is another dimension to evaluate one's reputation, be it a member or a community. Thus, we suggest an incentive mechanism that encourages revealing of information, by grading a CCR object with a *transparency* measure. The transparency computation is derived from the level of restriction imposed on the CCR provider while compiling the CCR object. It treats separately the restrictions imposed by all responding communities and those imposed by the member at subject. This separation is important, since it allows assigning a user with a high transparency level even if the communities involved have blocked the user's information.

A requesting community can evaluate CCR with compliance to its own transparency requirements. Accordingly, it can indicate to the CCR request's subject (who is a member of the community) that the level of transparency is not sufficient for presentation, or alternately present it with a low transparency indication. Following [5], it should be made clear to the user what she can gain from being more transparent about her reputation.

DEFINITION 3. *Transparency Level $\tau \in [0, 1]$ is a value that represents the extent to which a CCR is considered decoded, based on the policies related to it.*

Each CCR is assigned with a pair of transparency values $(\tau_{member}, \tau_{communities})$, the first derived from the member's policy and the second from the communities' policies.

Let $\Psi_{member}(CR)$ denote all policies defined by the member with respect to a CCR request CR and $\Psi_{communities}(CR)$ denote all policies defined by responding communities with respect to a CCR request CR. The requesting community provides the CCR service with weights reflecting the importance it attributes to each permission. The CCR service can use these weights to compute the transparency measures (member transparency and community transparency) of the CCR. The resulting CCR object reflects the combined policies of both the member and the communities while the transparency scores are provided separately.

Consider for example, the permissions granted for a CCR request CR, concerning member m, requesting community C, and responding communities $C_1, .., C_n$, depicted in table 1. The left-hand table displays $\Psi_{member}(CR)$, the valid permissions as obtained for CR from the policies of the member. The left-hand table displays $\Psi_{communities}(CR)$, the valid permissions as obtained for CR from the policies of the responding communities. A value of 1 stands for a granted permission and 0 stands for a denied permission. The bottom line in each table represents the portion of communities that granted the permission.

The valid permission for CR according to all relevant policies is the intersection of the two tables (see section 3.2). This is displayed in table 2.

The computation of transparency is also derived from the data in table 1. A simple approach to compute transparency is to multiply the portion of communities that granted each permission by the normalized weights of each permission

	P1	P2	P3	P4	P5	P6	P7	P8
C_1	1	1	1	0	1	1	1	0
C_2	1	1	1	0	1	1	1	0
C_3	1	1	1	0	1	1	1	1
C_4	1	1	1	0	1	1	0	1
C_5	1	1	1	0	1	1	0	1
%	1	1	1	0	1	1	0.6	0.6

(a)

	P1	P2	P3	P4	P5	P6	P7	P8
C_1	1	1	0	0	1	1	1	0
C_2	1	1	0	1	0	1	0	1
C_3	1	1	0	1	1	0	1	1
C_4	1	0	0	0	1	1	1	0
C_5	0	0	0	0	0	0	0	0
%	0.8	0.6	0	0.4	0.6	0.6	0.6	0.4

(b)

Table 1: (a) Permissions derived from the member's policies, (b) Permissions derived from the communities' policies

	P1	P2	P3	P4	P5	P6	P7	P8
C_1	1	1	0	0	1	1	1	0
C_2	1	1	0	0	0	1	0	0
C_3	1	1	0	0	1	0	1	1
C_4	1	0	0	0	1	1	0	0
C_5	0	0	0	0	0	0	0	0
%	0.8	0.6	0	0	0.6	0.6	0.4	0.2

Table 2: Valid permissions for CR

(as defined by the requesting community). Transparency computation is carried out separately for the member and the communities to obtain τ_{member} and $\tau_{communities}$, respectively. For instance the following vector of weights {0.3, 0.2, 0.2, 0.1, 0.1, 0, 0, 0.1} produces $\tau_{member} = 0.86$ and $\tau_{communities} = 0.5$. The requesting community can conclude that although a substantial part of the CCR information was blocked, the member acted in a relatively transparent manner.

A possible manipulation can be carried out by semi-honest members, assuming that CCR policies of communities are accessible to their users. A member may employ a strategy in which the only permissions she denies are the ones that are granted by the communities. This way the user gains maximum privacy with minimum accountability. A possible solution for this is to measure member transparency only according to the permissions that the member granted on top of the permissions that were granted by the communities.

The incentive mechanism described above aims at members. A member has an incentive to provide transparent reputation whenever it is clear that this reputation data is more valuable than the impaired privacy. In contrast, communities have motivation to hide information in order to preserve the privacy of their members and to keep community information protected. For example, revealing the attributes that a community uses may lead to the disclosure of the importance the community gives to each one of the attributes, and maybe even to the revealing of its computational model. Consequently, an incentive should also be presented to communities in order to motivate communities to share highly transparent reputation objects. This can be shaped in the form of ranking a community's transparency level. A community known to be transparent is perceived as a community that tends not to hide anything unless specifically required by its members. As a result, information received from such a community is considered more valuable, which in turn may translate to monetary advantages along others.

4. CONCLUSIONS

Sharing reputation across virtual communities entails many advantages to both users and communities. At the same time, it raises several new privacy concerns. This paper has outlined, discussed, and modeled the aforementioned privacy issues. The first addressed issue was the need for unlinkability between different pseudonyms of the same user. Another important issue is that of the dissemination of reputation data. We presented a policy-based model that enables both the users and the communities to have control over the dissemination of the data. Finally, we discussed the tradeoff between reputation and privacy and suggested the transparency measure for evaluating a CCR object.

5. REFERENCES

[1] Edelman Trust Barometer, http://www.edelman.com/trust/2010/.

[2] myOpenID, https://www.myopenid.com//.

[3] P. Dondio, L. longo, and S. Barrett. A translation mechanism for recommendations. In *Proceedings of the 2nd Joint iTrust and PST Conferences on Privacy, Trust Management and Security (IFIPTM'08)*, pages 87–102, Trondheim, Norway, June 2008.

[4] N. Gal-Oz, T. Grinshpoun, and E. Gudes. Sharing reputation across virtual communities. *Journal of Theoretical and Applied Electronic Commerce Research*, 5(2):1–25, 2010.

[5] L. Lilien and B. Bhargava. Trading privacy for trust in online interactions. Technical report, Purdu university, Computer Science Dept., 2008.

[6] F. Pingel and S. Steinbrecher. Multilateral secure cross-community reputation systems for internet communities. In *TrustBus '08: Proceedings of the 5th international conference on Trust, Privacy and Security in Digital Business*, pages 69–78, Turin, Italy, 2008.

[7] I. Pinyol, J. Sabater-Mir, and G. Cuni. How to talk about reputation using a common ontology: From definition to implementation. In *Proceedings of the Ninth Workshop on Trust in Agent Societies. Hawaii, USA*, pages 90–101, 2007.

[8] B. L. Rawlins. Measuring the relationship between organizational transparency and employee trust, Spring 2008.

[9] J.M. Seigneur and C. D. Jensen. Trading privacy for trust. In *iTrust*, pages 93–107, 2004.

[10] D. J. Solove. *The Future of Reputation: Gossip, Rumor, and Privacy on the Internet*. Yale University Press, New Haven, CT, USA, 2008.