Privacy Issues with Sharing and Computing Reputation across Communities

Nurit Gal-Oz[1], Tal Grinshpoun[2] and Ehud Gudes[1]
[1] *Department of Computer Science*
Ben-Gurion University of the Negev
Beer-Sheva, Israel
{*galoz,ehud*}*@cs.bgu.ac.il*
[2] *Department of Software Engineering*
Sami-Shamoon College of Engineering
Beer-Sheva, Israel
talgr1@sce.ac.il

Abstract

Online communities are the new-age platform for sharing information and experiences with groups of people. Members of a community are dealing with the daily conflict of having to decide what information they are willing to disclose in order to increase their reputation. Fear of possible privacy loss may lead a member to avoid sharing information with others, even when acting anonymously. Privacy concerns are amplified when users are members of multiple communities and their overall reputation is obtained from their reputation in each community. Disclosing a piece of reputation-related information in one community may cause privacy loss in another community and vice versa. 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

Recent years have seen a substantial growth of virtual communities across the Internet. These enable people to gather around some common goals or shared interests. The accessibility of information and services offered by these communities, makes it both possible and legitimate to communicate with strangers and carry out interactions anonymously, as rarely done in "real" life. On the other hand, virtual communities are prone to many types of deception, possibly exposing users to various threats. These range from people forging their identity and imposing as others, to people giving extremely bad or extremely good ratings to other members unrelated to the service they have received from them.

Trust and Reputation systems provide communities with means to reduce the potential risk when communicating with people hiding behind virtual identities. These systems utilize the experience and knowledge accumulated and shared by all participants for assigning reputation values to individuals. Moreover, they attempt to identify dishonest members and prevent their negative effect.

Centralized reputation systems, such as the commercial system eBay [1], collect and store reputation ratings from feedback providers in a centralized reputation database. The reputation score of a user is simply the sum of her accumulated ratings over a period of six months. Several authors have noted that reputation is a much more complex concept than simply aggregation of ratings. It may depend on interaction of multiple attributes (also exists in eBay), on the certainty of the rating [14], on the time the interaction and rating were performed, and on the trust between members. The last factor, trust between members, is crucial in obtaining reputation information that is specifically compatible with a user's profile or preferences. One usually gives higher weight to ratings provided by people she has trust

Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications, volume: 1, number: 4, pp. 16-34

in. The issue of trust between members has been investigated in the past (e.g. [4]). Specifically, when anonymity of users is required, trust between members may be computed based on the similarity of their past ratings [9].

The downside of the above systems and models is that the reputation engine assumes knowledge of all ratings and other reputation factors. As a result, the raters suffer a severe loss of privacy. An empirical study conducted on data sets extracted from eBay's reputation system reported a high correlation between buyer and seller ratings [26]. Moreover, most of the feedback provided was positive. A possible explanation for these results is that when feedback providers' identities (or pseudo-identities) are known, reputation ratings are provided based on reasons of reciprocation and retaliation, not properly reflecting the trustworthiness of the rated parties. Various privacy issues including the problem of preserving privacy while computing reputation become even more complicated when the reputation data is shared across different communities.

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. At the same time, a reputation that a user has gained at some point in time can leverage her state in new communities. In the real world, a user may be a member of multiple communities. Sharing reputation between such communities may have many benefits. Several researchers have studied the issue of transferring reputation data between agents (and communities). Pinyol and his colleagues [24] propose the use of a common ontology in order to exchange reputation between agents. Several preliminary ideas for translating recommendations are proposed in [5]. The exchange and translation of reputation data should not necessarily be bound to a pair of agents. For instance, 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 [22, 7]. 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 or less significant 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 for privacy reasons 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 [28]. 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). Hence, the privacy/trust tradeoff is a major issue in CCR systems.

- **Privacy-preserving reputation computation.** Computing reputation is a process that may impair the privacy of both the user that requests the reputation data and the user whose reputation is the subject of the request. This problem becomes even more significant when the reputation data is spread across different communities. The confidence of a community that requests reputation information in each responding community should not be disclosed to the latter. On the other hand, the identity of the responding communities and the actual reputation values they provide should remain undisclosed to the requesting community. Accordingly, private computation is considered.

The CCR model and the TRIC infrastructure that enables it were presented in detail in [7]. Policies were discussed in that paper as means to control the level and type of information that the owners of reputation are willing to reveal on top of a single aggregated reputation score. In the present paper we introduce *Social Credentials*, a special case of CCR issued by a CCR service upon a user request, for the purpose of presenting the user's reputation to some third party outside the scope of the communities she is active in. A social credential is context aware and consists of information from a limited set of communities that are relevant to the specified context. Social credentials elevate the potential concealed in *social capital* [3] and bring it to the practical level. Although this paper addresses the general privacy concerns of the CCR model, it also relates to the special case of social credentials.

The present paper outlines, discusses, and models the privacy issues in CCR systems. As far as we know this is the first time privacy issues for sharing reputation across multiple communities are discussed, and this is the major contribution of this paper.

The paper is organized as follows. Section 2 provides an overview of the background and related work. In section 3 we review the CCR model and introduce the concept of social credentials. In section 4 we model the privacy concerns in CCR and extend our policy-based approach to cope with these concerns. Section 5 concludes the paper and gives some future research directions.

2 Background and Related Work

A virtual community is a group of entities (e.g., people, nodes, peers or agents acting on behalf of people), interacting via computer networks for sharing information and experiences with each other. Different communities serve various needs of social groups through different levels of interactions. A community of strangers is a community of anonymous entities who would like to participate, i.e., contribute to and benefit from the community activities, without revealing personal identifying information. This is in contrast to the recently popular Internet communities of identified users (e.g., Facebook, LinkedIn).

In anonymous communities there is a strong motivation for introducing mechanisms that support trust and reputation among community members. A review on trust and reputation systems is provided by Jøsang and his colleagues [15]. Their review discusses the semantics of the trust and reputation concepts and the relations between them. The authors also provide an overview of reputation computation models and existing applications of online reputation systems. Sabater and Sierra [27] also present an overview of several proposed computational models for trust and reputation. The authors classify the models according to several criteria, such as the source of the information used, the assumptions made on agents' behavior, the visibility of trust, and whether reputation is considered as a personal/subjective property or as a global property. The concern for privacy in communities in general, and the privacy of reputation information in particular, was discussed in several papers that are detailed next.

The user-privacy concern in virtual communities is often addressed by the use of pseudonyms and anonymous or private certificates. However, pseudonyms must be certified by some trusted authority to make sure they correspond to a real user. A certification process that uses cryptographic attestation entities is described by Kinateder and Pearson [16]. Another method for generating and proving correctness

of pseudonyms, which is based on smart cards, is described by Yang and his colleagues [32]. Their work also describes a mechanism for giving incentives to feedback providers by using hash chains.

Next we present an overview of the related work, covering the four major issues raised in the introduction: Unilnkability, Reputation dissemination, the tradeoff between privacy and trust, and privacy preserving reputation computation.

Unlinkability is a strong requirement for anonymity and enhancing privacy when multiple communities are involved. Unlinkability can be achieved if no two communities expose the same identifying piece of information related to a user. An information theoretic model of reputation privacy was presented by Steinbrecher [31]. In her work, she attempts to model the amount of lost privacy when a single user uses different pseudonyms in the same community or in different communities, or when a user changes her pseudonym. Steinbrecher's measure enables the estimation of unlinkability in such cases. Pingel and Steinbrecher [23] discuss the issue of privacy when a single user is a member in multiple communities and requires the transfer of the reputation between these communities, thus creating the concept of cross-community reputation. Cross-community reputation requirements were analyzed in [8], including the issues of privacy and user control *vs.* community control.

Reputation dissemination is related to the subject of controlling information dissemination for privacy reasons. Information dissemination approaches related to our work have been focused especially on cryptographic techniques and key management schemes for allowing policy-based dissemination of database items [19] or XML documents consisting of private information [17]. In a recent work Shang and her colleagues [29] propose a protocol for content dissemination which assures policy-based access control and preserves users' privacy in a document broadcasting setting. They suggest a group key management scheme that allows qualified subscribers to efficiently extract decryption keys for the portions of documents they are allowed to access. This scheme is based on subscription information that subscribers receive from the document publisher. For the CCR scenario we assume that the owners of reputation data are both the user and the community that issued it. Both of them control the dissemination of the reputation across communities through a trusted third party acting as the publisher in practice. A CCR object is a composition of several reputation objects of different owners with different dissemination policies. As will be shown in the next section, the control over reputation dissemination is a major goal of the CCR framework.

The general issue of privacy *vs.* trust was discussed in the past. Lilien and Bhargava [18] present a comprehensive analysis of the topic. They first describe the different aspects of trust and privacy and enlist the various threats to privacy that exist in the virtual world. They also give a common scenario where in order to get an interactive service a client needs to present some digital credentials such as credit card information, thus exposing some private information. They state that higher level of trust requires loss of privacy and costs may be associated with each. Specifically, they deal with the questions of how much privacy is lost and how much trust is gained by disclosing a specific credential, and what is the minimal degree of privacy that must be sacrificed to obtain a required amount of trust gain. Finally, the authors present some entropy-based measures for privacy loss and trust gain, and a system called PRETTY for minimizing privacy loss in trust-based interactions. Seigneur and Jensen [28] present a computational model to compute the trust/privacy or trust/anonymity tradeoff. They give cost formulas to the various evidences which reflect loss of privacy as well as trust gain.

One of the major observations made by an empirical study on eBay's datasets [26] indicates that when feedback providers' identities (or pseudo-identities) are known, reputation ratings are provided based on reasons of reciprocation and retaliation, not properly reflecting the trustworthiness of the rated parties. Consequently, preserving privacy while computing reputation becomes an important issue. Several researchers have designed schemes which can compute reputation privately when the reputation components are stored in a distributed manner by each agent and are considered as private information. Pavlov and his colleagues [21] suggest several schemes for privately computing reputation information,

when the reputation is defined as an additive reputation system (e.g., the Beta reputation system [14]). The authors present three algorithms for computing additive reputation information with various degrees of privacy and with different abilities for protecting against malicious users. Another scheme which also uses the simple aggregation method of computing reputation is presented in [13]. The privacy-preserving computation becomes more complex when the reputation model is not a simple additive one. Several schemes for computing reputation privately in a distributed system were presented in the past [11, 12, 20]. The first two papers rely on the Knot model for computing reputation [9]. In the Knot model the reputation computation can be reduced to a sum of terms, where each term is a multiplication of a member's confidence in the user giving the evaluation and the evaluation value itself – both are assumed private. In [11] three schemes are presented, one with a trusted third party and two without a trusted third party. All schemes rely on the use of homomorphic encryption. Coming back to the CCR context, its obvious that a distributed system is unacceptable because of the requirement for unlinkability. Yet, some form of privacy-preserving computation of CCR can be supported. This is discussed in section4.4.

3 Cross-Community Reputation

The cross-community reputation (CCR) model is the basis for the privacy issues that are raised in the present paper. This section begins with an overview of the CCR model (a more detailed description can be found in [7]). It then goes on to introduce the notion of a social credential, which is an important feature of CCR that was not presented in detail in previous work. Social credentials deserve special attention in the present paper, due to the unique privacy issues that they raise.

3.1 Overview of the CCR Model

Before going into details, we provide a brief description of the CCR computation process. The process begins when a requesting community that wishes to receive CCR data regarding one of its users, 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 Figure1: (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.

The CCR Model consists of three major stages – preconditions, conversion of reputation values, and attribute mapping. These stages must be performed for each pair of communities that wish to enable CCR between them. The first two stages are described briefly in the preliminaries. Following that, the attribute mapping stage, as well as the final CCR compilation, are given in more detail.

3.1.1 Preliminaries

As a precondition for receiving any input, the requesting community must decide upon the level of confidence it has in each of the responding communities. This level is decided according to the similarity of the communities' categorization, the conversion uncertainty imposed by different value domains, or simply by an explicit assertion. The higher the confidence level of the responding community, the bigger the influence of that community's data in the CCR computation. Definitions for the three factors of confidence are given next. More detailed descriptions of these factors can be found in [7].

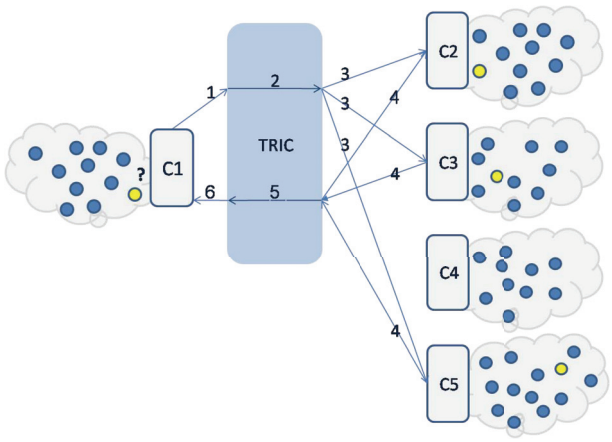

Figure 1: Request for CCR scenario

Definition 1. *Category Matching level is a value in [0,1] representing the correlation of two communities based on their categories as described by keywords.*

Definition 2. *Domain Confidence is a value in [0,1] representing the extent to which one community considers the input from another community as precise, based on conversion uncertainty.*

Definition 3. *Explicit Assertion is a confidence value explicitly provided by a representative of a community with respect to another community.*

An explicit assertion (if available) overrides the confidence that was computed by the first two factors.

Due to differences in reputation representation in different communities, reputation values from responding communities must be converted to values in the requesting community's domain before they can be used. This process subsequently follows the above precondition stage.

A reputation score can be more meaningful when one knows the distribution of ratings in the community it originated from. The same reputation score can be perceived as exceptionally high or as an average popular score, under different distributions. Thus, the conversion stage may also include some *statistical adjustments*.

3.1.2 Attribute Mapping and Computation

Reputation is usually represented by more than a single value. The rating criteria used within a community to evaluate a transaction serves as the set of attributes describing the reputation. An attribute in one community may have the same meaning as an attribute in another community even if they are labeled differently. On the other hand, an attribute may be only partially (or not at all) analogous to one or more attributes used by a different community.

To obtain the relative contribution of an attribute of one community to the CCR computation of another community's attribute, a set of generic attributes is defined. Generic attributes correspond to the rating criteria commonly used by the participating communities. Each community provides a mapping of its attributes to the relevant generic attributes. This mapping specifies the generic attributes that match each of the community's attributes and the level of matching. This information along with the actual attribute scores provided by the responding communities, enables the computation of the CCR attribute scores and the level of certainty one has in the firmness of each of these scores.

Definition 4. *Matching Level is a number in the range [0,1] specifying the extent to which the meaning of one attribute is considered analogous to that of another attribute.*

Let $GenericAtt = \{GAtt_1,..,GAtt_n\}$ be the set of generic attributes, and let $\{Att(A)_1,..,Att(A)_s\}$ be the set of attributes used in a requesting community A. For each attribute $Att(A)_i, i = 1..s$ there is a mapping to each attribute in $GenericAtt$ that it matches denoted by $Att(A)_i.ML(GAtt_j)$.

The process of computing the score and the certainty of an attribute of the requesting community has two parts. In the first, the score and the certainty of each of the relevant generic attributes are evaluated from their matching attributes in the responding communities. In the second part, the score and the certainty of the attribute at subject are evaluated from the generic attributes' scores and certainties as computed in the first part.

Let $Att(B_i) = \{Att(B_i)_1,..Att(B_i)_{s_i}\}$ be the sets of attributes of $B_i \in B_{res}$, the set of responding communities that passed the confidence threshold for A. Following that, $Att(B_i)_l.Score$ is the score of the inquired subject for attribute l in the responding community B_i, and $Att(B_i)_l.Support$ corresponds to the number of ratings that constitute the score. The certainty and score of the generic attribute $GAtt_j \in GenericAtt$ with respect to Community A is given by:

$$GAtt_j(A).Certainty = \sum_{B_i \in B_{res}} \sum_{l=1}^{s_i} Att(B_i)_l.ML(GAtt_j) \cdot Confidence(A,B_i) \cdot Att(B_i)_l.Support \quad (1)$$

$$GAtt_j(A).Score = \frac{\sum_{B_i \in B_{res}} \sum_{l=1}^{s_i} Att(B_i)_l.Score \cdot Att(B_i)_l.ML(GAtt_j) \cdot Confidence(A,B_i) \cdot Att(B_i)_l.Support}{GAtt_j(A).Certainty} \quad (2)$$

To compute the score and certainty of an attribute $Att(A)_k$ for the requesting community, we use only the set of generic attributes that match it with matching level > 0 denoted by $GenAtt(Att(A)_k)$.

$$CCR.Att(A)_k.Certainty = \sum_{GAtt_j \in GenAtt(Att(A)_k)} Att(A)_k.ML(GAtt_j) \cdot GAtt_j(A).Certainty \quad (3)$$

$$CCR.Att(A)_k.Score = \frac{\sum_{GAtt_j \in GenAtt(Att(A)_k)} GAtt_j.Score \cdot Att(A)_k.ML(GAtt_j) \cdot GAtt_j(A).Certainty}{CCR.Att(A)_k.Certainty} \quad (4)$$

We assume that the domain of values of all attributes within a community are the same and aligned with the domain of the aggregated reputation value. The conversion stage described in section3.1.1 must be applied to every attribute score provided by a responding community before the above computation is carried out.

3.1.3 Compiling CCR

A *reputation object* provided by a community contains in addition to the single reputation score, information related to the attributes of that community (score and certainty) and statistical information related to the reputation value (e.g., its standard score). A reputation object may also include textual comments written by users who have rated the user that is the subject of the reputation object.

A CCR object is a reputation object containing the aggregated attributes values (score and certainty) that are computed from all responding communities. It also contains a CCR single score computed from the CCR attribute values using the attributes-weights assigned by the requesting community.

Let $w_1,..,w_n$ be the weights that the requesting community A assigns to its reputation attributes $Att_1,..,Att_n$

$$CCR.SingleScore = \frac{\sum_{i=1}^{n} CCR.Att(A)_i.score \cdot w_i}{\sum_{i=1}^{n} w_i} \quad (5)$$

The weights used for each CCR attribute in the CCR single score computation are defined by a community. Clearly, in communities that support user defined weights to calculate internal reputation scores,

the CCR single score can be computed by the community, customizing the weights to the preferences of the viewing user (see [10]).

In addition we compute an inscrutable reputation that ignores attribute information. It is calculated as a weighted average of all single reputation scores provided by the responding communities, weighted by the confidence that the requesting community has in them:

$$CCR.InscrutableReputation = \frac{\sum_{i \in B} Confidence(A, B_i) \cdot B_i.ReputationScore}{\sum_{i \in B} Confidence(A, B_i)} \quad (6)$$

This score is important when there are relatively few attributes in the responding community that match the attributes of the requesting community. Unlike the CCR single score, it is insensitive to the internal representation of the responding communities' reputation scores and it does not express the weights of reputation attributes.

3.2 Social Credentials

The society tributes people in different occasions for various reasons. A thank-you certificate for voluntary help in some charity organization, a recognition certificate for winning the employee of the year award, or a bravery award for a courageous attempt, are all examples of credentials provided by the society that may assist in building a person's reputation in real life. We suggest the online version of these certificates and name it *Social Credentials*. The term social credential is derived from the term *social capital* which was defined by Adler and Kwon [3] as *"the goodwill available to individuals or groups"*. They explain that goodwill refers to the *"sympathy, trust, and forgiveness offered us by friends and acquaintances"*. Replacing friends and acquaintances with members of the community, the potential social capital that can be gained within the CCR model is tremendous. We attempt to leverage this potential in the form of social credentials.

Definition 5. *A Social Credential is a signed certificate issued by a CCR service for a member acting in one or more communities, stating the cross-community reputation of that member as compiled based on her reputation within some of the communities with respect to a predefined context.*

The context of a social credential refers to the community's categories as well as to its reputation attributes. An example for a context is "a real-estate expert", which refers to communities categorized as specializing in real-estate (or similar categories). The attributes in this context can be limited to the pure-professional ones such as "level of expertise" and "knowledge", while attributes such as "courtesy" or "manners" may be left out. The main difference between CCR and social credentials lies in the initiator of the request as well as in the viewpoint that configures the computation. In CCR the initiator is a community and the purpose is to provide reputation information derived from other communities about a member of the community. The initiator of a social credential is the member herself with the purpose of exhibiting the credential to some entity outside her communities. While a CCR represents the viewpoint of the requesting community, a social credential's point of view is derived from the context and from the initiating user.

Formally, the computation of a social credential (SC) is a special case of the CCR computation, in which there is no requesting community. The role of the requesting community is filled by the social credential request. The context of a social credential request states the attributes and categories (these are defined by the requesting community in the CCR scenario). The responding communities in the social credential scenario are all communities matching the defined categories that have relevant information in one of the attributes defined by the credential's context. The confidence assigned by the social credential

request in each one of the responding communities is determined by three factors – *Category Matching level*, *Domain Confidence*, and *Explicit Assertion* – stating the confidence that a member being the reputation subject has in the responding community.

Category Matching level is based on the correlation between the SC context's categories and the responding community and follows definition 1. The Domain Confidence follows definition 2, where the domain of the social credential's attribute values is real (as in the case of generic attributes). The actual computation of social credential is similar to that of CCR with one exception – the part of converting back to the requesting community set of attributes (see equations 3 and 4) is not required since these are the generic attributes. Consequently, the following equivalences hold:

$$CCR.Att(SC)_k.Certainty \equiv GAtt_k(SC).Certainty \tag{7}$$

$$CCR.Att(SC)_k.Score \equiv GAtt_k(SC).Score \tag{8}$$

We compute the certainty and score of each generic attribute $GAtt_j \in GenericAtt$ stated in the SC's context according to equations 1 and 2, where *SC* replaces the requesting community *A*:

$$GAtt_j(SC).Certainty = \sum_{B_i \in B_{res}} \sum_{l=1}^{s_i} Att(B_i)_l.ML(GAtt_j) \cdot Confidence(SC, B_i) \cdot Att(B_i)_l.Support \tag{9}$$

$$GAtt_j(SC).Score = \frac{\sum_{B_i \in B_{res}} \sum_{l=1}^{s_i} Att(B_i)_l.Score \cdot Att(B_i)_l.ML(GAtt_j) \cdot Confidence(SC,B_i) \cdot Att(B_i)_l.Support}{GAtt_j(SC).Certainty} \tag{10}$$

Social credentials serve as incentives for members of the community to build a highly positive reputation and to maintain it as such. A person holding bad reputation in some community may naturally avoid showing it or exclude it from her social credential. Moreover, as in CCR, a user may avoid the compilation of a detailed certificate from multiple communities in order to prevent linkability of her different pseudonyms. This creates a conflict since a person willing to share more information may have a less worthy credential than a person that carefully selects the sources of her credentials. Thus, openness and transparency should be appreciated more than restrictiveness. On the other hand, the user is the owner of her data and as such should be able to expose only parts it.

4 Privacy Concerns in CCR

In his book "The future of reputation" [30] 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 private the linkage between the two pseudonyms. 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.

4.1 Unlinkability

One of the major concerns of any application dealing with private data is the gradual accumulated information about a person. While each piece of data in itself is not private, their assembling may reveal private information and even lead to the real identity of a user in case it was anonymous. 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) [7]. 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 a 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. Nonetheless, 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. More details on how unlinkability is enforced in TRIC can be found in [7].

4.2 Control over Reputation Dissemination

> *"Reputation is a core component of our identity – it reflects who we are and shapes how we interact with others – yet it is not solely our own creation."* — Daniel Solove, *The future of reputation*

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 that should be addressed when discussing the control over reputation dissemination 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. However, the community as a platform 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. In section 3.1.1 we identify the confidence that one community has in another as part of the reason it should or should not share information with the other community. Moreover, 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. As discussed in section 3.1.3, 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. This case is treated as if all attributes of the requesting community are ranked the same as the final reputation score by the responding community.

A CCR object consists of the single score values (equation 5) that result from aggregation of the attributes computed from each of the responding communities. Using only this data results in inscrutable reputation (equation 6). 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 detail level at the reputation object's compilation stage.

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 differs between the details that can be used in order to compute the CCR score for this request and the details that can be compiled into the CCR object for the use of the requesting community.

Definition 6. *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.
- $(use, statistics)$ – use the responding community's reputation statistics 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.
- $(publish, statistics)$ – publish the responding community's reputation statistics in the compiled CCR object.

The permissions $(use, origin)$ and $(use, text)$ have no meaning at this point and were therefore omitted from the above list.

Definition 7. *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*:

$$C_{req} = \{C | ((C \in Names) \vee (C \in Categories)) \wedge Confidence(C_{res}, C_{req}) \geq Confidence\} \qquad (11)$$

where:

- *Names* is a list of community names, e.g., $Names = \{experts.com, JavaCoders.com\}$ defines all communities that appear in the list. The special name *SC* refer to social credentials. 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 section 3.1.1), 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.

Definition 8. *Member-Communities Policies Set* $\Psi(m, c_{req}, c_{res})$ *is the set of all policies that should be applied in a CCR request regarding member m from a requesting community* c_{req} *to a responding community* c_{res}*:* $\Psi(m, c_{req}, c_{res}) = \cup\{\psi = \{M, C_{req}, C_{res}, P\} | m \in M, c_{req} \in C_{req}, c_{res} \in C_{res}\}$

Let $\Psi(m, c_{req}, c_{res}) = \{\psi^1, .., \psi^k\}$ be the set of *k* policies concerning member *m*, requesting community c_{req} and responding community c_{res}, such that $\psi^i = \{M^i, C^i_{req}, C^i_{res}, P^i\}$. To carry out the CCR computation requested by c_{req} with respect to member *m*, the valid policy for each responding community c_{res} is determined by:

$$\psi(m, c_{req}, c_{res}) = \{m, c_{req}, c_{res}, \bigcap_{i=1}^{k} P_i\} \tag{12}$$

The valid policy for a CCR request is demonstrated by a short example in the next section.

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.

When there is a single responding community to a CCR request, information may be leaked even if not explicitly permitted. For example, if publishing of a responding community origin is allowed but other details are not, it is easier for curious users to track these details. Therefore, policies may be extended with another level of restriction on the permission in the form of a condition that specifies the terms that must be fulfilled in order for the permission to hold.

4.3 Tradeoff between Reputation and Privacy

The policies described in section 4.2 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 (inscrutable reputation). Restricting the set of responding communities may indicate for an example 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 [6] shows that trust and transparency are as important to corporate reputation as the quality of products and services. In his research on measuring the relationship between organizational transparency and employee trust [25], 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 the restrictions 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. In the social credentials scenario this is even more significant. A third party supporting SC may not accept an SC with low transparency. Alternatively, it can accept it as a less valuable credential. Following [18], it should be made clear to the user what she can gain from being more transparent about her reputation.

Definition 9. *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*. From these policies we can obtain the number of communities that grant each permission according to the member's policies. Similarly, the number of communities that grant each permission according to the communities' policies is denoted $\Psi_{communities}(CR)$. 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. Since the CCR service is acting as a trusted party in the CCR scenario, there is no real motivation in hiding the weights of the requesting community. The resulting CCR object reflects the combined policies of both the member and the communities while the transparency scores are provided separately. This is demonstrated next.

Consider the example in table 1 that describes a CCR request *CR*, concerning member *m*, requesting community *C*, and responding communities $C_1,..,C_n$. 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 4.2). 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 (as defined by the requesting community). This 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 potentially 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

	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*

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.

In the social credentials scenario the consumer of the SC is not part of the CCR framework. Thus, it is not reasonable to assume that it should reveal its transparency requirements to the CCR provider. On the other hand, the CCR provider cannot disclose the policy of the member nor the policies of the communities. Nonetheless, both sides should disclose some information to each other in order to allow the member to adjust her policy to the consumer's policy, if desired. This can be achieved by defining and following a disclosure protocol that is based on negotiation.

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.

4.4 Private Computation of CCR

As was discussed in section 4.1, unlinkability cannot be provided without some third party entity managing the CCR computation (CCR provider). Consequently, the meaning of privacy-preserving computation of reputation has a different meaning in the CCR environment than in the distributed environment described in section 2.

The CCR computation involves basically three sets of values – confidence of the requesting community, mapping of attributes between the generic attributes and the responding communities, and the local reputation and attribute values of the responding communities. Since the second set of values (mapping) is the same for all users, we assume it remains constant and that it is known to the CCR provider. We would like to protect the privacy of the requesting community by hiding the confidence value, and the privacy of the responding community by hiding the reputation (attributes) values. The above scheme can be easily adapted to other factors that should be kept private in the CCR computation.

Once we have decided which data should be protected, we can apply any scheme that ensures private computation of reputation while relying on a trusted third party. Several schemes for private computation of reputation were proposed in [11]. The most fitting to the CCR scenario is the first scheme, i.e., the one with a trusted third party. Basically one has to use homomorphic encryption on both the reputation and confidence values and apply homomorphic multiplication [11].

One additional issue is that of policies. A problem arises whenever a responding community (or user) does not allow the sharing of a user's reputation with the requesting community. In case the responding community decides to set the reputation value to zero in such a situation, this value will take part in the CCR computation, which will result in a wrong CCR value. Consequently, an initial step is needed whenever policies are considered. In this step every community willing to participate in the CCR computation outputs a reputation value of 1. The aggregated value will be used as a true normalization factor, entailing that the non-responsive communities will be able to send zero reputation values without affecting the true CCR aggregated value.

5 Conclusions

Sharing reputation across virtual communities entails many advantages to both users and communities. At the same time, it raises several new privacy concerns. The notion of social credentials that was introduced in the present paper raises some of its own privacy issues.

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. We then continued to a discussion over the tradeoff between reputation and privacy and suggested the transparency measure for evaluating a CCR object. In order to attain a high transparency rank members are encouraged to disclose their reputation-related information. Finally, we suggested an adaptation of a privacy-preserving computation scheme to the scenario of reputation sharing.

In future work we intend to further formalize a three-legged negotiation protocol that educates the user regarding the transparency deficiencies of her social credentials. This protocol will enable a member to adjust her CCR policies in order to meet the transparency requirements of her social credential consumer. The transparency of communities should be modeled within an incentive mechanism that encourages communities to leave the major control over reputation dissemination in the hands of the member who owns it. Nevertheless, communities should restrict dissemination of a member reputation-information that may affect the privacy of other members.

References

[1] eBay, http://www.ebay.com/.

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

[3] P. Adler and S. Kwon. Social capital: Prospects for a new concept. Technical report, Academy of Management.

[4] S. Chakraborty and I. Ray. Trustbac: integrating trust relationships into the rbac model for access control in open systems. In *Proc. of the eleventh ACM symposium on Access control models and technologies (SACMAT '06)*, pages 49–58, New York, NY, USA, 2006. ACM.

[5] 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.

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

[7] 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.

[8] N. Gal-Oz, T. Grinshpoun, E. Gudes, and A. Meisels. Cross-community reputation: Policies and alternatives. In *Proc. of International Conference on Web Based Communities (IADIS - WBC2008)*, 2008.

[9] N. Gal-Oz, E. Gudes, and D. Hendler. A robust and knot-aware trust-based reputation model. In *Proceedings of the 2nd Joint iTrust and PST Conferences on Privacy, Trust Management and Security (IFIPTM'08)*, volume 263, pages 167–182, Trondheim, Norway, June 2008.

[10] N. Gal-Oz, E. Gudes, and D. Hendler. A robust and knot-aware trust-based reputation model. In *Proceedings of the 2nd Joint iTrust and PST Conferences on Privacy, Trust Management and Security (IFIPTM'08)*, pages 167–182, Trondheim, Norway, June 2008.

[11] N. Gilboa, N. Gal-Oz, and E. Gudes. Schemes for orivately computing trust and reputation. In *Proc. of IFIPTM, Morioka, Japan, 2010*, pages 1–16, 2010.

[12] E. Gudes, N. Gal-Oz, and A. Grubshtein. Methods for computing trust and reputation while preserving privacy. In *Proc. of DBSEC, Montreal, Canada, 2009. Springer Lecture Notes 5645*, pages 291–298, 2009.

[13] O. Hasan, L. Brunie, and E. Bertino. k-shares: A privacy preserving reputation protocol for decentralized environments. In *The 25th IFIP International Information Security Conference (SEC 2010)*, 2010.

[14] A. Jøsang and R. Ismail. The beta reputation system. In *proceedings of 15th Bled Electronic Commerce Conference e-Reality: Constructing the e-Economy*, June 2002.

[15] A. Jøsang, R. Ismail, and C. Boyd. A survey of trust and reputation systems for online service provision. *Decis. Support Syst.*, 43(2):618–644, March 2007.

[16] M. Kinateder and S. Pearson. A privacy enhanced peer-to-peer reputation system. In *EC-Web*, pages 206–215, 2003.

[17] A. Kundu and Elisa Bertino. Structural signatures for tree data structures. *PVLDB*, 1(1):138–150, 2008.

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

[19] G. Miklau and D. Suciu. Controlling access to published data using cryptography. In *VLDB*, pages 898–909, 2003.

[20] J. Nin, B. Carminati, E. Ferrari, and V. Torra. Computing reputation for collaborative private networks. In *33rd IEEE Int. COMPSAC conference*, pages 246–253, 2009.

[21] E. Pavlov, J. S. Rosenschein, and Z. Topol. Supporting privacy in decentralized additive reputation systems. In *Proc. of 2nd Intl. Conf. on Trust Mgmt (iTrust'04)*, pages 108–119, 2004.

[22] 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.

[23] F. Pingel and S. Steinbrecher. Multilateral secure cross-community reputation systems for internet communities. In *proceedings of TrustBus (TrustBus '08)*, pages 69–78, 2008.

[24] 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.

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

[26] P. Resnick and R. Zeckhauser. Trust among strangers in Internet transactions: Empirical analysis of eBay's reputation system. In Michael R. Baye, editor, *The Economics of the Internet and E-Commerce*, volume 11 of *Advances in Applied Microeconomics*, pages 127–157. Elsevier Science, 2002.

[27] J. Sabater and C. Sierra. Review on computational trust and reputation models. *Artificial Intelligence Review*, 24(1):33–60, 2005.

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

[29] N. Shang, M. Nabeel, F. Paci, and E. Bertino. A privacy-preserving approach to policy-based content dissemination. In *ICDE*, pages 944–955, 2010.

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

[31] S. Steinbrecher. Design options for privacy-respecting reputation systems within centralised internet communities. In *Proc. of the IFIP TC-11 21st International Information Security Conference (SEC 2006)*, pages 123–134, 2006.

[32] B. Yang, M. Zhou, and G. Li. A reputation system with privacy and incentive. In *SNPD (1)*, pages 333–338, 2007.

Nurit Gal-Oz received her B.Sc. and M.Sc. (Hons) degrees from the Department of Mathematics and Computer Science at Ben-Gurion University, Israel. Prior to her Ph.D. studies she headed the R&D of several companies in the software industry in various domains including timetabling solutions, business intelligence and distributed web-based applications. Currently she is a research assistant in the Deutsche Telekom Laboratories at Ben-Gurion University, pursuing her Ph.D. degree under the supervision of Prof. Ehud Gudes. Her current research interests include Trust and Reputation systems, Privacy, and Data mining with special focus on role mining.

Tal Grinshpoun is a faculty member of the department of Software Engineering at the Sami-Shamoon College of Engineering, Israel. His research interests include wireless routing, distributed security, trust and reputation systems, and distributed search. Tal received his B.Sc. in Mathematics and Computer Science from Ben-Gurion University, Israel in 1998. He received his M.Sc. and Ph.D. in Computer Science from Ben-Gurion University, Israel in 2005 and 2010, respectively.

Ehud Gudes received his B.Sc. and M.Sc. from the Technion - Israel Institute of Technology, and his Ph.D. in Computer and Information Science from the Ohio State University in 1976. Following his Ph.D., he worked both in academia (Penn State University, Ben-Gurion University, Florida Atlantic University), where he did research in the areas of Database systems and Data security, and in Industry, where he developed Query languages, CAD software, and Expert systems. He has published over 120 papers in the above general areas, and was the chair of several international conferences, including the 2002 and 2009 IFIP WG11.3 conference on Data and Application security. He is currently

a Professor in Ben-Gurion University heading the Computer Science Department and leading several research projects in data security. Ehud Gudes is a member of both ACM and IEEE Computer Society, and an active member of the IFIP WG11.3 group on Database security. His research interests encompass the domain of knowledge and databases, data security and Data mining especially Graph mining and sequence mining.