Privacy and Liveliness for Reputation Systems

(no author shown for review)

Abstract. Privacy-respecting reputation systems have been constructed based on anonymous payment systems to implement raters' anonymity for privacy-respecting reputation systems. To the best of our knowledge, all these systems suffer from the problem having a "final state", that is a system state in which users have no incentive any longer to behave honestly, because they reached a maximum reputation or they can no longer be rated. Thus the reputation is in fact no longer lively. We propose a novel approach to address the problem of liveliness by the introduction of negative ratings. We tie ratings to actual interactions to force users to also deposit their negative ratings at the reputation server. Additionally we enhance raters' anonymity by limiting timing attacks through the use of transferable-eCash-based payment systems.

Keywords: Reputation, Trust, Privacy Enhancing Technology, Anonymity

1 Introduction

Internet users find various opportunities to interact with each other. They sell and buy various objects in electronic marketplaces such as eBay[1], discuss topics in numerous discussion fora, wikis and so on. When interacting with other users, they want to know what to expect from these and based on this expectation they have a certain amount of trust in the fulfillment of their expectations.

People usually build their trust on already existing relationships. On the Internet users often use pseudonyms; thus, already known interaction partners might appear as new. In order to support users in estimating what to expect from an (apparently) new interaction partner, reputation systems have been designed and established to collect the experiences of others, e.g. by Resnick et al. [1]. Before interacting with others, users may investigate on the interaction partner's reputation profile. Thereby users and designers of reputation systems assume implicitly that the users' past behavior gives a strong indication about their future behavior.

An overview of common reputation systems can be found in [2][2]. From these, eBay implements a popular reputation system. This system poses certain risks for user privacy, as it allows to gather profiles of a user's behavior, e.g., time and frequency of participation in interactions, and user's interest in specific products.

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

[2] Although this article is 10 years old the changes to reputation systems currently in use are only marginal.

Even if users can act pseudonymously, they run the risk of re-identification, as it typically happens for eBay partners during shipping and payment.

Reputation systems can be seen as databases that collect information about who interacted with whom in which context. Thus, they are a promising target for numerous data collectors. However, according to Bygrave [3], opinions about a natural person can be seen as personal data, so that the respective person's right on informational self-determination should be applied. Therefore, explicit reputation should only be accumulated about users who agreed on accumulation. Furthermore, reputation information should be protected by means of technical data protection, as outlined by Mahler and Olsen [4].

Hence, reputation systems that respect privacy are needed, while they still enable users to investigate reputation profiles, which allow an estimation what to expect from interaction partners. In Sect. 1.1 we outline solutions that take this approach. Some of them make use of anonymous payment systems to reach anonymity of raters. Related work on this area is outlined in Sect. 1.2. Abstracting from this related work, in Sect. 2 we present a general model on how to define and evaluate requirements focusing on privacy for reputation systems. By means of this model existing privacy-respecting systems based on anonymous payment systems are analyzed in Sect. 3. In Sect. 4.1 we describe our proposal for a new privacy-respecting reputation system system and demonstrate its advantages over existing approaches. Finally in Sect. 4.2 we analyze our protocol and conclude in Sect. 5.

1.1 Related Work on Privacy-respecting Reputation Systems

A central problem for privacy-respecting reputation systems is that they must guarantee that users cannot abolish negative reputation. This can either be reached by only allowing positive reputation, as proposed by Voss and Androulaki et al. [5,6], by making it difficult for the user to distinguish between positive and negative ratings, as proposed in [7] by Steinbrecher, or by a trusted third party. Thereby, this trusted third party can either be an external reputation provider, as proposed by Pingel et al. and Anwar et al. [8,9] or a trusted platform module for the user, as proposed in [10,5] by Kinateder et al. and Voss.

Anonymity of the users involved is not as easy as just using anonymizing services on the network layer. This approach reaches only anonymity for the users requesting reputation of others, as suggested in [11] by Pavlov et al. for the anonymized RING-Network. In order to obtain anonymity of raters *and* ratees, it needs to be ensured that many users are indistinguishable by an attacker, so that they are in large anonymity sets.

For anonymity of ratees, others should not be able to link previous interactions to a current one. The possibility to recognize users by reputation is limited if the set of possible reputations is limited as shown in [12] by Steinbrecher or the reputation is only published as an estimated reputation as proposed by Dellarocas [13]. The recognition of users by pseudonym can be avoided by using transaction pseudonyms [14,6].

Anonymity of raters needs interactions and ratings related to these interactions to be unlinkable. Again, this can be reached by a reputation provider who might only calculate a new user reputation after he collected not only one but several ratings [15] or who might only publish an estimation of the actual reputation [13]. A rater can also be anonymous against the reputation provider by using convertible credentials [12] or anonymous payment systems [6].

1.2 Related Work on Anonymous Payment Systems and One-show Credentials

We base our system on Chaum's eCash [16]. An electronic cash system aims at emulating regular cash. Users withdraw coins from a bank to pay merchants, that are special users, who offer a service. eCash is called transferable if a merchant can use such a coin to pay another user without the help of the bank. eCash provides anonymity properties. For our purposes, we assume a system that provides *perfect anonymity* as presented by Gouget et al. [17], that is a system where an adversary cannot link a spending to a withdrawal: he cannot decide if two coins are spent by the same user, and he cannot decide whether he already owned a coin or not. However, a user can see how old a coin is, i.e., how often it has been spent. This can be seen as a weakness, but we will deploy this property for our protocol.

Furthermore we use one-show credentials, as described, e.g., by Brands [18], which are a primitive similar to electronic coins: they can be spent only once. The main difference is that no account-keeping bank is needed. However, all one-show credentials already shown need to be published in such a way that every user can check their validity by executing the `Deposit` algorithm. For our application it is important that these coins are published anonymously since otherwise traffic analysis becomes too easy. Given an anonymous communication channel, the reputation provider can also manage the lists of shown one-show credentials and can support users by detecting double shows.

2 System Model

For our system environment shown in Figure 1, we assume a community system allowing pseudonymous interactions between users. This might be, e.g., a marketplace such as eBay where every user might be a seller (provider) or buyer (client). Let M be such a user offering interactions under the pseudonym P_M to other users. The community deploys a reputation system provided by a reputation provider ReP. The reputation system collects positive and negative experiences of users' behavior during interactions. Thus we assume that only interaction-derived reputation is aggregated by our system. If a user U becomes interested in the interaction offered by P_M, U inquires P_M's reputation under pseudonym P_{U_1}. If U decides to take part in this interaction, U uses another pseudonym P_{U_2} to interact. Afterwards, U rates P_M using a new pseudonym

P_{U_3}. M can now include the rating P_M got in the overall reputation account at ReP.

The above described usage of pseudonyms is called transaction pseudonyms, as defined by Pfitzmann et al. [19], since for every transaction a new and unlinkable pseudonym is used. In the reminder of the paper we call transaction pseudonyms just pseudonym, while long term pseudonyms are named by their role (e.g. user, peer, merchant).

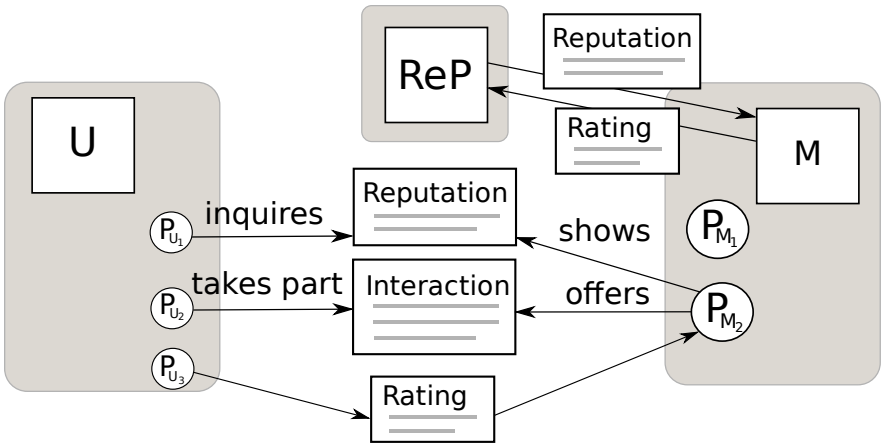

Fig. 1. Model of system environment

2.1 Requirements

The requirements we propose for a privacy-respecting reputation system have a significant overlap with the requirements for reputation systems derived in [12,20].

Rating. After an interaction between two pseudonyms P_M and P_{U_2}, the reputation system provides P_{U_2} with a rating function that allows him to rate P_M, now the so-called ratee. For the rating function the following requirements should be fulfilled:

1. *Integrity of ratings:* Users want ratings to be preserved from manipulations.
2. *Authorizability of ratings:* Only users who interacted with a ratee are allowed to rate him.
3. *Anonymity of raters:* Users want to rate anonymously to not necessarily allow attackers to link this rating to an interaction. This means the pseudonym P_{U_2} that interacted with P_M should not be linkable to the pseudonym P_{U_3} that rates P_M.

The reputation system updates M's global reputation aggregated from the received ratings. The rating of a user's behavior and the aggregation of his ratings to a reputation value have to follow specific rules fixed by the system designer. These rules typically depend on the application scenario and have to fulfill sociological and economic requirements. We abstract here from the concrete functions to allow a universal design interoperable with multiple application scenarios. An overview of possible functions is for example given by Mui [21]. For an economic introduction we refer to Dellarocas [22]. The following requirements should hold:

4. *Fairness of reputation:* Users want the aggregated reputation to consider all interactions, which a user was involved in, in a fair way. Note that this does not mean that a reputation function considers all ratings equally, but in a fair way that allows to predict future behavior of the ratee. Technically this is difficult to define/decide, but the function must not be limited technically, hence it needs the full history of ratings. Especially, users should not be able to manipulate the aggregated reputation in a way that it neglects or emphasizes certain ratings.
5. *Liveliness of reputation:* Reputation should always consider all recent interactions or give users an indication there are no more. Especially the reputation system should not offer users the possibility to reach a final state in which bad behavior no longer damages their reputation.

Showing Reputation. The aggregated reputation of the user M can be shown to other users on request. Therefore, the following requirements apply:

6. *Availability of reputation:* As a functional requirement, inquirers need to be able to access other users' reputation; however the query process might require the consent of the user whose reputation is queried.
7. *Anonymity of enquirers:* Users want to query reputation anonymously to prevent others from building personal behavior profiles of their interests.
8. *Unlinkability of ratees:* Ratees do not want to be linked to their past interactions, except that these contributed to their reputation, to prevent others from building profiles about all their interactions and interaction partners. This means that M wants to use different pseudonyms P_M for different interactions.

Example. We consider an eBay-like marketplace where products are advertised. In such a marketplace, an interaction is a sale, which needs a seller to offer it. However, these sellers act pseudonymously, but clients want to inform themselves about the trustworthiness of the sellers. Therefore they can query a seller's reputation using the contact pseudonym indicated on the advertisement.

Registration. Every user registers under a pseudonym with a reputation provider. Because the user is able to terminate this registration the following requirement should be fulfilled:

9. *Absolute linkability of a user's registration within a reputation system:* To prevent a user from leaving with a bad reputation and re-entering with a neutral reputation, registration actions of the same user have to be absolutely linkable. We want a user to register only once in the system and he should not be able to get rid of his reputation once collected.

2.2 Attacker model

Availability of reputation (6) goes beyond the capabilities of cryptographic primitives, since it depends on functioning communication lines and hardware. In this paper we only consider it as far as protocols raise new problems, e.g., denial of service attacks that become possible because of protocol requirements.

As described in [12], *absolute linkability of a user's registration within a reputation system (9)* can be achieved by an infrastructure such as a privacy-enhancing identity management system [23].

For the remaining six requirements we distinguish two types of attackers, namely, the privacy attacker and the security attacker.

Privacy attacker. As privacy attacks we subsume attacks on *anonymity of raters (3) and enquirers (7) as well as unlinkability of ratees (8)*. We assume that reputation can be queried anonymously (e.g. by its publication on a website as it is the case for eBay) and therefore we concentrate on anonymity of raters and ratees. We assume that the privacy attacker cannot observe who is communicating with whom, that is, all users are communicating via an anonymity service. Furthermore, the attacker might collude with the reputation provider, but cannot cheat on the reputation values, that is, a honest but curious attacker. In addition, the privacy attacker can only control a limited number of users so that a sufficient large anonymity set (which contains the users not controlled by the attacker) is preserved.

Security attacker. We see the security attacker as an attacker on the *integrity (1) and authorizability of ratings (2)* and on the *fairness (4) and liveliness of reputation (5)*. We assume a global attacker who might observe all interactions between the users and between users and reputation provider, but that cannot control the reputation provider. We show in our analysis that an attacker that controls all users in the system can only forge a reputation credential if the attacker can break the underlying eCash system or forge the credential itself.

3 Analysis of Current Privacy-respecting Reputation Protocols

In this section we present existing reputation systems that make use of anonymous payment systems in order to reach *anonymity of raters (3)*. We analyze the protocols with respect to the privacy and security attackers specified in Sect. 2.2.

For the protocols presented below, as well as for our approach presented in Sect. 4, the property of coins of an anonymous payment system that they can be spent anonymously but not twice is needed. This can be used to guarantee both anonymity and authorizability of ratings. Please note that the usage of coins of an anonymous payment system does not imply that reputation becomes a currency. In order to guarantee anonymity on the network layer all communication is assumed to be anonymous by the usage of an underlying anonymizing network e.g. AN.ON [24] or Tor [25].

The reputation systems presented are applicable to arbitrary anonymous interaction systems such as the communities in our model. Both require a trusted third party, the so-called reputation provider ReP.

3.1 Bounded Above Reputation

In [26] Voss describes a protocol that requires an anonymous payment system that allows personalizing coins on generation. These coins cannot be transferred to another identity without sharing the whole secrets of this identity, but possession of a coin can be proven without authentication. Coins are used both as reputation and collateral coins. Collateral coins that a user received as guarantee are ineligible for other transactions, but can be marked as invalid to lower the spenders reputation in case of misbehavior.

Registration. When registering with the reputation provider ReP a user M receives a pseudonym P_M and a secret to prove possession of this pseudonym. The reputation provider uses this pseudonym to personalize reputation and collateral coins for P_M. M withdraws a wallet with all his coins from ReP.

Showing Reputation and Interaction. Before an interaction, M gives some of his reputation coins to his interaction partner U as collateral. U together with ReP has to verify that the coins have not been used as collateral before. Thereby M does not show his whole reputation but only a part of it necessary as collateral and that might be damaged afterwards.

Rating. After an interaction, U hands over the collateral coins received on beforehand to ReP. If U wants to give a bad rating, U asks ReP to invalidate a number of the collateral coins. If U wants to give a good rating, U asks ReP to create a number of extra coins for P_M and hand it over to P_M. ReP does this only if U has not rated P_M before.

Privacy and Security Analysis.

Privacy attacker. The reputation provider knows U and M at least pseudonymously and that they interact(ed) but does not know anything about the interaction they took part in except the collateral and reputation coins they use. *Anonymity of the rater (3)* against the ratee is only given within the set of users the ratee interacted with in the same time frame. This set typically will be small because every interaction needs collateral coins that cannot be used as reputation coins anymore as long as the interaction has not been finished and the corresponding rating has not been given. *Anonymity of enquirers (7)* can easily

be achieved by transaction pseudonyms for the interaction planned. *Unlinkability of ratees (8)* is possible because the ratee shows in every interaction only the part of his reputation needed as collateral. After the interaction these coins are invalidated by the ReP and he possibly receives new coins as new reputation.

Security attacker. We assume the reputation provider to be trusted. Then, ratings can only be given if the ratee agreed beforehand to interact with the rater, because users will only hand over collateral coins to an interaction partner if they want to take part in an interaction with him. Thereby *authorisability of ratings (2)* is guaranteed. In this protocol it cannot be guaranteed whether the actual interaction really took place or not. The protocol could be extended in a way that both interaction partners hand over collateral coins to each other in a fair exchange. This allows both interacting users to rate the other one afterwards.

The *integrity of ratings (1)* is not addressed in this protocol but should be guaranteed by authentication systems between at least U and ReP. The *fairness of reputation (4)* needs all interaction partners who received coins to contact the reputation provider and initiate invalidation of the collateral coins. To prevent certain raters from giving too many ratings to interaction partners, every user is allowed to rate every pseudonym only once. This leads to the drawback that the reputation of a user has an upper bound at the number of users using the reputation system. After this maximum is reached *liveliness of reputation (5)* is breached.

3.2 Monotonic Reputation

Androulaki et al. [6] describe a protocol that requires a trusted third party, the reputation provider ReP, who keeps accounts of reputation coins for every user. All coins are assumed to have the same non-negative value. A user U can communicate using his publicly known identity, denoted as U, or he may use a randomly chosen pseudonym P_U. Figure 2 shows a flowchart of the protocol, while the single phases are described in the paragraphs below.

Registration. Every user withdraws a wallet from ReP, which contains a number of reputation coins. Let (S, π) be one of these coins. Thereby S denotes the serial number, while π denotes the cryptographic payload of the corresponding payment system. The number of coins a user can withdraw per time unit is limited to avoid inflation.

Rating. The User U, acting as P_U, wants to rate user M, acting as P_M, after an interaction. In order to do this, P_U awards a reputation coin (S, π) to P_M. In order to dispose the received reputation coin, P_M deposits it at ReP. In exchange, P_M gets a blinded permission $blind(\sigma)$ from ReP. M unblinds this permission and sends it back to ReP so that ReP can credit this coin to his reputation account and update M's reputation.

Showing Reputation. In order to demonstrate his reputation, M requests a credential from ReP. ReP aggregates the current reputation from the ratings[3] of

[3] As outlined above, the concrete design of a function for aggregation is out of scope and needs to be chosen for a specific application.

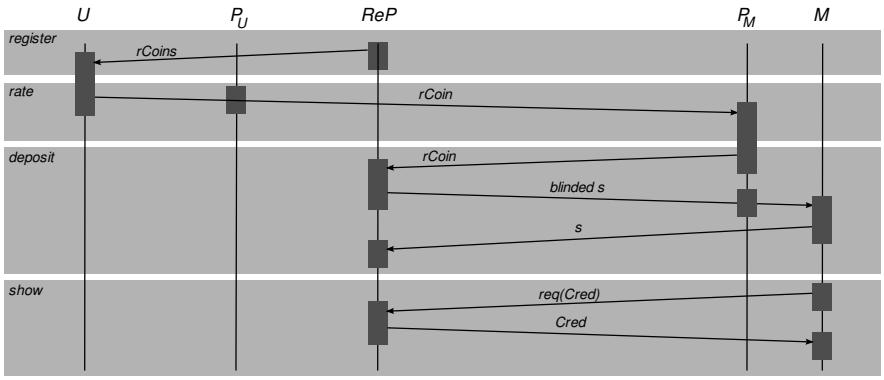

Fig. 2. The original protocol of Androulaki et al. as flowchart.

M. Then ReP issues the requested reputation credential containing M's current reputation to M. Later on, M, as P_M, can show this credential to any other pseudonymously acting user P_U.

Privacy and Security Analysis

Privacy attacker. With regards to privacy of the users, even the reputation provider ReP should not get information about who interacted with whom. However, ReP will always learn that a user was rated since it has to keep the reputation accounts.

The *rater's anonymity (3)* is based on the anonymity of coin spending and thus remains anonymous among all possible raters.

The *ratee's unlinkability (8)* is less protected. The problem is the step "deposit" (shown in Fig. 2), which consists of communication between ReP and P_M as well as ReP and M, and there is a dependency between both communications. So, the step "M sends σ to ReP" can only be performed by a M that deposited a reputation coin at ReP as P_M beforehand. As these steps will usually be performed by M without a significant time delay, ReP can decrease the set of pseudonyms that deposited a coin significantly by a timing attack. So, the ratee is only anonymous among all ratees that dispose their coin at the same time.

Furthermore, showing or querying a reputation might reveal personal information about both peers. However *anonymity of enquirers (7)* can be protected by transaction pseudonyms. The user who shows a reputation needs to be protected by the reputation system. It needs to ensure that repeated queries are not linkable, i.e., an attacker cannot tell if two reputation values are from the same user. Therefore, the reputation function must map only to a few reputation categories in order to keep the anonymity sets as large as possible.

Security attacker. The *integrity of ratings (1)* should be guaranteed by an authentication system between at least M and ReP.

Since the rater cannot give negative feedback, the reputation of the users will never decrease. Furthermore, the number of reputation values is fixed and small. Even if we do not specify a concrete reputation function here, this requirement must be met in order to restrict identifiability of users by their reputation values, see requirement *(7)*. All users will finally have the best reputation value and will keep it, thus the system reaches a final state and becomes useless, i.e., no user has an incentive to behave fairly, which violates *fairness (5) and liveliness of reputation (4)*.

Also, a decay of the reputation would not solve the problem sufficiently, since thereby inactive users are indistinguishable from misbehaving users and thus a highly active user could gather a good reputation and then misbehave for a while, but would appear as reputable as a user that was just inactive for a while.

Therefore negative feedback is needed. However, in the above protocol the ratee cannot be forced to deposit received reputation coins, i.e., the ratee can decide on his own whether he wants to deposit the received rating and of course the ratee would not deposit a negative coin. Blinding the coin value would not solve the problem either, since users usually know whether they misbehaved. However, to the best of our knowledge there is no blindeable eCash protocol proposed. In the next subsection we present a protocol that solves these problems.

4 Non-monotonic Reputation

A drawback of the reputation systems presented in Sect. 3 is that the *liveliness of reputation (5)* cannot be guaranteed, because both systems suffer from an explicit or de facto upper bound of reputation. If we allow negative ratings we will have to guarantee that these ratings cannot be suppressed by the ratee. As outlined in Sect. 1 a trusted third party can help to implement this. We propose an external reputation provider, which is described in the reminder of this section.

4.1 System Design

The reputation provider ReP keeps an interaction account and a reputation account for every user. ReP thereby guarantees that every interaction is actually rated, possibly also in a negative way, and considered for the user's reputation. We implement both accounts as accounts of an anonymous payment system and the ratings and interactions both as coins. Thereby, negative coins can be implemented by two instances of an anonymous payment protocol with a joint account, where coins of the first system are counted as +1 and coins of the second one as -1. For this, we use two instances of the protocol outlined in Sect. 3.2:

- *Interaction counter:* This instance is used to count the number of interactions a user U was involved in and should be rated for.

– *Reputation counter:* The other instance aggregates the ratings received, both positive and negative ones.

In the following paragraphs we outline the actual protocol. A flowchart of the protocol is given in Fig. 3.

Registration. In order to initialize the reputation system, every user withdraws a wallet from ReP, that contains n interaction coins (S_i, π_{it}) and reputation coins (S_i, π_{ir+}) for positive ratings and (S_i, π_{ir-}) for negative ratings. The coins are issued in triples with the same serial number S_i and π_{it}, π_{ir+} and π_{ir-} are the double-spending tacks and signatures with $i = 1 \dots n$.

Interaction. If user U wants to interact with an interaction partner M (whom U knows as P_M) using a pseudonym P_U, U starts the interaction by awarding an interaction coin (S, π_t) to P_M. P_M spends this coin to its registered pseudonym M, which deposits this coin and requests a one-show credential from the reputation provider stating the fact that the number of coins in the interaction account has been increased. P_M shows this credential to P_U. Now the actual interaction can take place. Furthermore every party needs to check the age of the coin to prevent undetectable double spending, as outlined in the analysis.

Rating. After an interaction, P_U rates P_M by awarding a reputation coin (S, π_{r+}) or (S, π_{r-}). P_M deposits (S, π_{r*}). During the deposit the reputation provider checks whether the serial number of an earlier deposited interaction coin equals the serial number of the reputation coin to avoid that M uses one of his own coins to rate himself with a positive rating instead of the (possibly negative) one received from P_U. As for the transaction coins, the age of the rating coins needs to be consistent.

Showing Reputation. If users want to show their reputation to someone, they need to request a reputation credential from the reputation provider. The reputation provider issues a reputation credential only if the interaction account and the reputation account contain the same number of coins. The reputation credential contains a time stamp to avoid that users can use old reputation credentials to show them to possible interaction partners while they misbehaved in the meantime. The reputation provider can also play the role of a global *time provider* in a very natural way by using the number of total (by every user) deposited coins as global time. This gives also an estimation how much users could cheat about their reputation, since the time difference between issuing the credential and now is the maximum number of possibly negative coins a user could have received in between.

However, highly active users might always have some open interactions and would never be able to show their reputation, hence the requirement of equal coin numbers in both accounts needs to be softer. That could be done by accepting a maximal number of missing coins or by filling up all missing coins by negative ratings; which solution is practical depends on the actual application.

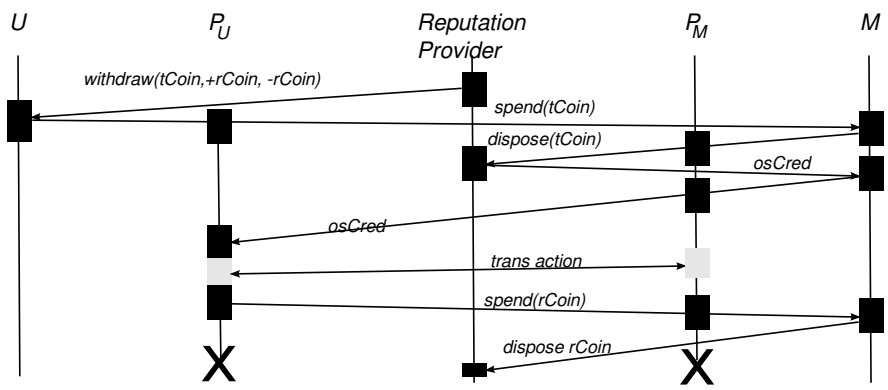

Fig. 3. The reputation granting is bound to interactions.

Batching. The protocol presented above might still raise timing issues. In order to minimize this problem, we propose to batch all user activities in rounds of three phases. In every round users get wallets with n coin triples and a sufficient amount of credentials about their reputation level, which they achieved in the round before. After that, users find their at most n interaction partners (using the credentials) and spend on them an interaction coin. In a second phase the interaction partners deposit their interaction coins and the actual interaction takes place. After the interaction the users spend on their interaction partner a reputation coin with the intended value. In the third phase all interaction partners deposit their reputation coins. In the following section we discuss the expected size of the anonymity sets of this protocol.

4.2 Privacy and Security Analysis

Availability. In the protocol presented in Sect. 4.1 the user might not give the reputation coin to the interaction partner. This blocks the ratee since the reputation provider does not issue new credentials if interaction counter and reputation counter do not contain the same number of coins. However simultaneous rating might solve the problem.

Security Attacker. The interaction registration phase depends on the security of the transferable eCash system: even if all users collude a double spending can be proven and traced back to its origin. The user U, who starts the interaction, cannot forge the interaction coin without revealing his registered user name U, since the dispose algorithm would recognize this double spending. The user M, however, might transfer the coin multiple times from P_M to M. In this case the deposit algorithm will return a proof that P_M double-spent the coin, where P_M is a non-registered pseudonym. However, since the number of hops for a coin is known, only a pseudonym controlled by M can double spent. Since M needs

to reveal its identity to the reputation provider it can get its deserved punishment in case of double spending. The argumentation for the rating is similar. These properties ensure the security properties *integrity (1) and authorisability of ratings (2)* as well as *fairness of reputation (4)*.

Privacy Attacker. Anonymity of the inquirer (7) can be guaranteed by inquiring with a one-time pseudonym or publication of P_M's reputation. The *rater's anonymity (3)* against the reputation provider is perfectly preserved by the anonymous payment system: the rater is anonymous among all the users who withdrew interaction and reputation coins during this round. The *unlinkability of the ratee (8)* M cannot be guaranteed because the disposal of the interaction coin before the interaction and the reputation coin after the interaction are in principle linkable to M. This is not a problem as long as it is assumed that ReP cannot observe any peer to peer traffic. Batching allows to relax this condition. Assume that ReP can observe which peers communicate, then ReP could link a P_U with its corresponding U if there is only one user who deposits a coin at this time. If it is assumed that many users deposit their coins at the same time, these users would be anonymous among each other. Batching allows concentrating these steps. Furthermore, batching helps to protect naive users from outside attackers who re-query the reputation of their interaction partners, since in every round the reputation of a user stays constant. However, batching is the more effective the longer the rounds are, but the longer a round is the longer a malicious node stays unpunished. The right trade-off between security and privacy depends on the application and is out of the scope of this work.

5 Conclusion

We have analyzed reputation protocols based on anonymous payment systems to enable anonymity of raters. We pointed out weaknesses of these protocols in terms of liveliness and anonymity. We have proposed a lively system, which binds ratings to interactions and we deploy transferable-eCash-based payment systems to limit timing attacks. An analysis of the security and privacy requirements is given in comparison to the existing systems.

The aim was to protect the link between interaction pseudonym P_M and registered user name M. Since P_M never communicates with the reputation provider and the anonymous payment system is assumed to be anonymous the reputation provider cannot link M and P_M unless P_M double-spent a coin.

The more interactions take place in one round the larger the anonymity set is. However, since the reputation of a user is fixed per round a user can misbehave within a round without being punished directly. Hence, the duration of a round is a trade-off between user anonymity and security. How to find this balance depends on the actual system and is out of the scope of this paper.

Finally in Table 1 we compare our system with the existing systems presented in Sect. 3.

Although the results of the analysis of our system are already quite promising for actual deployment future research is needed on denial of service prevention

		Bounded above reputation [26]	Monotonic reputation [6]	Non-monotonic reputation (this work)
1	Integrity of ratings	yes	yes	yes
2	Authorizability of ratings	yes	no	yes
3	Anonymity of raters	yes	yes	yes
4	Fairness of reputation	no	no	yes
5	Liveliness of reputation	no, upper bound	no, de facto upper bound by only-positive ratings	yes, negative ratings possible
7	Anonymity of ratees	yes	yes, but timing issues	yes, less timing issues

Table 1. Comparison of reputation protocols

and on the privacy problems caused by traffic analysis. Furthermore, the problem of self rating needs to be solved.

References

1. Resnick, P., Kuwabara, K., Zeckhauser, R., Friedman, E.: Reputation systems. Communications of the ACM **43**(12) (2000) 45–48
2. Kollock, P.: The production of trust in online markets. Advances in Group Processes **16** (1999) 99 – 123
3. Bygrave, L.: Data Protection Law, Approaching Its Rationale, Logic and Limits. Kluwer Law International, The Hague, London, New York (2002)
4. Mahler, T., Olsen, T.: Reputation systems and data protection law. In: eAdoption and the Knowledge Economy: Issues, Applications, Case Studies, Amsterdam, IOS Press (2004) 180–187
5. Voss, M., Heinemann, A., Mühlhäuser, M.: A Privacy Preserving Reputation System for Mobile Information Dissemination Networks. In: First International Conference on Security and Privacy for Emerging Areas in Communications Networks (SECURECOMM'05), IEEE (2005) 171–181
6. Androulaki, E., Choi, S.G., Bellovin, S.M., Malkin, T.: Reputation systems for anonymous networks. In: PETS '08: Proceedings of the 8th international symposium on Privacy Enhancing Technologies, Berlin, Heidelberg, Springer-Verlag (2008) 202–218
7. Steinbrecher, S., Groß, S., Meichau, M.: Jason: A scalable reputation system for the semantic web. In: Proceedings of IFIP Sec 2009, IFIP International Information Security Conference: Emerging Challenges for Security, Privacy and Trust. Volume 297 of IFIP AICT., Springer (May 2009) 421–431
8. Pingel, F., Steinbrecher, S.: Multilateral secure cross-community reputation systems. In Katsikas, S.F.S., Lioy, A., eds.: Proceedings of Trust and Privacy in Digital Business, Fifth International Conference, TrustBus. Volume 5185 of Lecture Notes in Computer Science., Springer (2008) 69–78
9. Anwar, M., Greer, J.: Reputation management in privacy-enhanced e-learning. In: The proceedings of the 3rd Annual Scientific Conference of the LORNET Research Network (I2LOR 2006). (November 2006)

10. Kinateder, M., Pearson, S.: A Privacy-Enhanced Peer-to-Peer Reputation System. In Bauknecht, K., Tjoa, A.M., Quirchmayr, G., eds.: Proceedings of the 4th International Conference on Electronic Commerce and Web Technologies (EC-Web 2003). Volume 2738 of LNCS., Prague, Czech Republic, Springer-Verlag (September 2003) 206–215
11. Pavlov, E., Rosenschein, J.S., Topol, Z.: Supporting privacy in decentralized additive reputation systems. In: The Second International Conference on Trust Management, Oxford, United Kingdom (March 2004) 108–119
12. Steinbrecher, S.: Enhancing multilateral security in and by reputation systems. In: Proceedings of the IFIP/FIDIS Internet Security and Privacy Summer School, Masaryk University Brno, 1-7 September 2008, to be published by Springer. Volume 298 of IFIP AICT., Springer (2009) 135–150
13. Dellarocas, C.: Immunizing online reputation reporting systems against unfair ratings and discriminatory behavior. In: EC '00: Proceedings of the 2nd ACM conference on Electronic commerce, New York, NY, USA, ACM Press (2000) 150–157
14. Steinbrecher, S.: Design options for privacy-respecting reputation systems within centralised internet communities. In: Proceedings of IFIP Sec 2006, 21st IFIP International Information Security Conference: Security and Privacy in Dynamic Environments. Volume 201 of IFIP., Springer (May 2006) 123–134
15. Dellarocas, C.: Research note – how often should reputation mechanisms update a trader's reputation profile? Information Systems Research **17**(3) (2006) 271–285
16. Chaum, D., Fiat, A., Naor, M.: Untraceable electronic cash. In: CRYPTO '88: Proceedings on Advances in cryptology, New York, NY, USA, Springer-Verlag New York, Inc. (1990) 319–327
17. Canard, S., Gouget, A.: Anonymity in transferable e-cash. In: Applied Cryptography and Network Security. Volume 5037 of Lecture Notes in Computer Science. (2008) 207 – 223
18. Brands, S.: A technical overview of digital credentials (1999)
19. Hansen, M., Pfitzmann, A.: Anonymity, unobservability, and pseudonymity - a proposal for terminology. Version 0.30 in: Rene Balzer, Stefan Köpsell, Horst Lazarek (Hg.): Fachterminologie Datenschutz und Datensicherheit Deutsch - Russisch - Englisch; FGI - Forschungsgesellschaft Informatik, Technische Universität Wien, Wien, Februar 2008, 111-144. Version 0.31 available from `http://dud.inf.tu-dresden.de/literatur/Anon_Terminology_v0.31.pdf` (2008)
20. ENISA: Position paper. reputation-based systems: a security analysis. available from `http://www.enisa.europa.eu/doc/pdf/deliverables/enisa_pp_reputation_based_system.pdf` (last visit 16/06/09) (2007)
21. Mui, L.: Computational Models of Trust and Reputation: Agents, Evolutionary Games, and Social Networks. PhD Thesis, Massachusetts Institute of Technology (2003)
22. Dellarocas, C.: The digitization of word-of-mouth: Promise and challenges of online feedback mechanisms. Management Science (October 2003) 1407–1424
23. Clauß, S., Pfitzmann, A., Hansen, M., Herreweghen, E.V.: Privacy-enhancing identity management. The IPTS Report **67** (September 2002) 8–16
24. Berthold, O., Federrath, H., Köpsell, S.: Web mixes: A system for anonymous and unobservable internet access. In Federrath, H., ed.: Designing Privacy Enhancing Technologies (PET'00). LNCS 2009, New York, NY, USA, Springer-Verlag (2001) 115–129

25. Dingledine, R., Mathewson, N., Syverson, P.: Tor: The second-generation onion router. In: Proceedings of the 13th USENIX Security Symposium. (August 2004) 21–21
26. Voss, M.: Privacy preserving online reputation systems. In: International Information Security Workshops, Kluwer (2004) 245–260