Saving Privacy in Trust-Based User-Centric Distributed Systems

Alessandro Aldini

Dipartimento di Scienze di Base e Fondamenti
University of Urbino “Carlo Bo”
Urbino, Italy
Email: alessandro.aldini@uniurb.it

***Abstract*—User-centricity is a design philosophy subsuming new models of Internet connectivity and resource sharing, whose development is mainly driven by what users offer and require. To promote user-centric services and collaborative behaviors, incentives are needed that are typically based on trust relations and remuneration. In this paper, we show that privacy-preserving mechanisms can favor user's involvement if privacy can be traded with trust and cost. In particular, we present and evaluate formally a model ensuring an adequate level of flexibility among privacy, trust, and cost in the setting of distributed systems.**

Keywords–Cooperation incentives; trust; privacy; remuneration; user-centric networks; model checking.

I. Introduction

Nowadays, user-driven services, like personal hotspot and peer-to-peer, are playing a fundamental role in the reshaping of the Internet value chain [1]. Essentially, they focus on the user experience, related needs, expectations, and attitude to cooperation. One of the key factors behind the success of community-scale user-centric initiatives is given by the user involvement as a *prosumer*, i.e., an actor combining the roles of service producer and consumer. Such an involvement must be strengthened through the adoption of incentive mechanisms stimulating the willingness to collaborate. In particular, even if cooperation is a natural consequence of sense of community and synergy, it cannot be taken for granted because of typical obstacles like, e.g., selfishness and, even worse, cheating, which represent a threat keeping users from trusting other community members.

Establishing trust relations among users is the objective of explicit trust and reputation systems, among which we concentrate on those aiming at providing computational estimations of user's trustworthiness as perceived by the community [2]. Basically, these estimations work effectively as an incentive to collaborate if they represent parameters influencing access to services at favorable conditions, among which we include the service cost as one of the most important aspects affecting the perceived quality of experience. At the same time, remuneration is another kind of incentive used to stimulate cooperation [3]. Whenever combined with trust, it enables a virtuous circle for the proliferation of user-centric services.

Trust is a concept that may involve and justify the collection of personally identifiable sensitive information, which in many real situations contrasts dramatically the idea of privacy and plays a deterrent role when users are getting involved in interactions. In particular, the lower the attitude to expose sensitive information is, the higher the probability of being untrusted when negotiating a service. Trading privacy for trust is thus a way for balancing the subjective value of what is revealed in exchange of what is obtained.

The above considerations suggest that an efficient cooperation infrastructure depends on the tradeoff among trust, privacy, and cost. As shown recently [4], these three dimensions can be balanced in order to favor collaborative behaviors depending on specific user's needs in terms of social (e.g., personal sensibility to trust and privacy issues) and economical (e.g., in terms of costs that can be afforded) requirements. More precisely, in the model proposed in [4], a balanced tradeoff is guaranteed by a centralized system in which reputation is managed by a trusted third party (TTP) collecting information about every transaction completed, while keeping the desired level of privacy for every user involved. In this paper, we provide a twofold contribution. On one hand, we show how to implement the model of [4] in the setting of distributed systems that cannot rely on TTP. On the other hand, we validate formally such a model through model checking based analysis [5]. This validation is done in the setting of a cooperation system that has been recently proposed to implement trust and remuneration based incentive mechanisms [6].

In the rest of this section, we comment on related work. In Section II, we briefly recall the model of [4] and then we illustrate a distributed solution for its implementation. In Section III, we estimate the validity of such a model in the setting of a real-world case study. Finally, some conclusions terminate the paper in Section IV.

A. Related Work

Making trust and service cost mutual dependent is a winning strategy if the aim is to stimulate honest behaviors while keeping users from cheats and selfishness [6]–[8], as also proved formally by means of formal methods, like game theory and model checking [9]–[13].

The contrast between privacy and trust is investigated in [14], where it is shown that these two aspects can be traded by employing a mechanism based on *pseudonyms*. In practice, users create freely pseudonyms identified by the so-called *crypto-id*, i.e., the hash of the public key of a locally generated asymmetric cryptography key pair. Then, in different environments, a user can use different pseudonyms to carry out actions logged as events signed with the private key of the chosen pseudonym. If needed to acquire more reputation, several pseudonyms can be linked together in order to augment the number of known actions and potentially increase the trust in the linked entity. Notice that in approaches such as this one the link is irrevocable.

Incentive mechanisms are proposed in [15] to achieve a balanced tradeoff between privacy and trust in the setting of data-centric ad-hoc networks. In [16], such an interplay is formulated as an optimization problem in which both privacy and trust are expressed as metrics. In [17], trust towards an entity is used to take decisions about the amount of sensitive information to reveal to the entity. Further works on unlinkability [18] and pseudonymity [19] [20] provide insights on the tradeoff between privacy and trust.

With respect to previous work, the novelty of the approach proposed in [4] is twofold. On one hand, the analysis of the tradeoff between privacy and trust takes into account also the service cost. On the other hand, it overcomes the limitations of the existing approaches, in which sensitive information linking is irrevocable and the privacy disclosure is incremental.

II. A Model for Independent Release of Privacy

In a classical view of privacy, a user exposes (part of) personal information in order to be trusted enough to get access to the service of interest. In other words, privacy disclosure is traded for the amount of reputation that the user may need to be considered as a trustworthy partner in some kind of negotiation in which, e.g., service cost may depend on trust. Typically, once different pieces of sensitive information (e.g., credentials, virtual identities, or simply the proof of being the user involved in a transaction previously conducted), say I_1 and I_2, are linked and exposed to be trusted by someone else, then such a link is irrevocably released. In this sense, we say that the disclosure of sensitive information is incremental along time.

In order to exemplify, as discussed in [14], I_1 and I_2 may identify two different transactions conducted by the user under two different pseudonyms, each one revealing different personal information about her. The user is obviously able to show that both I_1 and I_2 are associated with the same user and, if such a proof is provided, I_1 and I_2 become irrevocably linked together. As opposite to this scenario, in [4] an alternative, independent model of privacy release is proposed in which the link is not definitive. In order to work properly, such a model requires some form of uncertainty associated with the owners of specific actions. Basically, this is obtained by sharing pseudonyms among different users. Similarly as in [14], a virtual identity is represented by a crypto-id, which can be calculated using the SHA-3 cryptographic hash function over the public key of the user. Then, the basic idea of the independent model of privacy release is that trust and transactions are mapped to pieces of the crypto-id rather than to the crypto-id as a whole.

Let us explain such a mechanism through a typical handshake between Alice, who issues a service request, and Bob, who offers the service. Instead of revealing to be Alice, she accompanies the request with a portion of her crypto-id identified by applying a bitmask to the crypto-id through the bitwise AND operation. For the sake of presentation, consider a 8-bit crypto-id, e.g., 10010101, from which we obtain the portion $000\underline{1}00\underline{0}0$, called *chunk*, when applying the bitmask 00010010. Hence, a chunk is a subset of bits of the crypto-id, of which we know value and position. Amount and position of 1's occurrences in the bitmask are under Alice's control.

The transaction is then identified by the chunk chosen by Alice, together with the proof (which can be validated in Zero Knowledge) of being a proper owner of the chunk exposed. Therefore, a trust value (and related variation due to the feedback following the transaction execution) is not associated with Alice directly, but is related to the chunk of bits extracted from Alice's crypto-id through the chosen bitmask. In general, the same chunk is potentially shared by other crypto-ids belonging to several different users. In future interactions, Alice may select other chunks of her crypto-id. Moreover, she can also spend a set of different chunks of the crypto-id in order to exploit a combination of the trust levels associated with each of these chunks. Ideally, the overall trust associated with a crypto-id shall result from a combination of the trust values accumulated by every chunk of such a crypto-id spent in previous interactions. Thanks to the uncertainty relating chunks and associated owners, every time Alice exposes a chunk to Bob in order to negotiate a transaction, Bob cannot link the current transaction to any of the previous transactions conducted (by Alice or by other users) by using the same (or a portion of the current) chunk.

While in [4] the model above relies on the presence of a TTP managing chunk's reputation, in the following we tackle the problem of implementing the same idea in the setting of distributed systems without central authority and any prior knowledge about crypto-ids, which represent a more realistic scenario in several user-centric networks.

A. Design for Distributed Systems

Handling trust towards users by tracing the usage of (possibly shared) chunks is a hard task in the absence of a centralized reputation system. To deal with this problem, in order to estimate user's trustworthiness we define a local trust structure that allows any user offering a service to associate a trust value with every chunk received to negotiate the service.

Let $\mathcal{C}$ be the set of chunks with which the user has interacted in completed transactions and $\mathcal{T}$ be the trust domain, which we assume to be numeric and totally ordered. Chunks are ranged over by $C, C', \ldots$. Sometimes, we use the notation C_B to define a chunk identified by bitmask B and $C_B[i]$ (resp., $B[i]$) to denote the value of the i-th bit of the chunk (resp., bitmask). Set $\mathcal{C}$ forms a partially ordered set (*poset*, for short), $(\mathcal{C}, \leq)$, where the refinement operator $\leq$ is defined as follows.

Definition 1 (Chunk refinement): Let n be the crypto-id size. Given chunks $C_B, C_{B'}$, we say that $C_{B'}$ refines C_B, denoted $C_B \leq C_{B'}$, if and only if:

- for all $1 \leq i \leq n$: $B[i] \leq B'[i]$;
- for all $1 \leq i \leq n$: if $B[i] = 1$ then $C_B[i] = C_{B'}[i]$.

Notice that if $C_B \leq C_{B'}$ then B is a submask of B' and the information exposed by $C_{B'}$ includes that revealed by C_B. If two chunks are related through $\leq$ then they could be originated from the same crypto-id. As we will see, maintaining the poset structure provides the means to approximate the trust towards any (possibly unknown) crypto-id by employing the trust related to the potential constituting chunks.

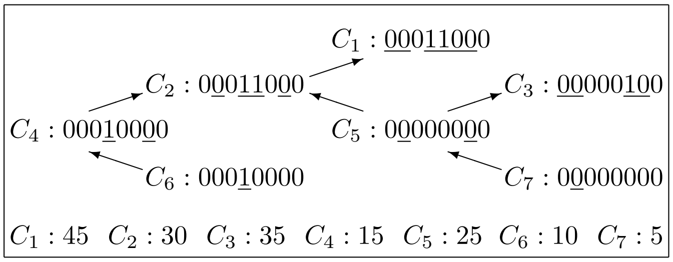

$C_1 : \underline{00}0\underline{1100}0$
$C_2 : 0\underline{0}0\underline{11}0\underline{0}0$
$C_3 : \underline{00}000\underline{10}0$
$C_4 : 000\underline{1}00\underline{0}0$
$C_5 : 0\underline{0}0000\underline{0}0$
$C_6 : 000\underline{1}0000$
$C_7 : 0\underline{0}000000$
$C_1 : 45$ $C_2 : 30$ $C_3 : 35$ $C_4 : 15$ $C_5 : 25$ $C_6 : 10$ $C_7 : 5$

Figure 1. Example of a local trust structure.

Each element of the poset $(\mathcal{C}, \leq)$ is labeled by a value of the trust domain $\mathcal{T}$. Such a value represents the trust of the user towards the related chunk resulting from interactions associated with such a chunk. Formally, we denote such an extended structure with $(\mathcal{C}, \leq, t)$, where $t : \mathcal{C} \rightarrow \mathcal{T}$ defines the mapping from chunks to trust values. Initially, for every unknown chunk C with which the user interacts for the first time, we assume $t(C)$ to be equal to the dispositional trust dt of the user, which represents the attitude to cooperate with unknown users.

Example: Figure 1, which in the following we use as running example, shows the graphical representation of a poset, where, e.g., $C_6 \leq C_4 \leq C_2 \leq C_1$, as well as $C_7 \leq C_5 \leq C_3$, while, e.g., C_6 and C_3 are not related with each other. Moreover, the figure reports also the trust associated with each known chunk at a given instant of time, by assuming the trust domain $[0, 50]$.

To emphasize the nature of the independent model of privacy release, notice that even if Alice invested chunk C_1 in a past interaction with Bob, whose reference trust structure is that depicted in Figure 1, then in the current transaction she may use chunk C_2 only, while Bob cannot infer the link between the user of the past interaction associated with C_1 and the current one. As a side effect, notice also that all the users with a crypto-id matching with C_2 actually benefit from the trust (or pay the mistrust) associated with C_2. ■

The obfuscation mechanism illustrated in the example above, which is crucial for the requirements of the independent model of privacy release, can be viewed as an additional incentive to take collaborative and honest decisions, as a high number of crypto-id chunks highly trusted contribute to increase the probability of obtaining services at a reasonable cost by preserving the desired level of privacy.

A fundamental issue for any trust system is given by the transaction feedback that induces a trust variation influencing the trust $t(C)$ towards the chunk C associated with the transaction. In our setting, it is worth observing that such a feedback should be weighted by the chunk size. More precisely, the user can decide to apply a discounting factor to the feedback result that is inversely proportional to the size of the chunk, in order to reflect that the amount of sensitive information exposed is proportional to the trustworthiness as perceived by the user.

Example: As a consequence of a positive transaction conducted through chunk C_2 and resulting in a trust variation equal to, e.g., $+5$, we would obtain $t(C_2) = 32.5$ if the discounting factor is applied, and $t(C_2) = 35$ otherwise. ■

On the other hand, it is also worth deciding whether and how the feedback related to chunk C has to be propagated to other elements of the trust structure $(\mathcal{C}, \leq, t)$. Since propagation would result in ambiguity if applied to chunks of the poset that cannot be related through $\leq$, let us examine the remaining cases. Depending on the feedback, which can be either positive or negative, and the propagation direction (towards finer or coarser chunks, or else both), every possible combination gives rise to a different propagation policy. For instance, in order to advocate a conservative policy, variations shall not be propagated to elements that refine C, because an interaction disclosing a small amount of sensitive information should not affect the trust level of chunks that expose more information. This policy contrasts also potential attacks by users preserving their identity and aiming at penalizing the trust of small chunks shared by a large number of users. On the other hand, in order to fully exploit the flexibility of the independent model of privacy release, it would be worth propagating the trust variation for C to every chunk C' in the poset that is refined by C. In this case, the trust variation for C' is discounted by a factor proportional to the difference between the size of C and the size of C'. In practice, the larger the difference between C and C' is, the slighter the impact of the trust variation of C upon C'.

Example: Consider chunk C_2 and the positive transaction of the previous example determining $t(C_2) = 32.5$. Then, by virtue of the propagation policy discussed above we have, e.g., $t(C_4) = 16.25$ and $t(C_1) = 45$. ■

As another important assumption, so far we assumed that any new chunk C that is added to the poset is initially associated with the dispositional trust of the user. Alternatively, the trust structure $(\mathcal{C}, \leq, t)$ can be employed to infer some trust information about C. Based on the same intuition behind feedback propagation, the trust values associated with known chunks that are in some relation with C can be combined. In fact, we can interpret C as an approximation of such chunks, which, however, must be pairwise unrelated by $\leq$ to avoid redundancy when counting the related trust values.

By following the conservative policy previously discussed, we initialize the trust towards C on the basis of the trust values associated with chunks that refine C.

Definition 2 (Chunk coverage): Given a trust structure $(\mathcal{C}, \leq, t)$ and a chunk $C \notin \mathcal{C}$, a coverage for C is a set $\{C_1, \ldots C_m\} \subseteq \mathcal{C}$ such that:

- $C_i \not\leq C_j$ for all $1 \leq i, j \leq m$;
- $C \leq C_i$ for all $1 \leq i \leq m$.

The initial trust associated with C by the coverage $\{C_1, \ldots C_m\}$ is $\frac{1}{m} \cdot \sum_{i=1}^{m} t(C_i)$.

Since in the poset several different coverages may exist for a chunk C, we can adopt different policies to select one of them, e.g., by choosing the coverage inducing the highest/lowest trust, or by keeping all of them and then calculating the average trust.

Example: A coverage for chunk $C_8 : 00000\underline{0}0$ is the set $\{C_4, C_5\}$, which determines the initial trust value 20. Other candidates are $\{C_2, C_3\}$, $\{C_3, C_4\}$, and $\{C_1\}$. The average trust resulting from all the possible coverages is 30.625. ■

In general, from the effectiveness standpoint, the trust structure $(\mathcal{C}, \leq, t)$ is used to manage locally information (about chunk's trust) allowing the user to approximate the trust towards other users, without any knowledge about their crypto-ids and actual behaviors. As far as efficiency issues are concerned, in order to circumvent the problem of dealing with a huge trust structure, it is possible to constrain the choice of the bitmask, e.g., by fixing a priori a rule for splitting the crypto-id into a limited set of chunks.

Finally, we emphasize that the presentation of the proposed design model abstracts away from the specific trust metric adopted. Indeed, basically, our method may be integrated with any computational notion of trust and with any recommendation mechanism used in classical trust systems for distributed environments [21] [22].

III. Formal Verification

In this section, we evaluate the proposed independent model of privacy release through a comparison with an abstraction of standard approaches in which information linking is irrevocable, in the following called incremental model of privacy release. To this aim, we employ the model checker PRISM [23] [24] [5], through which it is possible to build automatically probabilistic models – like discrete-time Markov chains and Markov decision processes – from state-based formal specifications. On the semantic models deriving from formal descriptions, quantitative properties expressed in probabilistic extensions of temporal logics are verified through model checking techniques.

The comparison is conducted by assuming that the two models of privacy release are applied in the setting of a real-world cooperation system [6], in which users providing services, called *requestees*, and recipients of such services, called *requesters*, are involved in a cooperation process balancing trustworthiness of each participant with access to services and related costs. In the following, we omit the specification of the formal description given in the PRISM modeling language and we briefly introduce the original trust model and its relation with service remuneration [6]. Then, we describe our modeling assumptions and the metrics that are used to evaluate how trading privacy for trust influences access to services and related costs. We finally discuss the obtained results.

A. Trust Model

Trust is a discrete metric with values ranging in the interval $[0, 50]$, such that $null = 0$, $low = 10$, $med = 25$, and $high = 40$. The trust T_{ij} of user i towards any credential j (which can be, e.g., a crypto-id chunk or an entity identity) is modeled abstractly as follows:

$$T_{ij} = \alpha \cdot trust_{ij} + (1 - \alpha) \cdot recs_{ij} \tag{1}$$

Parameter $\alpha \in [0, 1]$ is the risk factor balancing personal experience with recommendations by third parties. The trust metric $trust_{ij}$ is the result of previous direct interactions of i with j. Initially, $trust_{ij}$ is set to the dispositional trust of i, denoted by dt_i. After each positive interaction, $trust_{ij}$ is incremented by a factor v. Parameter $recs_{ij}$ is the average of the trust metrics towards j recommended to i by other users. For each service type, the service trust threshold *st* represents the minimum trust required to negotiate the service.

B. Service Cost Model

The joint combination of trust and remuneration is implemented by making the service cost function dependent on the trust T of the requestee towards the requester credential. The other main parameters are: C_{min}, which is the minimum cost asked by the requestee regardless of trust, C_{max}, which is the maximum cost asked to serve untrusted requests, and the threshold values T' and T'', such that $T'' < T'$.

The cost function proposed in [6] expresses linear dependence between trust and cost:

$$C(T) = \begin{cases} C_{min} + \frac{C_{max} - C_{min}}{T'} \cdot (T' - T) & \text{if } T < T' \\ C_{min} & \text{otherwise} \end{cases} \tag{2}$$

In order to examine thoroughly the trust/cost tradeoff, we consider two more functions approximating the linearity of the relation between trust and cost. In particular, a simple one-step function is as follows:

$$C(T) = \begin{cases} C_{max} & \text{if } T < T' \\ C_{min} & \text{otherwise} \end{cases} \tag{3}$$

while a possible two-steps function is as follows:

$$C(T) = \begin{cases} C_{max} & \text{if } T < T'' \\ C_{max}/2 & \text{if } T'' \leq T < T' \\ C_{min} & \text{otherwise} \end{cases} \tag{4}$$

C. Modeling Assumptions

Our objective is to compare the model of incremental release of privacy (represented in the figures by the curves named inc) with the model of independent release of privacy (represented in the figures by the curves named ind). For the sake of uniformity, for both models we assume abstractly that privacy is released (through the pseudonyms mechanism [14] and through the chunk mechanism, respectively) as a percentage of the total amount of sensitive information that the user may disclose. Similarly, in every trust-based formula we consider percentages of the trust involved.

The experiments are conducted by model checking several configurations of the system against formulas expressed in quantitative extensions of Computation Tree Logic [5]. For instance, Figure 2 refers to one requester interacting with one requestee with the aim of obtaining 10 services that can be of three different types. The figure reports the results for the best strategy, if one exists, allowing the requester to get access to all the services requested by minimizing the total expected cost (reported on the vertical axis) depending on the amount of revealed sensitive information (reported on the horizontal axis). The choice of the amount of privacy to spend for each request is under the control of the requester. The choice of the service type is either governed by the requester, or it is probabilistic with uniform distribution (see the curves denoted by $prob$ in the figure). Requestee's parameters are $dt = med$ and $v = 5$, as we assume that each transaction induces a positive feedback. The three service types are characterized by $st_1 = null$ and (2), $st_2 = low$ and (3), $st_3 = med$ and (4), respectively. The service cost parameters are $C_{min} = 0$, $C_{max} = 10$, $T' = high$, and $T'' = med$.

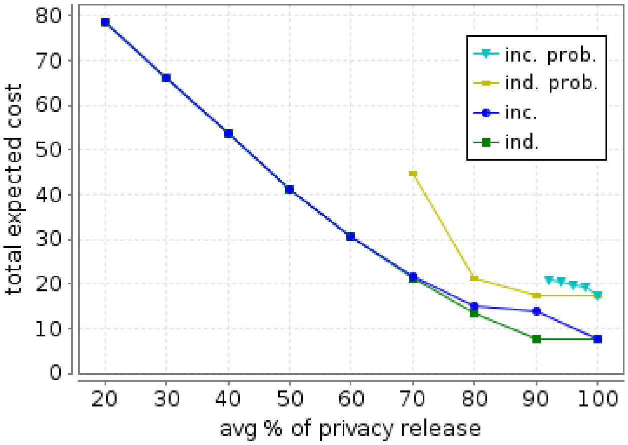


Figure 2. Trading cost for privacy.

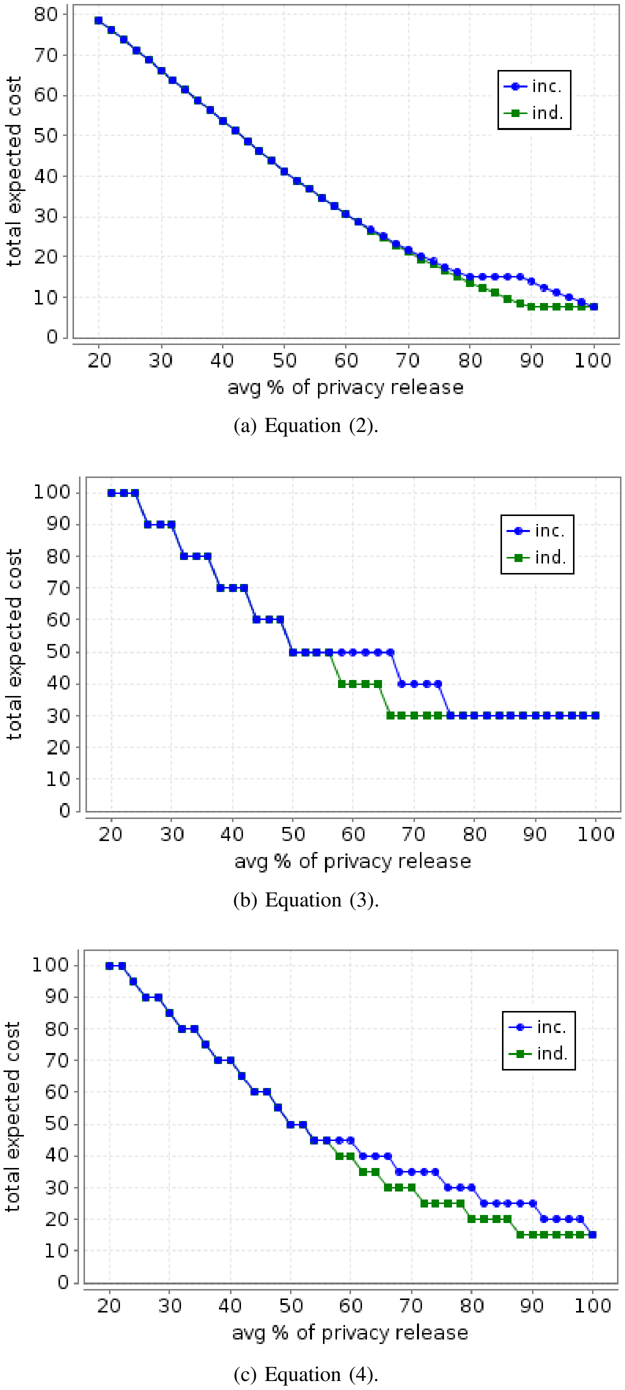
We complete the comparison with an experiment assuming one requester and two requestees, which are chosen nondeterministically by the requester. The number of issued requests is 10, while we consider only the first type of service. The analysis, reported in Figure 3, proposes the results obtained by changing the service cost function. Requestee's trust parameters are as follows: $dt = med$, $st = null$, $\alpha = 0.5$.

D. Discussion

We now comment on the obtained results, by first considering Figure 2, which reveals two interesting behaviors.

Firstly, if the choice of the service is under the control of the requester, then the difference between the two models is significant only for values of the privacy release higher than 70%. In order to interpret this result, we checked the best requester's strategy, which consists of choosing always the service offering the best ratio trust/cost, i.e., the one using (2). Whenever trust is high enough to apply the minimum cost, then it turns out to be convenient to select also the other two service types. According to this strategy, if the privacy disclosure is below 70% it happens that trust does not reach the threshold T'. Therefore, as a consequence of (2), the relation between trust and cost is always linear and the two privacy models turn out to be equivalent from the economic standpoint. On the other hand, if the requester is highly trustworthy, then the cost to pay becomes constantly equal to the minimum cost, meaning that the requester could invest less privacy to obtain the same cost, thus revealing the advantages of the independent model. In practice, independently of the privacy model, it is economically convenient for the requester to disclose the information needed to obtain rapidly the best cost. Instead, for high levels of trust, it would be convenient for requester's privacy to reduce as much as possible the amount of disclosed information. Whenever identity of the requester is always fully disclosed, then the two models experience the same performance.

Secondly, if the choice of the service is probabilistic, thus modeling, e.g., a situation in which the requester may require every type of service independently of their cost, then it is not possible to satisfy all the requests if a minimum disclosure of privacy is not guaranteed. However, such a minimum value is considerably higher for the incremental model, in which case at least an average privacy release of 92% is needed. Hence, if the requester is somehow forced to require certain services, then the independent model performs better.



(a) Equation (2).

(b) Equation (3).

(c) Equation (4).

Figure 3. Trading cost for privacy by varying cost function.

The role of the service cost function is emphasized by the curves of Figure 3, which show that whenever a step function is used, the independent model is able to exploit better the intervals of trust in which the service cost is constant.

In the previous experiments, priority is given to cost and to the average disclosure of privacy needed to optimize such a cost. However, if cost is not a fundamental issue, then the tradeoff of interest concerns trust and privacy. In order to analyze such a tradeoff, we reformulate the experiment of Figure 2 by focusing on the optimization of the average percentage of privacy release needed to obtain 10 services of a given type. In particular, we consider the second and third service types, for which the service trust threshold is *low* and *med*, respectively. Since to obtain such services the requester must be trusted by the requestee, we examine the tradeoff between such a trust and requester's privacy. For the second (resp., third) service type, the average percentage of privacy release is 38% (resp., 92%) when applying the incremental model, while it is equal to 28% (resp., 64%) in the case of the independent model. Therefore, the observed values show that through the independent model we obtain all the required services by disclosing much less privacy than through the incremental model. The related difference is directly proportional to the trust threshold needed to obtain the services.

IV. Conclusion

The attitude to cooperation is strongly affected by the tradeoff existing among privacy and trustworthiness of the involved parties and cost of the exchanged services. In order to balance the related incentive mechanisms, it is worth considering the constraints of the model of privacy release. Thanks to a mechanism based on the splitting of crypto-ids, it is possible to manage the disclosure of sensitive information in a less restrictive way with respect to classical models, even in distributed environments.

The formal evaluation has emphasized that the flexibility of the independent model ensures better performance with respect to the incremental model. This is always true if the main objective is trading privacy for trust. If services must be paid and cost depends on trust, then the adopted cost function affects the tradeoff among privacy, trust, and cost, by revealing the advantages of the independent model in the intervals of trust values in which cost is constant.

As work in progress, the integration of the proposed distributed trust system with the centralized reputation system of [4] is under development. Moreover, a successful deployment of the proposed model is strictly related to the choice of the trust policies and configuration parameters discussed in Section II, which are currently subject to sensitive analysis through formal verification.

References

[1] A. Aldini and A. Bogliolo, Eds., User-Centric Networking – Future Perspectives, ser. Lecture Notes in Social Networks. Springer, 2014.

[2] A. Jøsang, "Trust and reputation systems," in Foundations of Security Analysis and Design IV (FOSAD'07), ser. LNCS, A. Aldini and R. Gorrieri, Eds. Springer, 2007, vol. 4677, pp. 209–245.

[3] S. Greengard, "Social games, virtual goods," Communications of the ACM, vol. 54, no. 4, 2011, pp. 19–22.

[4] A. Aldini, A. Bogliolo, C. Ballester, and J.-M. Seigneur, "On the tradeoff among trust, privacy, and cost in incentive-based networks," in 8th IFIP WG 11.11 Int. Conf. on Trust Management, ser. IFIP AICT, J. Zhou et al., Eds., vol. 430. Springer, 2014, pp. 205–212.

[5] V. Forejt, M. Kwiatkowska, G. Norman, and D. Parker, "Automated verification techniques for probabilistic systems," in Formal Methods for Eternal Networked Software Systems, ser. LNCS, M. Bernardo and V. Issarny, Eds. Springer, 2011, vol. 6659, pp. 53–113.

[6] A. Bogliolo et al., "Virtual currency and reputation-based cooperation incentives in user-centric networks," in 8th Int. Wireless Communications and Mobile Computing Conf. (IWCMC'12). IEEE, 2012, pp. 895–900.

[7] Y. Zhang, L. Lin, and J. Huai, "Balancing trust and incentive in peer-to-peer collaborative system," Journal of Network Security, vol. 5, 2007, pp. 73–81.

[8] M. Yildiz, M.-A. Khan, F. Sivrikaya, and S. Albayrak, "Cooperation incentives based load balancing in UCN: a probabilistic approach," in Global Communications Conf. (GLOBECOM'12). IEEE, 2012, pp. 2746–2752.

[9] Z. Li and H. Shen, "Game-theoretic analysis of cooperation incentives strategies in mobile ad hoc networks," IEEE Transactions on Mobile Computing, vol. 11, no. 8, 2012, pp. 1287–1303.

[10] A. Aldini and A. Bogliolo, "Model checking of trust-based user-centric cooperative networks," in 4th Int. Conf. on Advances in Future Internet (AFIN2012). IARIA, 2012, pp. 32–41.

[11] A. Aldini, "Formal approach to design and automatic verification of cooperation-based networks," Journal On Advances in Internet Technology, vol. 6, 2013, pp. 42–56.

[12] M. Kwiatkowska, D. Parker, and A. Simaitis, "Strategic analysis of trust models for user-centric networks," in Int. Workshop on Strategic Reasoning (SR'13), vol. 112. EPTCS, 2013, pp. 53–60.

[13] A. Aldini and A. Bogliolo, "Modeling and verification of cooperation incentive mechanisms in user-centric wireless communications," in Security, Privacy, Trust, and Resource Management in Mobile and Wireless Communications, D. Rawat, B. Bista, and G. Yan, Eds. IGI Global, 2014, pp. 432–461.

[14] J.-M. Seigneur and C.-D. Jensen, "Trading privacy for trust," in 2nd Int. Conf. on Trust Management (iTrust'04), ser. LNCS, vol. 2995. Springer, 2004, pp. 93–107.

[15] M. Raya, R. Shokri, and J.-P. Hubaux, "On the tradeoff between trust and privacy in wireless ad hoc networks," in 3rd ACM Conf. on Wireless Network Security (WiSec'10), 2010, pp. 75–80.

[16] L. Lilien and B. Bhargava, "Privacy and trust in online interactions," in Online Consumer Protection: Theories of Human Relativism. IGI Global, 2009, pp. 85–122.

[17] W. Wagealla, M. Carbone, C. English, S. Terzis, and P. Nixon, "A formal model of trust lifecycle management," in Workshop on Formal Aspects of Security and Trust (FAST'03), 2003.

[18] S. Köpsell and S. Steinbrecher, "Modeling unlinkability," in 3rd Workshop on Privacy Enhancing Technologies, ser. LNCS, vol. 2760. Springer, 2003, pp. 32–47.

[19] I. Goldberg, "A pseudonymous communications infrastructure for the internet," Ph.D. dissertation, University of California at Berkeley, 2000.

[20] A. Kobsa and J. Schreck, "Privacy through pseudonymity in user-adaptive systems," ACM Transactions on Internet Technology, vol. 3, no. 2, 2003, pp. 149–183.

[21] S.-D. Kamvar, M.-T. Schlosser, and H. Garcia-Molina, "The eigentrust algorithm for reputation management in p2p networks," in 12th Conf. on World Wide Web (WWW'03). ACM, 2003, pp. 640–651.

[22] R. Zhou and K. Hwang, "Powertrust: a robust and scalable reputation system for trusted peer-to-peer computing," IEEE Transactions on Parallel and Distributed Systems, vol. 18, no. 4, 2007, pp. 460–473.

[23] T. Chen, V. Forejt, M. Kwiatkowska, D. Parker, and A. Simaitis, "Prism-games: a model checker for stochastic multi-player games," in 19th Int. Conf. on Tools and Algorithms for the Construction and Analysis of Systems (TACAS'13), ser. LNCS, vol. 7795. Springer, 2013, pp. 185–191.

[24] M. Kwiatkowska, G. Norman, and D. Parker, "Prism 4.0: verification of probabilistic real-time systems," in 23rd Int. Conf. on Computer Aided Verification (CAV'11), ser. LNCS, vol. 6806. Springer, 2011, pp. 585–591.