Enabling Trustworthy Service Evaluation in Service-Oriented Mobile Social Networks

Xiaohui Liang, *Student Member*, *IEEE*, Xiaodong Lin, *Member*, *IEEE*, and
Xuemin (Sherman) Shen, *Fellow*, *IEEE*

Abstract—In this paper, we propose a Trustworthy Service Evaluation (TSE) system to enable users to share service reviews in service-oriented mobile social networks (S-MSNs). Each service provider independently maintains a TSE for itself, which collects and stores users' reviews about its services without requiring any third trusted authority. The service reviews can then be made available to interested users in making wise service selection decisions. We identify three unique service review attacks, i.e., linkability, rejection, and modification attacks, and develop sophisticated security mechanisms for the TSE to deal with these attacks. Specifically, the basic TSE (bTSE) enables users to distributedly and cooperatively submit their reviews in an integrated chain form by using hierarchical and aggregate signature techniques. It restricts the service providers to reject, modify, or delete the reviews. Thus, the integrity and authenticity of reviews are improved. Further, we extend the bTSE to a Sybil-resisted TSE (SrTSE) to enable the detection of two typical sybil attacks. In the SrTSE, if a user generates multiple reviews toward a vendor in a predefined time slot with different pseudonyms, the real identity of that user will be revealed. Through security analysis and numerical results, we show that the bTSE and the SrTSE effectively resist the service review attacks and the SrTSE additionally detects the sybil attacks in an efficient manner. Through performance evaluation, we show that the bTSE achieves better performance in terms of submission rate and delay than a service review system that does not adopt user cooperation.

Index Terms—Mobile social networks, trust evaluation, sybil attack, distributed system

1 Introduction

Service-oriented mobile social networks (S-MSNs) [1], [2], [3] are emerging social networking platforms over which one or more individuals are able to communicate with local service providers using handheld wireless communication devices such as smartphones. In the S-MSNs, service providers (restaurants and grocery stores) offer location-based services to local users and aim to attract the users by employing various advertising approaches, for example, sending e-flyers to the nearby passengers via wireless connections. Unlike the global counterparts, the interests of the local service providers are in serving the users in close geographic vicinity because most users choose services based on the comparison of the service quality and the distance advantage. In the S-MSNs, to establish the trust relations between the service providers and the users is particularly important. With a higher reputation, a service provider is more likely to be chosen by the users. However, the S-MSNs are autonomous and distributed networks where no third trusted authority exists for bootstrapping the trust relations. Therefore, for the users in the S-MSNs, how to enable the trust evaluation of the service providers is a challenging problem.

Trustworthy service evaluation (TSE) systems [4], [5] enable service providers or any third trusted authority to receive user feedback, known as service reviews or simply reviews, such as compliments and complaints about their services or products. By using the TSE, the service providers learn the service experiences of the users and are able to improve their service strategy in time. In addition, the collected reviews can be made available to the public, which enhances service advertising and assists the users in making wise service selections. The TSE is often maintained by a third trusted authority who is trusted to host authentic reviews. Popular TSE can be found in web-based social networks such as Facebook and online stores like eBay. They are important marketing tools for service providers who target the global market. In this paper, we move the TSE into the S-MSN settings. We require service providers to maintain the TSE by themselves. In the meantime, we consider the users participate in the TSE in a cooperative manner. We will study possible malicious behaviors conducted by the service providers and the users. For ease of presentation, we refer to service providers as vendors in the sequel.

We consider an S-MSN composed of static vendors and mobile users that interconnect opportunistically. Each vendor is equipped with a wireless communication device that has a large storage space [4]. In the TSE, the vendor stores and disseminates service information to the users. Note that the adoption of the TSE is subject to vendors' own

- *X. Liang and X. Shen are with the Department of Electrical and Computer Engineering, University of Waterloo, 200 University Avenue West, Waterloo, ON N2L 3G1, Canada. E-mail: {x27liang, xshen}@bbcr.uwaterloo.ca.*
- *X. Lin is with the Faculty of Business and Information Technology, University of Ontario Institute of Technology, Oshawa, ON L1H 7K4, Canada. E-mail: xiaodong.lin@uoit.ca.*

Manuscript received 17 Sept. 2012; revised 23 Dec. 2012; accepted 11 Jan. 2013; published online 14 Feb. 2013.
Recommended for acceptance by X. Li, P. McDaniel, R. Poovendran, and G. Wang.
For information on obtaining reprints of this article, please send e-mail to: tpds@computer.org, and reference IEEECS Log Number TPDSSI-2012-09-0959.
Digital Object Identifier no. 10.1109/TPDS.2013.37.

1045-9219/14/$31.00 © 2014 IEEE Published by the IEEE Computer Society

decisions. However, the users expect to read comprehensive and authentic reviews of services, and this expectation makes vendors who support the TSE appear more attractive than the others.

Without in-network third trusted authorities in the S-MSN, vendors are required to manage reviews for themselves. This requirement brings unique security problems to the review submission process. For example, vendors may reject or delete negative reviews and insert forged positive ones, and the malicious users can leave false negative reviews or drop the reviews from others to decrease the reputation of some particular vendors. In the design of the TSE for the S-MSN, security mechanisms must be included to resist these attacks. Notorious sybil attacks [5], [6], [7], [8] also cause huge damage to the effectiveness of the TSE. The multiple pseudonym techniques [9] are generally adopted in many distributed networking systems for privacy preservation of the identities and locations of users. On the one hand, users are able to frequently change their pseudonyms to prevent the linkage of their behaviors at different time/location. Their behavior cannot be tracked and their personal information cannot be disclosed. As a result, they are more willing to use mobile applications. On the other hand, in trust systems like the TSE, if users abuse their pseudonyms to leave reviews toward a vendor, the reputation of the vendor can be easily increased or decreased. Even if a trusted authority later identifies the malicious behavior, the detection delay cannot be tolerated in the TSE. It is necessary to tackle how to resist the sybil attacks and guarantee both review integrity and review authenticity in the design of the TSE for the S-MSN.

Our contributions can be summarized as follows: We propose a basic trustworthy service evaluation (bTSE) system and an extended Sybil-resisted TSE (SrTSE) system for the S-MSNs. In both systems, no third trusted authorities are involved, and the vendor locally maintains reviews left by the users. The vendor initializes a number of tokens, which are then circulated among the users to synchronize their review submission processes. After being serviced by a vendor, a user generates and submits a nonforgeable review to the vendor. The user cannot proceed with the review submission until it receives a token from the vendor. If the review submission succeeds, the user will forward the token to a nearby user who is wishing to submit a review to the same vendor; otherwise, the user will forward both the token and its own review to the receiver, expecting that receiver user will cooperate and submit their reviews together. During token circulation, a hierarchical signature technique [10], [11] is adopted to specify and record each forwarding step in the token, and a modified aggregate signature technique [12] is employed to reduce token size. Both signature techniques are also used during cooperative review submission for reducing communication overhead and improving review integrity. Specifically, we identify three unique review attacks, i.e., review linkability attack, review rejection attack, and review modification attack in the bTSE. We also introduce two typical sybil attacks, which cause huge damage to the bTSE. Under the sybil attacks, the bTSE system cannot work as expected because a single user can abuse the pseudonyms to generate multiple unlinkable false reviews in a short time. To resist such attacks, in the SrTSE, the pseudonyms are embedded with a trapdoor; if any user leaves multiple false reviews toward a vendor in a predefined time slot, its real identity will be revealed to the public. Through the security analysis and numerical results, we show that both the bTSE and the extended SrTSE are secure against the possible attacks. We further evaluate the performance of the proposed bTSE in comparison with a noncooperative (NCP) system that does not engage cooperative review submission. Simulation results indicate that the bTSE achieves significantly (up to 100 percent) higher submission rates (SRs) in the presence of the review rejection attacks, and (up to 75 percent) lower submission delays (SDs) in general than the NCP system, at the cost of reasonable cooperation overhead.

The remainder of this paper is organized as follows: Section 2 summarizes the related work in literature. Section 3 defines the frequently-used notations, the network model, and the security model. The system details of the bTSE and the SrTSE are presented in Section 4. The security analysis on the review attacks and the sybil attacks are given in Section 5. The numerical results of detecting the sybil attacks are provided in Section 6. The performance evaluation is shown in Section 7, respectively. Section 8 finally concludes the paper.

2 Related Work

Location-based services recently emerge as an imperative need of mobile users. It can be integrated into various types of networks to obtain promising applications while their implementation has many outstanding and independent research issues, such as content delivery [13], service discovery [14], security, and privacy problems [15]. Trust evaluation of service providers is a key component to the success of location-based services in a distributed and autonomous network. Location-based services require a unique and efficient way to impress the local users and earn their trust so that the service providers can obtain profits [16]. Rajan and Hosamani [17] used an extra monitor deployed at the untrusted vendor's site to guarantee the integrity of the evaluation results. Wang and Li [18] proposed a two-dimensional trust rating aggregation approach to enable a small set of trust vectors to represent a large set of trust ratings. Aydey and Fekri [19] approached the trust management as an inference problem and proposed a belief propagation algorithm to efficiently compute the marginal probability distribution functions representing reputation values. Das and Islam [20] introduced a dynamic trust computation model to cope with the strategically altering behavior of malicious agents. In this paper, we enable mobile users to submit their reviews to a system maintained by the local vendor, where the reviews represent the evaluation results toward the services of the vendor. Similar to [17], [20], we consider the malicious behaviors by the vendor and the users including the review attacks and the sybil attacks. Instead of using an extra monitor device on the vendor's site, we explore user cooperation efforts and make use of efficient cryptography-based techniques to increase SR, reduce SD, and mitigate the effect of the malicious behaviors.

Distributed systems are vulnerable to *sybil attacks*, where an adversary manipulates bogus identities or abuse pseudonyms to compromise the effectiveness of the systems. For example, in the peer-to-peer networks, Douceur [21] indicated that the sybil attacks can compromise the redundancy of distributed storage systems. In the sensor networks, Karlof and Wagner [22] showed that the sybil attacks can damage the routing efficiency. Newsome et al. [6] proposed many defense mechanisms, such as, radio resource testing, key validation for random key predistribution, and position verification. In vehicular ad hoc networks, Lu et al. [8] proposed an efficient detection mechanism on double registration, which can be conducted to mitigate the possible sybil attacks. The sybil attacks in social networks have attracted great attention recently [23], [24], [25]. In social networks, Wei et al. [25] mentioned the existence of a trusted authority can mitigate the effect of the sybil attacks, but they considered that such requirements impose additional burdens on users which is not acceptable. In this paper, we study the sybil attacks in the S-MSNs, where the registered users can legally apply for multiple pseudonyms and alternatively use the pseudonyms for preserving their identity and location privacy. In the meantime, the lack of the in-network third trusted authority makes it very difficult to detect the sybil attacks. We identify two typical types of the sybil attacks, propose a sophisticated pseudonym design, and built the SrTSE based on the bTSE [2] to resist the two sybil attacks.

3 Notations and Models

In this section, we describe the network model and security model where the TSE will be developed.

3.1 Network Model

An S-MSN contains multiple vendors offering different or similar services to users. Because each vendor maintains the TSE independently for itself, without loss of generality we consider an S-MSN with a single vendor. There is no third trusted authority in the network. For simplicity, the vendor is assumed to offer a single service. However, the TSE may be trivially extended to multivendor multiservice scenarios by assigning unique identifiers to different vendors and services.

The vendor is equipped with a wireless communication device that has a large storage space. Each user has a handheld device such as cell phone; the transmission range of the device is the same for all users, and smaller than the vendor's transmission range. From a social perspective [26], users spontaneously form different social groups based on their common interests, termed as "attributes." Suppose that there are p social groups $\{g_1, \ldots, g_p\}$ and π users $\{u_1, \ldots, u_\pi\}$. Let $\mathcal{A}_u$ be the universal attribute set. Denote a social group g_h's attribute set by $\mathcal{A}_h$ ($\mathcal{A}_h \subseteq \mathcal{A}_u$) for $1 \leq h \leq p$. Every user u_j belongs to at least one social group. It inherits the attributes of the social groups that it belongs to. Thus, the attribute set of u_j is $\mathcal{P}_j = \bigcup_{h \in \mathcal{H}} \mathcal{A}_h$, where u_j is a member of g_h. The vendor (precisely, its service) is also tagged by an attribute set $\mathcal{V} \subseteq \mathcal{A}_u$. Each group g_h relies on a group authority c_h for membership management. c_h has a public/private key pair (pk_h, sk_h), and publishes the public key to all users. A multiauthority identity-based signature scheme [12] is used to implement group membership. Note that c_h is not a part of the network, and the management of group membership is performed offline. Every user u_j has a private unique identity id_j. When it joins g_h, c_h verifies the validity of u_j's identity id_j and assigns u_j a number of randomly generated pseudonyms $pid_{j,h,1}, pid_{j,h,2}, \ldots$. These pseudonyms contain the group information and can be linked to g_h. Thus, reviews are associated with pseudonyms, which in turn belong to social groups. c_h also sends u_j a number of secret keys $psk_{j,h,*}$, each corresponding to $pid_{j,h,*}$.

3.2 Security Model

Due to the lack of centralized control, the S-MSN is vulnerable to various security threats. The group authorities are trusted but not a part of the network. In the following, we describe several malicious attacks that aim particularly at the TSE.

Review attack 1. Review linkability attack is executed by malicious users, who claim to be members of a specific group, but disable the group authority to trace the review back to its unique identity, thus breaking review linkability.

Review attack 2. Review rejection attack is launched by the vendor when a user submits a negative review to it. In the attack, the vendor drops the review silently without responding to the submission request from the user, and hides public opinions and mislead users.

Review attack 3. Review modification attack is performed by the vendor toward locally stored review collections. The vendor inserts forged complimentary reviews, or modifies/deletes negative reviews in a review collection. Such attacks aim at false advertising by breaking review integrity and influencing user behaviors.

In addition, we consider attacks where legitimate users generate false reviews. As reviews are subjective in nature, it is difficult to determine whether the content of an authentic review is false or not. However, the TSE must prevent the sybil attacks, which subvert the system by creating a large number of pseudonymous entities, using them to gain a disproportionately large influence. Since the TSE assigns multiple pseudonyms to a registered user, the sybil attacks can easily happen in the TSE as follows:

Sybil attack 1. Such an attack is launched by malicious users: One registered user leaves multiple reviews toward a vendor in a time slot, where the reviews are false and negative to the service.

Sybil attack 2. Such an attack is launched by malicious vendors with colluded users: A malicious vendor asks one registered user to leave multiple reviews toward itself in a time slot, where the reviews are positive to the service.

The above two sybil attacks produce inaccurate information, which is unfair to either vendors or users, and disrupt the effectiveness of the TSE. The linkability of reviews ensures that the reviews can be linked to real identities by the group authorities. However, the group authorities are not part of the network, and the detection of the sybil attacks by the group authorities is inefficient and probably with huge delay. To this end, we propose another security mechanism to effectively resist the sybil attacks by restricting each user to generate only one review toward a

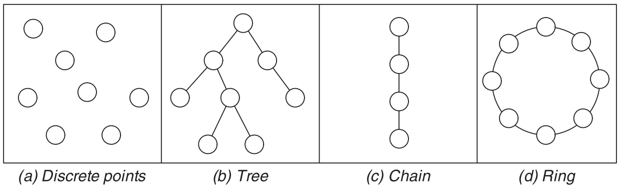


Fig. 1. Basic review structures.

vendor in a predefined time slot. If any user generates two or more than two reviews with different pseudonyms toward a vendor in a time slot, its real identity will be exposed to the public.

Note that, restricting the number of reviews per each user in a time slot can limit the sybil attacks. However, any user can still generate false reviews using multiple pseudonyms for different time slots, and the reviews cannot be linked immediately. Since reviews are linked to the social groups, false reviews will damage group reputation in a long run. Group reputation can, therefore, be taken as a weighting factor for the reviews generated by the group members. To further mitigate the effect by the false reviews, users may also make their service selection decisions based on the group reputation.

4 The Design of TSE

We present the bTSE based on the above-defined models. In the bTSE, a user, after being serviced by the vendor, submits a review to the vendor, which then stores the review in its local repository. The review consists of two parts: (α, σ), where α is the review content and σ the signature proving the authenticity of the content. Review submission may need cooperations from other users; the user forwards its review to a nearby user who wants to submit a review to the same vendor and expects that user to submit their reviews together. User cooperation increases SR and reduces SD at the cost of additional transmission efforts.

During review submission, data integrity, authenticity, and nonrepudiation can be obtained by directly applying traditional cryptography techniques such as hashing and digital signature on review content. As these techniques are widely discussed, we do not detail them here. However, it is challenging to resist the three review attacks and the two sybil attacks introduced in Section 3.2.

4.1 Structured Reviews

In the bTSE, reviews are structured to reflect their adjacency (i.e., submission order) through user cooperation. As such, vendors simply rejecting or modifying reviews will break the integrity of the review structure, thus being detected by the public. Consider a collection of n reviews received by a vendor v. We define four basic review structures (as illustrated in Fig. 1) and indicate vendors' review modification capabilities corresponding to them.

In Fig. 1a, reviews appear as discrete points, meaning that they are submitted separately and independent of each other. This independence gives the vendor maximum capability of manipulating the n reviews, and its modification capability is therefore $O(n)$. A logarithm modification capability is shown in Fig. 1b, where the reviews are presented in a tree-like structure. In this scenario, v is able to delete any single review corresponding to the leaf node, and the number of such reviews is $O(\log n)$. Figs. 1c and 1d exhibit a chain structure and a ring structure. They, respectively, lead to constant $O(1)$ and zero modification capabilities. Clearly, the strength of the modification capabilities follows the order of $O(n) > O(\log n) > O(1) > 0$.

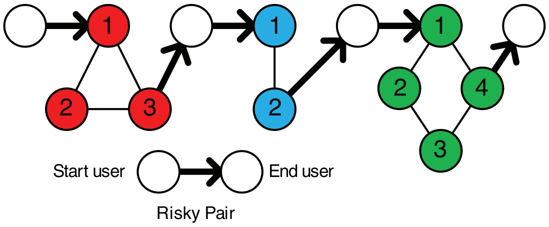


Fig. 2. A hybrid review structure.

To form a ring structure requires extensive cooperation efforts from users, i.e., the first user that submitted a review must be aware of the pseudonyms of the users who are going to submit reviews subsequently. Such an assumption in the decentralized S-MSN is unrealistic. Therefore, in the bTSE, we adopt a hybrid structure (chain and ring), as shown in Fig. 2, to limit the modification capability of the vendor below $O(1)$. Because this structure has a chain as its skeleton, in the sequel we refer to it as "chain" for ease of our presentation.

4.2 Synchronization Tokens

The chain structure requires reviews to be submitted sequentially. The bTSE uses a token technique to synchronize review submission. The vendor spontaneously initializes a number of tokens and issues them to distinct users, one per user. The tokens will then be circulated among users according to their local decision on token forwarding. A user cannot submit a review unless it currently holds one of the tokens. A token may be lost due to user mobility or malicious dropping. The vendor considers a token lost if it has not received any review submission associated to the token for a predefined maximum time duration θ_{exp}. It replaces lost tokens with new ones so as to maintain a constant number of active tokens and stable system performance.

Each token leads to an independent review chain. The vendor's review modification capability is proportional to the number of review chains. The more review chains, the less trustworthy the reviews from users' viewpoint. Thus, the vendor has the motivation to keep the token number as small as possible. On the other hand, there should be sufficient tokens to avoid token starvation problem, where some user never obtains a token to leave its review. In Section 7, we will study the impact of token number on the system performance.

A user, when having a review to submit, transmits a token request message. After receiving the request, a nearby user currently holding a token or the vendor (if having a spare token) may send the token to the requesting user. The requesting user accepts the first arrived valid token and replies with an ACK message. For other received tokens, it replies with a RETURN message, indicating that it no longer needs a token. The token request, ACK and RETURN messages are signed by senders using (pseudonym) secret keys, which are nonforgeable. Token forwarding happens toward one user at a time; successfully forwarded tokens (replied with ACKs) are no longer passed to any other user. Transmission retrials may be made up to a maximum number of times to tolerate communication failure.

The vendor maintains a token-pseudonym (TP) list. In this list, each token is linked to a pseudonym that belongs to a user who most recently submitted a review using the token. The list is updated whenever the vendor receives a new review, and is periodically broadcasted to all users in the vendor's transmission range. Once a token's information is published, the vendor cannot simply remove the token from the TP list because any modification to the list will cause inconsistency with previously published information and be noticed by the public. A user having a token will forward the token, after using it, to a randomly selected neighboring user who is wishing to submit a review. Below, we explain token structure and how a token is forwarded among users.

Consider three users u_1, u_2, and u_3, with u_1 neighboring u_2, and u_2 neighboring u_3. They are, respectively, members of groups g_1, g_2, g_3 and have obtained pseudonyms $pid_{1,1,*}, pid_{2,2,*}, pid_{3,3,*}$ from the corresponding group authorities. The vendor initializes a token with an identifier tok. It generates a public/private key pair (pk_t, sk_t) for tok and publishes the public key pk_t. Suppose that it intends to issue the token to u_1. Then, the token initially is a signature $\sigma_1 = Sign_{sk_t}(g_1 \| pid_{1,1,*} \| T)$, where T is current time stamp. We denote this initial version σ_1 by tok_1. It implies that u_1 is the first user who can submit a review and must submit the review using pseudonym $pid_{1,1,*}$. The pseudonym $pid_{1,1,*}$ is exposed to the vendor by u_i.

After submitting a review using tok_1 and $pid_{1,1,*}$, u_1 updates tok_1 to tok_2 and passes tok_2 to u_2 as a response to u_2's token request. The updated version tok_2 is $(PF_1, \sigma_2 = Sign_{psk_{1,1,*}}(g_2 \| pid_{2,2,*} \| T_1))$, where $PF_1 = (g_1, pid_{1,1,*}, \sigma_1)$ is the token forwarding proof of u_1. Note that, $(pk_t, tok, pid_{1,1,*})$ is currently included in the TP list. Suppose that tok_2 is the first token received by u_2. u_2 does the following: validate tok_2 by checking the authenticity of PF_1 using signatures σ_1 and σ_2, check if the user with $pid_{1,1,*}$ is the one that lastly forwards tok (by looking at the TP list), send an ACK to u_1, submit its review using tok_2 and $pid_{2,2,*}$, and update tok_2 to $tok_3 = (PF_1, PF_2, \sigma_3 = Sign_{psk_{2,2,*}}(g_3 \| pid_{3,3,*} \| T_2))$, where $PF_2 = (g_2, pid_{2,2,*}, \sigma_2)$, and send tok_3 to u_3.

The token forwarding process is repeated among users until tok expires or is brought out of the network. tok is always in the form of $(\{PF_x = (pid_{x,*,*}, \sigma_x)\}_{x \in \mathcal{X}}, \sigma_y)$ where u_x has forwarded the token and u_y the receiver user. It includes the hierarchical signatures that define the order of review submission and organizes submitted reviews in a chain structure. Note that malicious token drop is handled by the vendor through token replacement, as discussed previously.

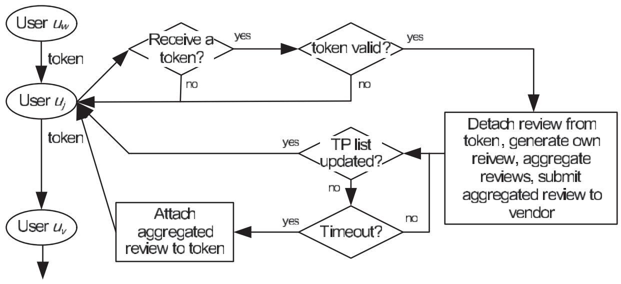

User u_w
token
User u_j
token
User u_v
Receive a token?
yes
no
token valid?
yes
no
Detach review from token, generate own reivew, aggregate reviews, submit aggregated review to vendor
TP list updated?
yes
no
Timeout?
yes
no
Attach aggregated review to token

Fig. 3. Review generation and submission.

Reducing token size by signature aggregation. We introduce an aggregate signature technique within multiple-authority settings, which is a variant of the scheme presented in [12]. This technique aggregates the signatures of different users from different social groups, and the signatures can be on different messages. By this technique, the signatures in a token can be aggregated, and the token size, thus the communication cost can be reduced. The aggregate signature technique will also be used for review aggregation in the next section, and the associated $Sign$ and $Verify$ functions will be instantiated as explained below.

Let $\mathbb{G}$ and $\mathbb{G}_T$ be two cyclic additive groups with the same order q, and $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ be a bilinear pairing [29]. P is a generator of $\mathbb{G}$. A group authority c_h picks a random $s_h \in \mathbb{Z}/q\mathbb{Z}$ and sets $Q_h = s_h P$. It also chooses two cryptographic hash functions $H_1 : \{0,1\}^* \rightarrow \mathbb{G}$, and $H_2 : \{0,1\}^* \rightarrow \mathbb{Z}/q\mathbb{Z}$.

Key generation. A user u_j if registering to a group authority c_{h_j} will receive a bunch of pseudonym secret keys corresponding to randomly generated pseudonyms $pid_{j,h_j,*}$. Within a social group, the pseudonyms are never repeatedly assigned to users. The pseudonym secret keys $psk_{j,h_j,*} = (k_{j,0}, k_{j,1})$, where $k_{j,0} = s_{h_j} P_{j,0} = s_{h_j} H_1(pid_{j,h_j,*} \| 0)$ and $k_{j,1} = s_{h_j} P_{j,1} = s_{h_j} H_1(pid_{j,h_j,*} \| 1)$.

Signing. u_j generates a string as $str = ``v"$, where v represents the identity of the vendor. Note that, all tokens are toward a specific vendor at a time period t. Therefore, the string can be obtained by other similar users. The signature on m_j will be $\sigma_j = Sign_{psk_{j,h_j,*}}(m_j) = (str, S_j, R_j)$

$$S_j = r_j P_s + k_{j,0} + \beta_j k_{j,1} \text{ and } R_j = r_j P, \tag{1}$$

where $P_s = H_1(str)$, $\beta_j = H_2(m_j, pid_{j,h_j,*}, str)$, and r_j is randomly chosen from $\mathbb{Z}/q\mathbb{Z}$.

Aggregation. Multiple signatures with the common str can be aggregated. Consider $\sigma_j = (str, S_j, R_j)$ for $1 \leq j \leq n$ are the signatures with common string str. The aggregated signature $\sigma_{agg} = (str, S_{agg}, R_{agg})$ can be obtained, where $S_{agg} = \Sigma_{j=1}^n S_j$ and $R_{agg} = \Sigma_{j=1}^n R_j$.

Verification. Consider $\sigma_{agg} = (str, S_{agg}, R_{agg})$ is the aggregated signature for $\{(str, S_j, R_j)_{1 \leq j \leq n}\}$. The function $Verify(pid_{1,h_1,*} \| \cdots \| pid_{n,h_n,*}, m_1 \| \cdots \| m_n, \sigma_{agg})$ outputs 1 if the following condition holds; 0 otherwise

$$\begin{aligned} e(S_{agg}, P) \overset{?}{=} e(R_{agg}, P_s) \cdot \Sigma_{j=1}^n e(H_1(pid_{j,h_j,*} \| 0) \\ + \beta_j H_1(pid_{j,h_j,*} \| 1), Q_{h_j}), \end{aligned} \tag{2}$$

where $\beta_j = H_2(m_j, pid_{j,h_j,*}, str)$. A user will only use $pid_{j,h_j,*}$ to generate a review on m_j for v only once to resist existential forgery attack [28].

4.3 Review Generation and Submission

Review generation and submission involve multiple steps as shown in Fig. 3. Review generation does not rely on tokens, which gives users flexibility to generate review. Consider a user u_j who just received a token tok from a nearby user u_w with pseudonym $pid_{w,*,*}$. It checks if the received tok is valid. This validation step has two perspectives: 1) to ensure that tok is indeed originated from the vendor and has been properly forwarded in the past; 2) to ensure that tok is sent by the user who lastly used it. The first goal can be realized by using the public key pk_t of the vendor and the forwarder information (including secrets, pseudonyms, and time stamps) embedded in tok. The second one can be achieved by checking if the association $(tok, pid_{w,*,*})$ exists in the latest TP list provided by the vendor.

During token forwarding, a token is supposed to be passed to only one user that is wishing to submit a review to the same vendor. When multiple such users are present, a random selection can be made. In case that the token is passed to multiple users, whether accidentally (due to the failure in transmitting ACK message) or intentionally, the vendor will only accept the first subsequently submitted review using the token. With the second check on the TP list during token validation, the other users holding the token will find that the token is no longer valid and then try to find a new token to submit their reviews.

After confirming that tok is valid, u_j separates the attached review REV_w from tok. It checks the authenticity of REV_w. It is able to do so because u_w's pseudonym $pid_{w,*,*}$ is included in tok. If REV_w is invalid, u_j will discard it. After the review authenticity check, u_j generates its own review rev_j. Denote the review content by α_j. Suppose that u_j will use the pseudonym $pid_{j,h,*}$ from social group g_h for the review generation, and set T_j to current time which is larger than all the time stamps embedded in tok. It computes

$$\begin{aligned} \sigma_j &= Sign_{psk_{j,h,*}}(\alpha_j \| v \| T_j) \\ rev_j &= \langle g_h, pid_{j,h,*}, \alpha_j, v, T_j, \sigma_j \rangle. \end{aligned} \tag{3}$$

The signature σ_j can be verified by checking $Verify(pid_{j,h,*}, \alpha_j \| v \| T_j, \sigma_j) \overset{?}{=} 1$ (see the previous section for the details of functions $Sign$ and $Verify$). The receiver then knows that rev_j is indeed generated by a user from g_h at time T_j, not forged by the vendor or a user from a different group. Having generated rev_j, u_j aggregates it with REV_w (by the signature aggregation technique in Section 4.2) and submits the aggregated reviews REV_j ($REV_j = rev_j$ if $REV_w = null$) together with tok to the vendor. The vendor checks the validity of REV_j and tok, and broadcast the updated TP list. Review aggregation is the same process as signature aggregation during token forwarding. Review aggregation has two advantages: 1) it effectively resists the review attacks; 2) it largely reduces the communication overhead.

Note that u_j is unable to forge a review of u_w because it cannot obtain any pseudonym secret key $psk_{w,*,*}$, and u_j is unable to replace the review with any other review received from u_w in the past because time stamp is used to prevent review replay. Direct replacement can be easily detected and rejected by the vendor. Further, u_j cannot forward the token without submitting REV_w and/or rev_j because the token records the forwarding history and the vendor will detect the review missing when it later receives the token as part of a review submission made by another user.

After submitting REV_j and tok to the vendor, u_j checks the updated TP list from the vendor. An unsuccessful submission can be due to communication failure or review rejection. To tolerate communication failure, a number of submission retrials can be made before drawing a submission failure conclusion. Upon receiving the updated TP list, u_j will check which pseudonym tok is related to in the list. If tok is related to $pid_{j,h,*}$, meaning that u_j have successfully submitted REV_j, u_j will forward tok to a nearby user as described in the previous section. If tok is still related to $pid_{w,*,*}$, meaning that u_j's submission failed, u_j will resort for cooperative submission by sending tok and REV_j together to a nearby user that is requesting for a token. If tok is related to a different pseudonym, implying that u_w must have sent the token to multiple users and u_j's submission failed, u_j will try to find a new token from nearby users and submit REV_j using it.

4.4 Sybil Attack Detection

We further extend the bTSE to a Sybil-resisted TSE, named SrTSE, which effectively prevents the sybil attacks.

Sybil Attacks. In Section 3.2, we define two types of sybil attacks: The sybil attack 1 is launched by a group of registered users. They aim at telling other users the bad service from a vendor while the service of the vendor is good. With the valid registration, these malicious users are able to leave false reviews toward a specific vendor. Even realizing the reviews are not in accord with the service, the vendor cannot simply delete or reject the reviews. If the vendor does, users will detect such behavior and regard the vendor as a dishonest service provider. Besides, the sybil attack 2 is launched by a vendor and a group of registered users. They aim at raising the reputation of the service from a vendor while the service of the vendor is not that good. The reviews generated by these malicious users cannot be distinguished from other reviews by well-behaving users. In the bTSE, every user receives multiple pseudonyms and the corresponding secret keys. For example, u_j has $pid_{j,h,1}, pid_{j,h,2}, \ldots$ in social group g_h. Since these pseudonyms are random numbers and cannot be linked by anyone except group authorities, u_j can use $pid_{j,h,1}, pid_{j,h,2}, \ldots$ to generate multiple reviews toward a vendor for a short time period. In addition, u_j can form the false reviews in chain structure or ring structure. Therefore, from the perspective of other users, they cannot tell if these reviews are from the same user or not.

SrTSE. In the SrTSE, we introduce a novel solution to prevent the two sybil attacks. In the S-MSN, we consider that a user has no need to generate multiple reviews toward a vendor in a short time period. The SrTSE allows a user to leave only one review toward a vendor for a predefined time slot. If a user generates multiple reviews with the same pseudonyms, the linkability of the reviews can be easily verified by the public; if a user generates multiple reviews with different pseudonyms toward a vendor in a time slot, its real identity will be exposed to the public. To achieve the above properties, we modify the pseudonym generation and the signature scheme of the bTSE.

Let $\mathbb{G}$ and $\mathbb{G}_T$ be two cyclic additive groups with the same order q, and $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ be a bilinear pairing [29]. P, Q are two generators of $\mathbb{G}$. A group authority c_h

picks a random $s_h \in \mathbb{Z}/q\mathbb{Z}$ and sets $Q_h = s_h P$. It also chooses two cryptographic hash functions $H_1 : \{0,1\}^* \rightarrow \mathbb{G}$, and $H_2 : \{0,1\}^* \rightarrow \mathbb{Z}/q\mathbb{Z}$.

Consider a user u_j registers to the social group g_h in the SrTSE. Then, u_j obtains the following values:

- $pid_{j,h_j,*}$, a published random number.
- $a_{j,*} = \rho H_2(pid_{j,h_j,*}) + id_j$, where id_j is the real identity of u_j, and ρ is a coefficient in $\mathbb{Z}/q\mathbb{Z}$.
- $b_{j,*} = (r_* P, s_h Q + r_* H_1(a_{j,*} r_* P \| pid_{j,h_j,*}))$, where r_* is a random number. This is a signature on $a_{j,*} r_* P$ by the group authority c_h.

For multiple random numbers $pid_{j,h_j,*}$, u_j obtains multiple tuples $(pid_{j,h_j,*}, a_{j,*}, b_{j,*})$ from c_h. Then, u_j regards $pid_{j,h_j,*}$ as the pseudonym and $psk_{j,h_j,*} = a_{j,*}$ as the secret key. u_j generates a signature on message m_j as follows:

- u_j calculates $a_{j,*} H_1(m_j)$.
- u_j generates a random number $\bar{r} \in \mathbb{Z}/q\mathbb{Z}$, and outputs a signature $\sigma_j = (pid_{j,h_j,*}, \sigma_{j,1}, \sigma_{j,2}, \sigma_{j,3}, \sigma_{j,4})$, where

$$\begin{aligned}\sigma_{j,1} = a_{j,*} H_1(m_j), \sigma_{j,2} = a_{j,*} r_* P, \sigma_{j,3} = r_* P, \sigma_{j,4}\\ = s_h Q + r_* H_1(a_{j,*} r_* P \| r_* P \| pid_{j,h_j,*}).\end{aligned}$$

If an entity receives σ_j, it checks if

$$e(\sigma_{j,4}, P) \stackrel{?}{=} e(Q, Q_h) e(H_1(\sigma_{j,2} \| \sigma_{j,3} \| pid_{j,h_j,*}), \sigma_{j,3})$$

and $e(\sigma_{j,1}, \sigma_{j,3}) \stackrel{?}{=} e(H_1(m_j), \sigma_{j,2})$. Note that, $(\sigma_{j,3}, \sigma_{j,4})$ is a signature generated by the group authority c_h because s_h is the secret key only known to c_h. From Step 1, the authenticity of $\sigma_{j,2}$ and $pid_{j,h_j,*}$ can be guaranteed. In addition, from Step 2, if the equality holds, it is publicly verified that u_j knows the value $a_{j,*}$. In fact, we build our signature scheme based on identity-based signature [29] and short signature [30].

Sybil attack detection. For each review, we require users to sign on $m_j = v \| t$ where v is the vendor's name and t is the time slot. If users do not output the signature on m_j, its review will not be accepted by the public. We consider a sybil attack launched by u_j who generate two reviews with two different pseudonyms $pid_{j,h_j,1}$ and $pid_{j,h_j,2}$. If both reviews are authentic, they must contain both $a_{j,1} H_1(m_j)$ and $a_{j,2} H_1(m_j)$ which can be accessed by the public. Thus, the public is able to calculate $Tr = id_j H_1(m_j)$ from

$$a_{j,1} = \rho H_2(pid_{j,h_j,1}) + id_j, \; a_{j,2} = \rho H_2(pid_{j,h_j,2}) + id_j, \quad (4)$$

because

$$id_j = \frac{a_{j,1} H_2(pid_{j,h_j,2}) - a_{j,2} H_2(pid_{j,h_j,1})}{H_2(pid_{j,h_j,2}) - H_2(pid_{j,h_j,1})}. \quad (5)$$

To recover the real identity of the sybil attacker, any entity calculates $Tr' = id H_1(m_j)$ for every possible id and tests if $Tr' \stackrel{?}{=} Tr$. The entity outputs the recovered identity id, upon satisfaction of the above equation.

Note that, similar to [10], [11], the vendors or the users can precalculate values $id H_1(m_j)$ for every possible identity, and then, they just need to check the equality between Tr and these values. Within a constant time, the real identity of the sybil attacker can be revealed.

TABLE 1
Security of the Proposed Systems

	$\mathcal{L}$	$\mathcal{R}$	$\mathcal{M}$	$\mathcal{S}$	S_t1	S_tk	S_r1	S_rk
NCP	Y	N	N	N	N/A	N/A	N/A	N/A
bTSE	Y	Y	Y	N	$2\lvert\mathbb{G}\rvert$	$2\lvert\mathbb{G}\rvert$	$2\lvert\mathbb{G}\rvert$	$2\lvert\mathbb{G}\rvert$
SrTSE	Y	Y	Y	Y	$2\lvert\mathbb{G}\rvert$	$2\lvert\mathbb{G}\rvert$	$4\lvert\mathbb{G}\rvert$	$(3k+1)\lvert\mathbb{G}\rvert$

Aggregate signature in the SrTSE. The signature aggregation plays an important role in the bTSE because it largely reduces the communication overhead. We will also explore the possible aggregation scheme for the newly developed signatures in the SrTSE. Observing the modified signature scheme, the pseudonyms and the corresponding secret keys have to be equipped with a trapdoor such that other entity (not group authority) is able to recover the real identity of the sybil attacker. Therefore, the aggregation on signatures becomes more difficult. From the verification Step 1 and Step 2, we can see that $\sigma_{j,1}$, $\sigma_{j,2}$, and $\sigma_{j,3}$ cannot be aggregated because $\sigma_{j,2}$ and $\sigma_{j,3}$ have to be individually input in the hash function and $\sigma_{j,1}$ is paired with different $\sigma_{j,3}$ every time. But $\sigma_{j,4}$ from different users can be aggregated in the form of $\prod_j \sigma_{j,4}$ because it is always paired with P. The verification on the aggregate signature is changed to

$$e\left(\prod_j \sigma_{j,4}, P\right) = e(Q, Q_h) \prod_j e(H_1(\sigma_{j,2} \| pid_{j,h_j,*}), \sigma_{j,3}).$$

4.5 Summary of bTSE and SrTSE

We have proposed two trustworthy service evaluation systems: One considers the review attacks only and the other one considers both the review attacks and the sybil attacks. In the following, we summarize the efficiency and security properties of these two systems. We also consider the NCP system, where pseudonyms are employed and the reviews are individually submitted by users. Let "$\mathcal{L}$, $\mathcal{R}$, $\mathcal{M}$, $\mathcal{S}$, S_t1, S_r1, S_tk, and S_rk" denote "review linkability attacks, review rejection attacks, review modification attacks, sybil attacks, the size of signature on one token, the size of signature on one review, the size of k-aggregated signatures on tokens, the size of k-aggregated signatures on reviews, respectively. Let "Y, N" denote "resist, not resist," respectively.

From the above the security comparisons in Table 1, it can be seen that both the bTSE and the SrTSE outperforms the NCP system in terms of security. Moreover, the bTSE resists "$\mathcal{L}$," "$\mathcal{R}$," and "$\mathcal{M}$," while the SrTSE additionally resists "$\mathcal{S}$."

We also give the analysis of communication overhead in the above Table 1. We do not count the sizes of messages and the common strings because their sizes are negligible compared to the signatures. From the Table 1, both bTSE and SrTSE have very efficient review and token generation due to the signature aggregation. To resist the sybil attacks, SrTSE employs a trapdoor in the pseudonym, which leads to a linearly increasing size in review generation of the SrTSE.

5 Security Analysis

In this section, we focus on the sybil attacks. The security analysis of review attacks can be found in [2]. To resist the sybil attacks, we need to prove that the SrTSE satisfies the following two properties:

- *P1.* If a user leaves two or more false reviews with different pseudonyms toward a vendor in a time slot, its real identity can be derived by the vendor and other users.
- *P2.* If a user leaves only one review toward a vendor in a time slot, its real identity can be protected.

We first consider the property *P1* of the SrTSE. We consider a malicious user u_j generates two false reviews that include two signatures on m_j. The two pseudonyms are different. From the signature, $\sigma_{j,1}$ can be obtained. If both signatures are valid, the relations of $\sigma_{j,1}$, $\sigma_{j,2}$, and $\sigma_{j,3}$ can be verified. Since $\sigma_{j,2}$ and $\sigma_{j,3}$ are included in the message of $\sigma_{j,4}$, their authenticity can also be verified. From the two reviews, anyone can obtain $a_{j,1}H_1(m_j)$ and $a_{j,2}H_1(m_j)$, and derive

$$id_jH_1(m_j) = \frac{H_2(pid_{j,h_j,2})a_{j,1} - H_2(pid_{j,h_j,1})a_{j,2}}{H_2(pid_{j,h_j,2}) - H_2(pid_{j,h_j,1})} \cdot H_1(m_j). \quad (6)$$

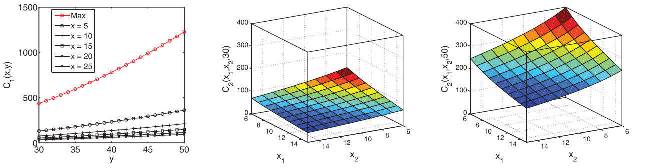
By executing the equality checks, the real identity id_j of u_j will be determined. Note that, ρ is determined by the group authorities. Different groups generate different ρ. We consider the used two pseudonyms $pid_{j,h_j,1}$ and $pid_{j,h_j,2}$ are from the same social group. We can further require a trusted third authority to coordinate all the group authorities to generate the same ρ for one user, and then the sybil attacks using two pseudonyms from different groups can be resisted.

We then consider the property *P2* of the SrTSE. From a signature σ_j, the real identity can be disclosed from $a_{j,*}$ which is contained in $\sigma_{j,1} = a_{j,*}H_1(m_j)$ and $\sigma_{j,2} = a_{j,*}r_*P$. Denote $H_1(m_j) = r'P$. Thus, we have two values $\sigma_{j,1} = (\rho H_2(pid_{j,h_j,*}) + id)r'P$ and $\sigma_{j,2} = (\rho H_2(pid_{j,h_j,*}) + id)r_*P$. If multiple signatures with different pseudonyms are generated toward different m_j by u_j, we can obtain:

$$\begin{aligned}(\rho H_2(pid_{j,h_j,1}) + id_j)r'_1P,&(\rho H_2(pid_{j,h_j,1})\\ + id_j)r_1P,&(\rho H_2(pid_{j,h_j,2})\\ + id_j)r'_2P,&(\rho H_2(pid_{j,h_j,2})\\ + id_j)r_2P,&\cdots\end{aligned}$$

Since $(r'_1, r_1, r'_2, r_2, \ldots)$ are independent and unknown to the public. The random number ρ cannot be removed by the linear combination of these values. Therefore, the real identity id_j is always anonymized by ρ, and thus id_j is protected.

6 Numerical Results



(a) 1 attacker (b) 2 attackers (y = 30) (c) 2 attackers (y = 50)

Fig. 4. Efforts on detecting the sybil attack.

The SrTSE can resist the sybil attack, i.e., the sybil attack can be detected without the involvement of the group authorities. In the following, we study the performance of the SrTSE under the sybil attack. We will evaluate how much computation costs needed to detect the false reviews by the sybil attack. We first consider the case of a single malicious user in the SrTSE. The sybil attacker generates x false reviews toward the vendor in time slot t using its x different pseudonyms. The vendor totally receives y reviews in time slot t $(y \geq x)$. From (6), the vendor needs to do every calculation for any pair of the received reviews. That means, the maximum number of calculation is $\binom{y}{2}$. We denote the number of calculations needed to filter all the false reviews by $C_1(x,y)(\leq \binom{y}{2})$. In fact, if two reviews have been identified to be associated with the attacker, all the rest false reviews can be easily identified by comparing them with the detected false reviews. Thus, the expected value of $C_1(x,y)$ is calculated by

$$\begin{aligned}C_1(x,y) &= \frac{y-x}{y}(y - 1 + C_1(x, y-1)) + \frac{x}{y}(y-1)\\ &= y - 1 + \frac{y-x}{y}C_1(x, y-1).\end{aligned} \quad (7)$$

In the above equation, $\frac{y-x}{y}$ and $\frac{x}{y}$ represent the probabilities of choosing a valid review and a false review, respectively. If a valid review is chosen, we need to do $y-1$ calculations between the chosen review with the rest $y-1$ reviews and $C_1(x,y-1)$ calculations among $y-1$ reviews. If a false review is chosen, we need to do the first $y-1$ calculations and then all the false reviews will be detected. Similarly, we further derive the number of calculations $C_2(x_1,x_2,y)$ in case of two malicious users, as shown in (8), where x_1 and x_2 represent the numbers of false reviews of two malicious users, respectively. We have $x_1 + x_2 \leq y$.

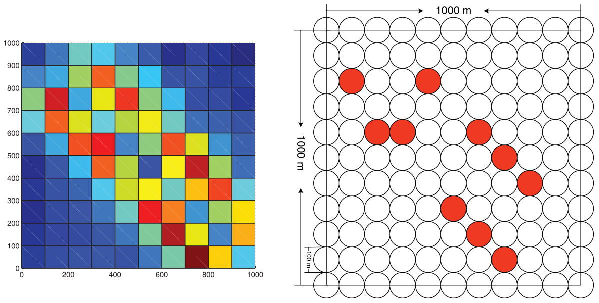
For the initial values, we have $C_1(x,x) = x - 1$, $C_2(0, x_2, y - x_1) = C_1(x_2, y - x_1)$ and

$$C_2(x_1, 0, y - x_2) = C_1(x_1, y - x_2).$$

If $x_1 + x_2 = y$, $C_2(x_1,x_2,y) = y - 2 + \frac{2x_1x_2}{y}$

$$\begin{aligned}C_2(x_1,x_2,y) &= y - 1 + \frac{y - x_1 - x_2}{y}C_1(x_1,x_2,y-1)\\ &+ \frac{x_1}{y}C_2(0,x_2,y-x_1) + \frac{x_2}{y}C_2(x_1,0,y-x_2).\end{aligned} \quad (8)$$

Then, we plot $C_1(x,y)$ and $C_2(x_1,x_2,y)$ and $C_2(x,y)$ in terms of x, y, x_1, and y_1, respectively, as shown in Fig. 4. From Fig. 4a, in case of one malicious user, it can be seen that the number of calculations almost increases linearly as the number of received reviews increases. When more reviews received at the vendor, more calculation efforts are needed to find the false reviews. Moreover, when the number of false reviews increases, the calculation efforts can be reduced because the probability of finding a false review is larger. From Figs. 4b and 4c, we can observe that when the number of false reviews decreases or the number of received reviews increases, the number of calculations to detect all false reviews increases. Note that when $x_1 = x_2 = 15$ and $y = 30$, the number of calculations is 43. In this case, 30 reviews are all false reviews, and 43 calculations are needed on average to detect them. The reason is that the



(a) Population density (b) The selected hotspots

Fig. 5. Candidate positions for placing the vendor.

calculations cannot detect any false reviews when the two reviews are separately from two users.

7 Performance Evaluation

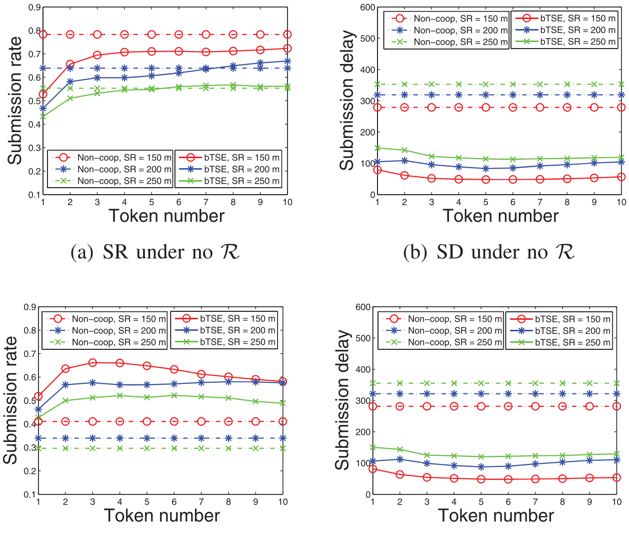
In this section, we evaluate the performance of the bTSE through trace-based custom simulations. We choose to compare the bTSE with a NCP system, where each user directly submits its review to the vendor without any synchronization constraint (use of tokens). We use the following three performance metrics:

- *SR*. It is defined as the ratio of the number of successfully submitted reviews to the total number of generated reviews in the network.
- *SD*. It is defined as the average duration between the time when a review is generated and the time when it is successfully received by the vendor.

7.1 Simulation Setup

We use the real trace log [31] obtained from pedestrian runners to generate user mobility. According to the log, 100 users are randomly scattered in a $1{,}000 \times 1{,}000\ \mathrm{m}^2$ square region and move at random velocities with a mean value of $1\ \mathrm{m/s}$. The log records user location changes in successive 900 time slots. We divide the region into a 10×10 grid, where each cell is a square of side length $100\ \mathrm{m}$. We create a circle of radius $50\ \mathrm{m}$ around each grid point. There are 121 circles. The areas enclosed by these circles are called spots and shown in Fig. 5b. No two spots overlap. We analyze the trace log and found 10 hotspots, as follows: Let $d_{m,n}$ denote the number of users in spot a_m at time n, where integers $m \in [1, 121]$ and $n \in [1, 900]$. We sort the spots in an descending order by $d_m = \sum_{n=1}^{900} d_{m,n}$, and choose the top ten spots as hotspots. Fig. 5 shows the population density d_m of the spots and highlights the selected hotspots, which are the candidate places to host the vendor.

We define a universal attribute set of 50 elements. The set is known by all users. Users are organized into 10 social groups, each being tagged with five random attributes. Each user has a membership with 1-5 random social groups, that is, it may have 5-25 attributes, inherited from the belonged social groups. The vendor (precisely, its service) has three random attributes. If a user shares a common attribute with the vendor, it will be interested in the vendor (service). For simplicity, we do not implement users random state transition from “not interested” to “interested” caused by the recommendation from its friends. Each user has a transmission range of $80\ \mathrm{m}$. The vendor has a transmission range equal to its SR. A user interested in the vendor wishes to submit a review to the vendor when it enters the vendor’s service range for the first time. Direct review submission is possible only when the vendor is within the user’s transmission range. As the trace log covers a small region and a small period time, we do not implement the token timeout interval θ_{exp} (see Section 4.2).



(a) SR under no $\mathcal{R}$ (b) SD under no $\mathcal{R}$

(c) SR under $\mathcal{R}$ (d) SD under $\mathcal{R}$

Fig. 6. Performance evaluation of TSE.

We conduct two sets of simulations under the situations with/without the review rejection attacks ($\mathcal{R}$). We vary SR between 150 and $300\ \mathrm{m}$, and token number TN between 1 and 10. As analyzed in Section 5, the bTSE resists the review linkability and modification attacks through cryptography techniques and specially designed review structure, and mitigates review rejection attack through cooperative review submission. The first two attacks have no influence on review submission. In our simulation study, we are, therefore, interested only in the impact of review rejection attack on the system performance. Each review is a value ranged in $[0, 1]$. A review is negative if its value is lower than 0.5. The vendor performs review rejection action by rejecting all negative reviews. When multiple reviews are aggregated and submitted together, the vendor accepts them all if their average value is no less than 0.5, or rejects them all otherwise. We place the vendor at the centers of the 10 hotspots in turn and conduct 50 simulation runs for each placement. Using the total 500 simulation runs, we obtain the average results to be analyzed in the next section.

7.2 Simulation Results

7.2.1 Under No Review Rejection Attack

We first study the system performance in relation with SR from Figs. 6a and 6b. When SR goes up, the number of users who enter the service range and, thus, generate reviews increases. Recall that each user has a transmission range much smaller than SR. In the NCP system, users have to move close enough to the vendor to submit their reviews. Hence, the system shows a decreasing SR and increasing SD with SR. In the bTSE, review submission is constrained by token possession in addition to user-to-vendor distance on one hand. On the other hand, cooperative review submission is triggered when direct submission is not possible. The interplay of the two factors renders the bTSE exhibiting a performance trend similar to the NCP system's in SR and SD as SR varies. From Fig. 6b, the bTSE has lower SD than the NCP system, up to 75 percent lower.

We then look at how TN impacts the system performance. Intuitively, when TN goes up, users have increased opportunity to submit reviews, leading to raised system performance. This intuition is confirmed by the results in Figs. 6a and 6b. We observe an arguable phenomenon: SR and delay both stabilize after TN is beyond certain value. In the case of $SR = 150$, it occurs after $TN = 20$ and is, however, not shown here. The reason for this phenomenon is as follows: When there are more tokens circulating in the network, initially users can easily get tokens and submit their reviews. Recall that users no longer participate in the review system once their reviews are submitted to the vendor or forwarded to others. Over time, the network of participating users becomes sparse, and these users have less chance to receive a token due to decreased network density.

7.2.2 Under Review Rejection Attack

Figs. 6c and 6d show the performance comparison of the bTSE and the NCP system when the vendor launches the review rejection attack. We observe that the NCP system has a performance drop (>25 percent) in SR. Indeed, it is not equipped with any security mechanism against the attack and suffers performance degradation. SD does not shown any noticeable change because only direct submission is engaged in the NCP system and only successfully submitted reviews are considered during delay calculation. Compared with the case of no review rejection attack, the bTSE only has slightly reduced (<10 percent smaller) SR and nearly unchanged SD thanks to the user cooperation and review aggregation mechanisms. The bTSE achieves significantly higher SR than the NCP system, up to 100 percent. These simulation results indicate that the bTSE can effectively resist the review rejection attack.

8 Conclusions

In this paper, we have proposed a TSE system for S-MSNs. The system engages hierarchical signature and aggregate signature techniques to transform independent reviews into structured review chains. This transformation involves distributed user cooperation, which improves review integrity and significantly reduces vendors' modification capability. We have presented three review attacks and shown that the bTSE can effectively resist the review attacks without relying on a third trusted authority. We have also considered the notorious sybil attacks and demonstrated that such attacks cause huge damage to the bTSE. We have subsequently modified the construction of pseudonyms and the corresponding secret keys in the bTSE, and obtained a SrTSE system. The SrTSE allows users to leave only one review toward a vendor in a predefined time slot. If multiple reviews with different pseudonyms from one user are generated, the real identity will be disclosed to the public. Security analysis and numerical results show the effectiveness of the SrTSE to resist the sybil attacks. Further trace-based simulation study demonstrates that the bTSE can achieve high SRs and low SDs.

References

[1] W. Dong, V. Dave, L. Qiu, and Y. Zhang, "Secure Friend Discovery in Mobile Social Networks," *Proc. IEEE INFOCOM,* pp. 1647-1655, 2011.

[2] X. Liang, X. Li, R. Lu, X. Lin, and X. Shen, "Seer: A Secure and Efficient Service Review System for Service-Oriented Mobile Social Networks," *Proc. IEEE 32nd Int'l Conf. Distributed Computing Systems (ICDCS),* pp. 647-656, 2012.

[3] X. Liang, X. Li, T. Luan, R. Lu, X. Lin, and X. Shen, "Morality-Driven Data Forwarding with Privacy Preservation in Mobile Social Networks," *IEEE Trans. Vehicular Technology,* vol. 61, no. 7, pp. 3209-3222, Sept. 2012.

[4] T.H. Luan, L.X. Cai, J. Chen, X. Shen, and F. Bai, "VTube: Towards the Media Rich City Life with Autonomous Vehicular Content Distribution," *Proc. IEEE CS Eighth Ann. Conf. Sensor, Mesh and Ad Hoc Comm. Networks (SECON),* pp. 359-367, 2011.

[5] J.R. Douceur, "The Sybil Attack," *Proc. Revised Papers First Int'l Workshop Peer-to-Peer Systems (IPTPS),* pp. 251-260, 2002.

[6] J. Newsome, E. Shi, D.X. Song, and A. Perrig, "The Sybil Attack in Sensor Networks: Analysis & Defenses," *Proc. Third Int'l Symp. Information Processing in Sensor Networks (IPSN),* pp. 259-268, 2004.

[7] D. Quercia and S. Hailes, "Sybil Attacks Against Mobile Users: Friends and Foes to the Rescue," *Proc. IEEE INFOCOM,* pp. 336-340, 2010.

[8] R. Lu, X. Lin, X. Liang, and X. Shen, "A Dynamic Privacy-Preserving Key Management Scheme for Location-Based Services in VANETs," *IEEE Trans. Intelligent Transportation Systems,* vol. 13, no. 1, pp. 127-139, Mar. 2012.

[9] R. Lu, X. Lin, T. Luan, X. Liang, and X. Shen, "Pseudonym Changing at Social Spots: An Effective Strategy for Location Privacy in VANETs," *IEEE Trans. Vehicular Technology,* vol. 61, no. 1, pp. 86-96, Jan. 2012.

[10] X. Boyen and B. Waters, "Full-Domain Subgroup Hiding and Constant-Size Group Signatures," *Proc. 10th Int'l Conf. Practice and Theory Public Key Cryptography,* pp. 1-15, 2007.

[11] X. Liang, Z. Cao, J. Shao, and H. Lin, "Short Group Signature without Random Oracles," *Proc. Ninth Int'l Conf. Information and Comm. Security (ICICS),* pp. 69-82, 2007.

[12] C. Gentry and Z. Ramzan, "Identity-Based Aggregate Signatures," *Proc. Int'l Conf. Public Key Cryptography,* pp. 257-273, 2006.

[13] Y. Zhang, Z. Wu, and W. Trappe, "Adaptive Location-Oriented Content Delivery in Delay-Sensitive Pervasive Applications," *IEEE Trans. Mobile Computing,* vol. 10, no. 3, pp. 362-376, Mar. 2011.

[14] H. Tsai, T. Chen, and C. Chu, "Service Discovery in Mobile Ad Hoc Networks Based on Grid," *IEEE Trans. Vehicular Technology,* vol. 58, no. 3, pp. 1528-1545, Mar. 2009.

[15] Z. Zhu and G. Cao, "Toward Privacy Preserving and Collusion Resistance in a Location Proof Updating System," *IEEE Trans. Mobile Computing,* vol. 12, no. 1, pp. 51-64, Jan. 2013.

[16] I. Wen, "Factors Affecting the Online Travel Buying Decision: A Review," *Int'l J. Contemporary Hospitality Management,* vol. 21, no. 6, pp. 752-765, 2009.

[17] H. Rajan and M. Hosamani, "Tisa: Toward Trustworthy Services in a Service-Oriented Architecture," *IEEE Trans. Services Computing,* vol. 1, no. 4, pp. 201-213, Oct.-Dec. 2008.

[18] Y. Wang and L. Li, "Two-Dimensional Trust Rating Aggregations in Service-Oriented Applications," *IEEE Trans. Service Computing,* vol. 4, no. 4, pp. 257-271, Oct.-Dec. 2011.

[19] E. Ayday and F. Fekri, "Iterative Trust and Reputation Management Using Belief Propagation," *IEEE Trans. Dependable and Secure Computing,* vol. 9, no. 3, pp. 375-386, May/June 2012.

[20] A. Das and M.M. Islam, "SecuredTrust: A Dynamic Trust Computation Model for Secured Communication in Multiagent Systems," *IEEE Trans. Dependable and Secure Computing,* vol. 9, no. 2, pp. 261-274, Mar./Apr. 2012.

[21] J. Douceur, "The Sybil Attack," *Proc. Revised Papers First Int'l Workshop Peer-to-Peer Systems,* pp. 251-260, 2002.

[22] C. Karlof and D. Wagner, "Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures," *Ad Hoc Networks,* vol. 1, no. 2/3, pp. 293-315, 2003.

[23] B. Viswanath, A. Post, P.K. Gummadi, and A. Mislove, "An Analysis of Social Network-Based Sybil Defenses," *Proc. ACM SIGCOMM,* pp. 363-374, 2010.

[24] A. Mohaisen, N. Hopper, and Y. Kim, "Keep Your Friends Close: Incorporating Trust into Social Network-Based Sybil Defenses," *Proc. IEEE INFOCOM,* pp. 1943-1951, 2011.

[25] W. Wei, F. Xu, C.C. Tan, and Q. Li, "Sybildefender: Defend Against Sybil Attacks in Large Social Networks," *Proc. IEEE INFOCOM*, pp. 1951-1959, 2012.

[26] "Social Group,"Wikipedia, http://en.wikipedia.org/wiki/Social_group, 2013.

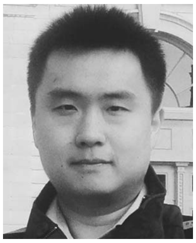
[27] D. Boneh and M.K. Franklin, "Identity-Based Encryption from the Weil Pairing," *Proc. 21st Ann. Int'l Cryptology Conf. Advances in Cryptology (CRYPTO)*, pp. 213-229, 2001.

[28] D. Pointcheval and J. Stern, "Security Proofs for Signature Schemes," *Proc. Int'l Conf. Theory and Application of Cryptographic Techniques (EUROCRYPT)*, pp. 387-398, 1996.

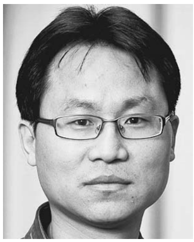
[29] B. Waters, "Efficient Identity-Based Encryption without Random Oracles," *Proc. Int'l Conf. Theory and Application of Cryptographic Techniques (EUROCRYPT)*, pp. 114-127, 2005.

[30] D. Boneh and X. Boyen, "Short Signatures without Random Oracles," *Proc. Int'l Conf. Theory and Application of Cryptographic Techniques (EUROCRYPT)*, pp. 56-73, 2004.

[31] X. Li, N. Mitton, and D. Simplot-Ryl, "Mobility Prediction Based Neighborhood Discovery for Mobile Ad Hoc Networks," *Proc. IFIP Int'l Conf. Networking (NETWORKING)*, pp. 138-151, 2011.



Xiaohui Liang (S'10) received the BSc degree in computer science and engineering, the MSc degree in computer software and theory from Shanghai Jiao Tong University (SJTU), China, in 2006 and 2009, respectively, and is currently working toward the PhD degree in the Department of Electrical and Computer Engineering, University of Waterloo, Canada. His research interests include applied cryptography, and security and privacy issues for e-healthcare system, cloud computing, mobile social networks, and smart grid. He is a student member of the IEEE.



Xiaodong Lin (S'07-M'09) received the PhD degree in information engineering from Beijing University of Posts and Telecommunications, Beijing, China, in 1998, and the PhD degree (with Outstanding Achievement in Graduate Studies Award) in electrical and computer engineering from the University of Waterloo, Waterloo, ON, Canada, in 2008. He is currently an assistant professor of information security with the Faculty of Business and Information Technology, University of Ontario Institute of Technology, Oshawa, ON, Canada. His research interests include wireless network security, applied cryptography, computer forensics, software security, and wireless networking and mobile computing. He received a Natural Sciences and Engineering Research Council of Canada (NSERC) Canada Graduate Scholarships (CGS) Doctoral and the Best Paper Awards of the 18th International Conference on Computer Communications and Networks (ICCCN 2009), the 5th International Conference on Body Area Networks (BodyNets 2010), and IEEE International Conference on Communications (ICC 2007). He is a member of the IEEE.

Xuemin (Sherman) Shen (M'97-SM'02-F'09) received the BSc degree in 1982 from Dalian Maritime University, China, and the MSc and PhD degrees in 1987 and 1990 from Rutgers University, New Jersey, all in electrical engineering. He is a professor and the University Research chair, the Department of Electrical and Computer Engineering, University of Waterloo, Canada. He was the associate chair for Graduate Studies from 2004 to 2008. His research focuses on resource management in interconnected wireless/wired networks, wireless network security, wireless body area networks, vehicular ad hoc and sensor networks. He is a coauthor/editor of six books, and has published more than 600 papers and book chapters in wireless communications and networks, control and filtering. He served as the Technical Program Committee chair for IEEE VTC'10 Fall, the Symposia chair for IEEE ICC'10, the Tutorial chair for IEEE VTC'11 Spring and IEEE ICC'08, the Technical Program Committee chair for IEEE Globecom'07, the General co-chair for Chinacom'07 and QShine'06, the chair for IEEE Communications Society Technical Committee on Wireless Communications, and P2P Communications and Networking. He also serves/served as the editor-in-chief for *IEEE Network*, *Peer-to-Peer Networking and Application*, and *IET Communications*; a founding area editor for *IEEE Transactions on Wireless Communications*; an associate editor for *IEEE Transactions on Vehicular Technology*, *Computer Networks*, and *ACM/Wireless Networks*, and so on; and the guest editor for *IEEE JSAC*, *IEEE Wireless Communications*, *IEEE Communications Magazine*, and ACM Mobile Networks and Applications, and so on. He received the Excellent Graduate Supervision Award in 2006, and the Outstanding Performance Award in 2004, 2007, and 2010 from the University of Waterloo, the Premier's Research Excellence Award (PREA) in 2003 from the Province of Ontario, Canada, and the Distinguished Performance Award in 2002 and 2007 from the Faculty of Engineering, University of Waterloo. He is a registered professional engineer of Ontario, Canada, an Engineering Institute of Canada Fellow, a Canadian Academy of Engineering Fellow, and a distinguished lecturer of IEEE Vehicular Technology Society and Communications Society. He has been a guest professor of Tsinghua University, Shanghai Jiao Tong University, Zhejiang University, Beijing Jiao Tong University, Northeast University, and so on. He is a fellow of the IEEE.