2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology

Improving PGP Web of Trust through the Expansion of Trusted Neighborhood

Guibing Guo, Jie Zhang, and Julita Vassileva*
School of Computer Engineering, Nanyang Technological University, Singapore
*Department of Computer Science, University of Saskatchewan, Canada
{gguo1@e.ntu.edu.sg}

***Abstract*—PGP Web of Trust where users can sign digital signatures on public key certificates of other users has been successfully applied in securing emails and files transmitted over the Internet. However, its rigorous restrictions on utilizable trust relationships and acceptable signatures limit its performance. In this paper, we first make some modification and extension to PGP Web of Trust by relaxing those constraints. In addition, we propose a novel method to further expand trusted neighborhood of users by merging the signatures of the trusted neighbors and finding the similar users based on the merged signature set. Confirmed by the experiments carried out in different simulated real-life scenarios, our method applied to both the modified and extended PGP methods can improve their performance. With the expansion of trusted neighborhood, the performance of the original PGP Web of Trust is also improved considerably.**

I. Introduction

Pretty Good Privacy (PGP) is widely accepted as the first successful attempt to make cryptography freely available to the public [1]. It does not rely on a trusted authority to cryptographically create a trusted digital certificate to specify the real owner of a public key. Instead, PGP allows the user who has a private key to create a digital certificate for the corresponding public key. To address the issue where the user may specify an arbitrary (unreal) owner for the public key in the certificate, PGP allows other users to digitally sign certificates that they believe to be *authentic*, i.e. the specified owner in the certificate is indeed the real owner of the public key. A user can verify a public key by checking whether there are digital signatures signed by other users whom she trusts. This solution is referred to as PGP Web of Trust.

However, one limitation about PGP Web of Trust is that it imposes restrictions on the signature feedbacks of certificates. More specifically, when a user encounters a certificate, if she chooses to sign the certificate, she can only sign it with a positive feedback stating that she believes the certificate to be authentic; therefore no one is able to directly state the opposite opinion. Another limitation is that PGP makes use of only direct trust relationships between users. To be more specific, only when a digital signature is signed by other users whom the user directly trusts, the user will believe the certificate to be authentic. In other words, PGP does not consider the transitivity property of trust. The performance of PGP is thus limited by the two restrictions. Our work is aimed at improving the performance of PGP by eliminating these restrictions and exploiting potential indirect trust relationships of users.

In this paper, we first modify PGP Web of Trust to accept negative feedbacks in signatures from users indicating that a certificate is believed to be inauthentic. Accordingly, the way of verifying the authenticity of a certificate based on provided feedbacks is also modified. We then extend PGP Web of Trust by also considering indirect trust relationships of users. Indirect trust is computed using the concept of trust propagation [2]. In this way, the pool of other users trusted by a user (called trusted neighborhood) may get expanded.

More importantly, we propose a novel method to further expand trusted neighborhood. It first merges the feedbacks on certificates provided by trusted neighbors of an active user, which may include both directly trusted neighbors specified by the user (including herself because the user should trust herself) and indirectly trusted ones identified by trust propagation used in the extended PGP Web of Trust. By relying on the majority opinion and ensuring the high consistency among the feedbacks, the merged feedback set can then well represent the opinions of this active user. Based on the merged feedback set, our method then finds other similar users of the active user who are not in the original trusted neighborhood. In this way, the trusted neighborhood is further expanded. Thus, the essence of our approach is to enlarge the original trusted neighborhood such that much more reliable information is available during the course of verifying the target certificates.

To evaluate the performance (both coverage and accuracy) of our proposed method and different versions of PGP, we simulate an environment where a large number of users create certificates for public keys and sign the certificates with digital signatures. Different scenarios are simulated to represent potential real-life situations. Experimental results confirm that both the modified and extended PGP can well increase the coverage, but the accuracy is often decreased. Our method, on another hand, can further largely improve both the coverage and accuracy of the modified and extended PGP, in most of the simulated scenarios. Compared with the original PGP, applying our method of expanding trusted neighborhood can not only increase the coverage but also the accuracy. Our method thus represents an important improvement on PGP Web of Trust, which can potentially widen its usage.

II. Related Work

The fundamental question that we attempt to answer in this work is how to expand trusted neighborhood of a user by exploring indirect trust relationships between this user and

978-0-7695-4513-4/11 $26.00 © 2011 IEEE
DOI 10.1109/WI-IAT.2011.134

others in an accurate manner. By doing so, the performance of PGP Web of Trust can be much improved in terms of both coverage and accuracy. Different attempts have been proposed to explore indirect trust relationship. A common way to compute such indirect trust is trust propagation [2], where trust in the stranger is calculated iteratively according to the trust a user places in a third user and the trust the third user places in the stranger. Trust is then computed along each chain and aggregated for all paths. Guha et al. [3] categorize trust propagation into four basic types of propagations: direct propagation, co-citation, transpose trust and trust coupling. Direct propagation is the same as the trust propagation discussed earlier. In transpose trust, if A trusts B then C trusting B implies that C also trusts A. Transpose trust is interesting as it connects two users who do not directly trust each other, on the basis of a common user. However, the authors do not specify the extent to which this kind of propagation should be taken into account and how much trust C should place in A.

Huang and Nicol [4] have applied the direct trust propagation method on PGP Web of Trust. In our work, we also apply the direct propagation method to extend the modified PGP Web of Trust. However, our experiments show that this extension can only well improve the coverage of PGP Web of Trust, but often not the accuracy. On another hand, our proposed method for further expanding trusted neighborhood can improve both the coverage and accuracy of the extended PGP Web of Trust. Our method is in fact similar to both the direct propagation and the transpose trust to some extent. To illustrate, suppose a user A has a trusted neighbor B. Our method finds another user C who is similar to B. The assumption here is that if two users are similar, they should trust each other. Thus, the similar users B and C trust each other. From the direction where B trusts C, our method is similar to the direct propagation, but from the opposite direction, our method is similar to the transpose trust. The difference is that user B in our method is a virtual user created based on the trusted neighbors of A in a way that A can (almost) fully trust B. Then, the similarity between B and C can be equivalent to the trust that A can place in C.

Our method of expanding trusted neighborhood is somewhat similar in spirit to classic collaborative filtering [5] widely used in the area of recommender systems. Collaborative filtering first finds a set of similar users to an active user. It then provides recommendations of items based on the similar users' feedback about the items. We expand trusted neighborhood by making use of this concept that similar users can be trusted to provide accurate prediction about whether a target certificate is authentic. In fact, some studies, for example [6] have already proposed the idea of dynamically updating the trust values based on how similar the feedback received from other users is to the user's own experience. The similarity of feedbacks provided by two users has also been successfully applied to compute the trust between the two users [7].

III. Modeling Original PGP Web of Trust

In this section, we summarize and mathematically model the original PGP Web of Trust (refer to [1] for more details). All public key certificates created by every user form the target space $\mathcal{T}$. Every public key certificate is a target $t \in \mathcal{T}$. Without loss of generality, there is a universal truth for every target t, i.e. whether or not it is authentic.

A user who digitally signs a public key certificate is a feedback provider p. All feedback providers form the feedback provider space $\mathcal{P}$. A user who verifies a public key certificate is a relying entity e. All relying entities form the relying entity space $\mathcal{E}$. Note that a user can be both a feedback provider and a relying entity. A digital signature signed on a public key certificate corresponds to a piece of feedback $f = (p, t, s)$. Note that the score s is not explicitly stated but implicitly implied as "I believe the public key certificate is authentic". This opinion is denoted by $s = 1$. Thus the score space $\mathcal{S} = \{1\}$. All digital signatures signed by a feedback provider p correspond to $\mathcal{F}_p$. All signatures accessible to a relying entity e form the view of this relying entity, i.e. $\mathcal{V}_e$.

A relying entity e also directly assigns every feedback provider p with a trust value as follows:

$$B_e(p) = \begin{cases} 1 & \text{if } p \text{ is completely trusted;} \\ 1/2 & \text{if } p \text{ is marginally trusted;} \\ 0 & \text{if } p \text{ is untrusted or unknown.} \end{cases} \quad (1)$$

Then, a group of trusted neighbors is selected as:

$$\mathcal{TN}_e = \{p \in \mathcal{P} : B_e(p) \geq 1/2\}. \quad (2)$$

A user (acting as a relying entity e) verifying a public key certificate (a target t_0) corresponds to the generation of a piece of advice concerning the target t_0 for the relying entity e. This process by default[1] is defined as:

$$r_e(t_0) = \begin{cases} 1 & \text{if } cc \geq 1 \text{ or } mc \geq 2; \\ 1/2 & \text{if } cc = 0 \text{ and } 0 < mc < 2; \\ 0 & \text{if } cc = 0 \text{ and } mc = 0, \end{cases} \quad (3)$$

where $cc = |\{(f_p^{t_0} \in \mathcal{V}_e, B_e(p)) : B_e(p) = 1\}|$ and $mc = |\{(f_p^{t_0} \in \mathcal{V}_e, B_e(p)) : B_e(p) = 1/2\}|$. Finally, the relying entity e accepts the public key certificate t_0 only if $r_e(t_0) = 1$.

IV. Modification and Extension to Original PGP

As can be seen from the previous section, the original PGP Web of Trust limits the score space to $\mathcal{S} = \{1\}$ and makes use of only direct trust relationships of users. In this section, we modify it to accept the score of -1 implying that a public key certificate is believed to be inauthentic. In addition, we extend it to also consider indirect trust relationships.

A. Modified PGP Web of Trust

Recall that there is only one value in the score space $\mathcal{S}$ of the original PGP Web of Trust, i.e. $s = 1$. This limitation disables a feedback provider from explicitly expressing her belief that a public key certificate is *not* authentic. Besides, it also disables discovery of any conflicting feedback. Therefore we propose to allow a feedback provider to explicitly express her different opinions on the authenticity of a public key certificate. In more technical detail, we propose that instead of signing a public key certificate, a user digitally signs a tuple containing this public

[1]When estimating $r_e(t_0)$, two thresholds for cc and mc are adjustable. By default they are set to 1 and 2, respectively.

key certificate and a score $s \in \{1, -1\}$, where $s = -1$ denotes that she believes the certificate is *not* authentic. Now the new score space is $\mathcal{S} = \{1, -1\}$. Note that the score space $\mathcal{S}$ can have more values to represent user's more granular belief in the authenticity of a public key certificate. But in this paper, we only use two values, for simplicity. This modification enables us to further modify PGP Web of Trust to adopt our proposed approach of expanding trusted neighborhood.

Due to the introduction of the new score -1, the method of verifying a target public key certificate t_0 (see Equation 3) also needs to be modified. The idea of the new method is to first weight the score in each feedback by the trustworthiness of the feedback provider and sum up the weights of scores 1 and -1 respectively. Then, the two total weights will be aggregated to produce the advice as the authenticity of t_0 for the relying entity e. Formally, let w_1 denote the total weight of score 1 and w_{-1} the total wight of score -1. The advice for entity e regarding target t_0 is then given by:

$$r_e(t_0) = \frac{w_1 - w_{-1}}{w_1 + w_{-1}}, \tag{4}$$

where $w_1 = |\sum B_e(p) * s_p|$, $w_{-1} = |\sum B_e(p') * s_{p'}|$, $p, p' \in \mathcal{TN}_e$, $s_p = 1$, and $s_{p'} = -1$. This method ensures that the calculated result $r_e(t_0)$ is in $[-1, 1]$ where 1 and -1 mean that the target t_0 is completely authentic and inauthentic, respectively. The computational cost of the method is also low.

B. Extended PGP Web of Trust

In the original PGP Web of Trust, the trusted neighborhood $\mathcal{TN}$ of a user contains only the trusted neighbors who are directly signed or specified by the user (see Equations 1 and 2). This works well for an experienced user who has specified her trust in many other users (feedback providers). However, for a newcomer who just joined the system and signed only a few users, the original PGP may not be able to provide good advice about the authenticity of many certificates. To address this issue, we extend the original PGP by also considering indirect trust relationships among users. Indirect trust is inferred based on the existing direct trust relationships of all users in the system and through the propagation of trust.[2] In this way, users who are not directly signed or specified by the user but turn out to be highly trustworthy can be found and included in the user's trusted neighborhood.

More specifically, based on the directly trusted neighborhood of all users, a directed graph can be constructed to connect the users (as nodes) together. An edge from a user to another user in the graph represents the direct trust relationship from the first user to the second one. By traversing the graph, there may be multiple paths from a user e to a target user p who is not directly connected with (or directly trusted by) e. Along the i-th path, user e's trust in p can be calculated according to the well-known computational trust model in [2]:

$$CT_i = B_e(p_2) \prod_{j=2}^{m-1} B_{p_j}(p_{j+1}) \tag{5}$$

[2]We assume a trusted central server in charge of the computation of indirect trust and the expansion of trusted neighborhood in the next section. However, we do not assume the server knows the real owner of public keys.

where p_j is the j-th node on the path, $p_1 = e$, and $p_m = p$ (the target user). Note that it is necessary to set a proper parameter to constrain the maximum number of nodes on a path. One concern is the computational complexity. Another concern is that trust propagation may result in unreliable outcomes if the paths are getting too long. It is set 4 in our experiments.

Having the trust of e in p along each valid path, the overall trust e has in p can be aggregated by averaging the trust on each path as $B_e(p) = \frac{1}{n}\sum_{i=1}^{n} CT_i$. Then, a group of all trusted neighbors is selected as:

$$\mathcal{TN}_e{}' = \mathcal{TN}_e \bigcup \{p \in \mathcal{P} : p \notin \mathcal{TN}_e,\ B_e(p) \geq \theta\}, \tag{6}$$

where θ is a threshold for determining the minimum indirect trust needed for a user to be included in the trusted neighborhood. θ should normally be set to be relatively high, i.e. $\theta \geq 1/2$, to minimize the effect of unreliable indirect trust inference caused by the propagation of trust.

V. Further Expansion of Trusted Neighborhood

In this section, we describe a novel method to further expand trusted neighborhood based on direct trust identified by users as well as indirect trust relationships inferred by trust propagation. Our method first merges feedback provided by trusted neighbors and then uses the merged feedback set to find similar users who may become trusted neighbors.

In the first step, our method merges feedbacks provided by trusted neighbors by converting the set of feedbacks provided by the trusted neighbors (including the user herself) about each certificate into a single feedback, if the number of feedbacks about the certificate is large and the feedbacks are consistent. More formally, for a user e with the trusted neighborhood $\mathcal{TN}_e$, a set of feedback provided by the trusted neighbors about a certificate t is denoted as $\mathcal{F}(t)$. We adapt the confidence measure proposed by Wang and Singh [8] to determine whether the set of feedbacks $\mathcal{F}(t)$ about the certificate t can be merged as follows:

$$c(w_1, w_{-1}) = \int_0^1 |\frac{x^{w_1}(1-x)^{w_{-1}}}{\int_0^1 x^{w_1}(1-x)^{w_{-1}} d_x} - 1| d_x \tag{7}$$

The confidence $c(w_1, w_{-1})$ is high only when the number of feedbacks ($w_1 + w_{-1}$) is large and the feedbacks are consistent. Only when the confidence $c(w_1, w_{-1})$ exceeds a predefined threshold λ,[3] the set of feedbacks about the certificate t will be converted to a single feedback as follows:

$$f(t) = \begin{cases} 1 & \text{if } w_1 \geq w_{-1}; \\ -1 & \text{otherwise.} \end{cases} \tag{8}$$

By setting a high confidence threshold, we believe that the merged feedback set from the feedbacks provided by the trusted neighbors of a user e may be fully trusted by e or can well represent e's own opinions about certificates, for two reasons. One reason is that the trusted neighbors who provide the original set of feedbacks are already highly trusted by the user e. Another reason is that the high confidence requirement can further eliminate some potentially unreliable feedbacks.

[3]We will analyze the effect of the confidence threshold on the performance of our method through experiments in Section VI.

Thus, in the second step of our method, based on the merged feedback vector, we find for the user e a set of similar users by measuring the similarity of the merged feedback vector with the feedback vectors of the other users. As similarity has often been used to infer trust in the literature (see Section II), the trusted neighborhood of user e can then be expanded by including those similar users. More specifically, let $\mathcal{F}_e^{'}$ denote the merged feedback vector of user e, and $\mathcal{F}_u$ denote the feedback vector provided by user u. Assume that $\vec{x}$ and $\vec{y}$ are two vectors in k-dimensional space, where $k = |\{t \in \mathcal{T} | f_x(t) \neq \emptyset,\ f_y(t) \neq \emptyset\}|$, $f_x(t) \in \mathcal{F}_e^{'}$ and $f_y(t) \in \mathcal{F}_u$. Then the similarity between two vectors $\mathcal{F}_e^{'}$ and $\mathcal{F}_u$ can be measured by the cosine similarity measure [5]:

$$sim(\mathcal{F}_e^{'}, \mathcal{F}_u) = \frac{\sum_{i=1}^{k} f_x(t) f_y(t)}{\sqrt{\sum_{i=1}^{k} f_x(t)^2} \sqrt{\sum_{i=1}^{k} f_y(t)^2}} \qquad (9)$$

If the similarity $sim(\mathcal{F}_e^{'}, \mathcal{F}_u)$ exceeds a threshold γ, the user u will be included in the trusted neighborhood of user e.

VI. Experimental Evaluation

In this section, we carry out experiments to evaluate the performance (both coverage and accuracy) of the original PGP Web of Trust (denoted as OPGP), the modified version (MPGP), the extended version (EPGP) and our method of further expanding trusted neighborhood applied on the extended version (EPGP+). Our simulated environment involves 500 users, each of whom creates a public key certificate. A certain percentage of these certificates may be inauthentic. Each user can also be a feedback provider. As feedback providers, users are categorized into three groups. In terms of their honesty in signing certificates, they can be honest, neutral or dishonest. Among all the certificates signed by a honest feedback provider, 80% to 100% of them will be truthfully signed. A percentage value will be randomly chosen within this range for each honest feedback provider. Similarly, among all the certificates signed by a neutral and dishonest feedback provider, 40%-60% and 0%-20% of them will be truthfully signed, respectively. Within each category of feedback providers, some of them are experienced, medium experienced or newbies. Experienced feedback providers have signed 15%-20% of all certificates, medium experienced users 8%-13%, and newbies 0%-5%. Each user also indicates her trust on another user when she signs the certificate created by the other user. To reflect the real-life situation, some of the users may make mistakes when specifying their trust on others. Three different types of users are simulated: the ones making many mistakes, the ones making a normal number of mistakes and the ones making a few mistakes. For the users who make many mistakes, the trustworthiness of 80%-100% of others will be wrongly specified by the users. The percentage ranges for the users making a normal number of mistakes and a few mistakes are 40%-60% and 0%-20%, respectively. In our experiments (Section VI-B), we vary the percentages of honest, neutral and dishonest feedback providers, the percentages of experienced, medium experienced users and newbies, and the percentages of users who make many mistakes, a normal number of mistakes and a few mistakes, respectively, to create different possible real-life scenarios. Some other relevant parameters are set as follows. The percentages of authentic and inauthentic public key certificates are 70% and 30% respectively. And, the trust threshold θ for a user to be included in a trusted neighborhood is set to 0.5 for both the methods of EPGP and EPGP+ by default.

As mentioned earlier, the performance of different versions of PGP Web of Trust can be evaluated based on two measures, coverage and accuracy. Coverage measures the extension to which a method is able to predict the authenticity of certificates. In our experiment, the authenticity of a certificate can be predicted for a user if at least one of the feedback providers who have signed the certificate is in the trusted neighborhood of the user. Coverage can then be calculated by averaging across all users the ratio of the predictable certificates to the total number of certificates. Accuracy measures the extent to which the method is able to provide accurate prediction. We use the mean absolute error (MAE) to represent the accuracy:

$$\text{MAE} = \frac{\sum_{j=1}^{M} [\frac{1}{N_j} \sum_{i=1}^{N_j} |r_j(t_i) - g(t_i)|]}{M}, \qquad (10)$$

where N_j is the number of public certificates predictable to user j and M is the total number of users (500 in our case). $r_j(t_i)$ is the estimated authenticity of the certificate t_i by user j, and $g(t_i)$ is the ground truth about the authenticity of t_i. Each experiment has been run for a sufficient number of times.

A. Choosing Proper Parameters

The main purpose of the first set of experiments is to investigate how the performance of our approach (EPGP+) is affected by different values of the parameters in our approach. The two main parameters are the confidence threshold λ and similarity threshold γ. As described in Section V, the confidence threshold is to decide whether a set of feedbacks provided by trusted neighbors for a particular certificate will be merged into a single feedback. The similarity threshold is to decide how similar a user needs to be with another user in order to be added into the trusted neighborhood of the second user. We fix one parameter and study another in two separate experiments. In these experiments, we simulate uniform scenario. More specifically, we have an equal number of honest, neutral, and dishonest feedback providers. We also have an equal number of experienced, medium experienced and new users. The numbers of users who make many mistakes, a normal number of mistakes and a few mistakes are also the same in this uniform scenario.

We first fix the similarity threshold to 0.8 and vary the confidence parameter from 0.0 to 0.95. Results are shown in Figure 1(a). From this figure, we can see that when the confidence threshold increases from 0 up to 0.95, the coverage of our approach increases first and then decreases. Similarly, the accuracy also increases first and decreases later on. When the confidence threshold is set very low, the feedbacks merged from trusted neighbors may not be accurate. Using these inaccurate feedbacks to find similar neighbors will result in a smaller number of users being added into trusted neighborhood. Thus, the coverage is low in this case. Since many

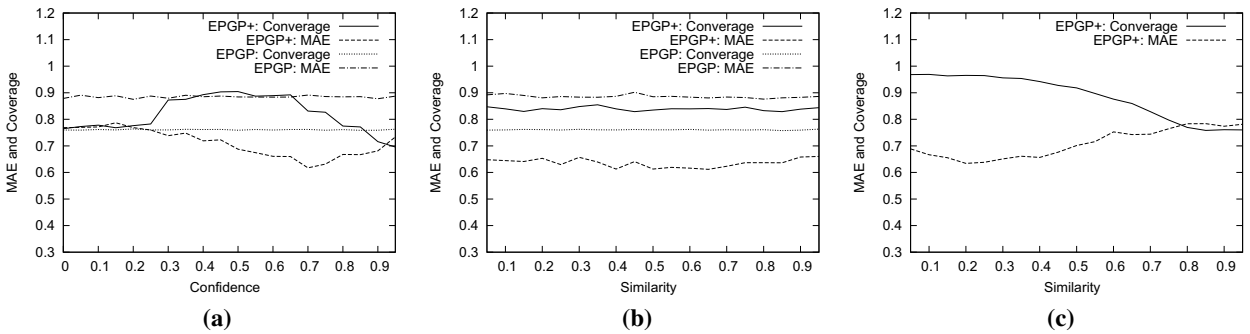

Fig. 1. (a) Performance of Our Approach with Varying Confidence Threshold but with the Fixed Similarity Threshold of 0.8; (b) Performance of Our Approach with Varying Similarity Threshold but with the Fixed Confidence Threshold of 0.7; (c) Performance of Our Approach with Varying Similarity Threshold but with the Fixed Confidence Threshold of 0.2

merged feedbacks are inaccurate, the similar users found are then in fact dishonest. The accuracy is thus decreased (i.e. the MAE value is increased). When the confidence is set very high, not many feedbacks will be merged. The number of similar users found is thus also small. In consequence, the coverage is low in this case. Because the size of the trusted neighborhood is small, the accuracy will also not be high. As shown in the figure, our approach improves the coverage and accuracy of EPGP. More specifically, when the confidence threshold is between 0.2 and 0.8, our approach improves the coverage of EPGP. Our approach improves the accuracy of EPGP for every possible confidence threshold. Based on these results, we choose the confidence threshold of 0.7 where the accuracy is the best and the coverage is relatively high.

We now fix the confidence threshold to 0.7 and vary the similarity threshold from 0 to 0.95. Results are shown in Figure 1(b). When the confidence threshold is set to the proper value, the performance of our approach is not sensitive to the similarity threshold. Our best theory for explaining this phenomenon is that after setting the confidence threshold to the proper value, the similar users included in the trusted neighborhood are honest and well experienced so that further adding more or less similar users into the trusted neighborhood will not affect the performance much. Nevertheless, the similarity threshold of around 0.6 gives relatively better performance. In the experiments in Section VI-B, we will use the confidence threshold of 0.7 and the similarity threshold of 0.6.

In order to show how the performance of our approach is affected by the similarity threshold, in this experiment we fix the confidence threshold to 0.2 and vary the similarity threshold. The results in Figure 1(c) show that both the coverage and accuracy decrease when the similarity threshold increases from 0.0 to 0.95. When the similarity threshold increases, the number of users being added into the trusted neighborhoods will be smaller, thus the coverage will be lower. Because the confidence threshold is set low, the accuracy of merged feedbacks is also low. The similar users found based on these merged feedbacks will be inaccurate as well. Thus, the accuracy of our approach is decreased. Note that the accuracy of our approach when setting both confidence and similarity thresholds to 0.2 is 0.634, which is the best accuracy in Figure 1(c). However, this accuracy is still worse than that when setting confidence threshold to 0.7 and similarity threshold to 0.6, which is 0.616 (see the accuracy of EPGP+ in the uniform scenario in Table I).

B. Effectiveness of Our Approach in Different Scenarios

We vary environmental parameters to create different possible real-life scenarios, to study the effectiveness of our approach for handling those different situations, compared with the other versions of PGP Web of Trust.

1) Uniform Scenario: Here, we present the experimental results of our approach when choosing the proper parameters (0.7 for the confidence threshold and 0.6 for the similarity threshold) in the uniform scenario described in the previous section. Results are summarized in Table I. We can see that MPGP and EPGP can increase the coverage, but the accuracy is much decreased because the inferred indirect trust of users is less accurate compared to direct trust. Interestingly, although the coverage of OPGP is low, its accuracy is higher compared to MPGP and EPGP. The reason is that OPGP considers only the feedback of 1 and thus less untruthful feedbacks will be involved in the estimation of certificate authenticity. Our approach EPGP+ improves both coverage and accuracy of MPGP and EPGP. It is also much better than OPGP in terms of both coverage and accuracy. These performance results in the uniform scenario are used as a benchmark that will be compared by other scenarios.

2) Ideal Scenario: We also inspect an ideal environment where most users intend to be honest in signing certificates and make only few mistakes when identifying directly trusted neighbors. More specifically, the percentage of honest feedback providers is set to 0.8, neutral 0.1 and dishonest 0.1. And the percentage of users making few mistakes is set to 0.8, a normal number of mistakes 0.1, and many mistakes also 0.1. The results in the Ideal Scenario column of Table I show that the performance (especially accuracy) of all approaches is substantially better in this scenario than the uniform scenario. Both the coverage (0.962) and accuracy (0.195) of EPGP+ approach the perfect results.

3) Sparse Scenario: In this experiment, we create a sparse scenario where many users are new to the system and they have not signed many certificates yet. In particular, the system involves only 10% of experienced users and 10% of medium

TABLE I
PERFORMANCE OF APPROACHES FOR UNIFORM, IDEAL, SPARSE AND SUFFICIENT INFORMATION SCENARIOS

	Uniform Scenario		Ideal Scenario		Sparse Scenario		Sufficient Information	
Approaches	Coverage	MAE	Coverage	MAE	Coverage	MAE	Coverage	MAE
OPGP	0.499±0.004	0.744±0.015	0.546±0.004	0.357±0.006	0.143±0.003	0.902±0.045	0.773±0.002	0.622±0.012
MPGP	0.626±0.004	0.788±0.016	0.676±0.003	0.300±0.008	0.223±0.005	0.825±0.017	0.856±0.002	0.782±0.009
EPGP	0.636±0.002	0.794±0.027	0.717±0.007	0.228±0.004	0.385±0.021	0.784±0.022	0.855±0.003	0.773±0.003
EPGP+	0.839±0.008	0.616±0.018	0.962±0.012	0.195±0.008	0.646±0.033	0.709±0.021	0.953±0.007	0.559±0.025

TABLE II
PERFORMANCE OF APPROACHES FOR MANY MISTAKES AND HIGHLY MALICIOUS SCENARIOS

	Many Mistakes		Highly Malicious (a)		Highly Malicious (b)		Highly Malicious (c)	
Approaches	Coverage	MAE	Coverage	MAE	Coverage	MAE	Coverage	MAE
OPGP	0.503±0.009	0.867±0.022	0.307±0.014	0.906±0.036	0.291±0.011	0.932±0.013	0.292±0.009	0.912±0.002
MPGP	0.633±0.003	0.913±0.023	0.512±0.012	1.320±0.027	0.501±0.012	1.359±0.029	0.504±0.005	1.343±0.039
EPGP	0.644±0.005	0.914±0.003	0.521±0.007	1.351±0.021	0.533±0.005	1.329±0.034	0.504±0.006	1.342±0.038
EPGP+	0.827±0.017	0.814±0.053	0.701±0.013	1.502±0.024	0.534±0.008	1.414±0.018	0.676±0.009	1.498±0.037

experienced users. Up to 80% of users are newcomers. In this case, the coverage of different approaches is lower as expected (see the Sparse Scenario column of Table I). In this scenario, the accuracy of EPGP+ is still better than both MPGP and EPGP. Its coverage is also comparably sufficient. Compared with OPGP, our EPGP+ has significantly larger coverage while still maintaining better accuracy.

4) Scenario of Sufficient Information: We also examine another scenario with sufficient information where the majority (80%) of the users are experienced feedback providers who have singed a large number of certificates. Compared with the uniform scenario, the coverage of all approaches is considerably enhanced due to the large number of available feedbacks (see the Sufficient Information column of Table I). However, for OPGP, MPGP and EPGP, the accuracy is either decreased or stays the similar. The accuracy of our EPGP+ is still increased. Our approach can also effectively improve the coverage and accuracy of OPGP, MPGP and EPGP.

5) Scenario of Many Mistakes: We then imitate a scenario where many (80%) users make a lot of mistakes when they sign direct trust in another user. In this case, the coverage of different approaches stays the similar as that in the uniform scenario (see the Scenario of Many Mistakes column in Table II), whereas the accuracy has been largely decreased. However, the accuracy of EPGP+ is still the best among all the approaches in this scenario.

6) Highly Malicious Scenario: Finally, we simulate a highly malicious situation in which there are 80% of feedback providers are dishonest. In this kind of scenarios, our approach does not show any advantages (see the Highly Malicious (a) column in Table II). We then set confidence threshold to 1 and the similarity threshold to 1 and adjust the trust threshold θ to 0.8 respectively. From the results in the Highly Malicious (b) and (c) columns of Table II, we can notice that our EPGP+ does not hold any advantages either in this extreme case. We also notice that both MPGP and EPGP do not work well either in the highly malicious scenario. A reasonable explanation to such phenomena is that when the environment is filled with malicious users, the possibility of incorrectly inferring trusted neighbors is high. As opposed to the other scenarios, we may instead rely on other methods such as OPGP.

VII. CONCLUSIONS AND FUTURE WORK

To conclude, the main contributions of our current work include: 1) the modification on PGP Web of Trust to also accept negative feedbacks in digital signatures for public key certificates indicating that the certificates are believed to be inauthentic; 2) the extension on PGP Web of Trust to also consider indirect trust relationships of users inferred through trust propagation so that trusted neighborhood of users gets expanded; 3) a novel method for the further expansion of trusted neighborhood of users by merging feedbacks provided by the trusted neighbors and finding similar users based on the merged feedbacks; 4) detailed experimental evaluation confirming the value of our proposed method in different simulated real-life scenarios. For future work, we will investigate how to accurately quantify the trust a user should place on the merged feedback set based on the trust the user has of the trusted neighbors. By doing so, we expect our method to be more robust to highly malicious environments.

Acknowledgement: This work has been made possible thank to the Institute for Media Innovation who has given a scholarship to the first author, and Qin Li for his invaluable helps.

REFERENCES

[1] A. Abdul-Rahman, "The pgp trust model," in *EDI-Forum: the Journal of Electronic Commerce*, vol. 10, no. 3, 1997, pp. 27–31.
[2] L. Mui, M. Mohtashemi, and A. Halberstadt, "A computational model of trust and reputation," in *Proceedings of the 35th Annual Hawaii International Conference on System Sciences*, 2002, pp. 2431 – 9.
[3] R. Guha, R. Kumar, P. Raghavan, and A. Tomkins, "Propagation of trust and distrust," in *Proceedings of the 13th International Conference on World Wide Web*, 2004.
[4] J. Huang and D. Nicol, "A formal-semantics-based calculus of trust," *IEEE Internet Computing*, vol. 14, no. 5, pp. 38–46, 2010.
[5] G. Linden, B. Smith, and J. York, "Amazon.com recommendations: item-to-item collaborative filtering," *IEEE Internet Computing*, vol. 7, pp. 76 – 80, 2003.
[6] S. Moghaddam, M. Jamali, M. Ester, and J. Habibi, "Feedbacktrust: using feedback effects in trust-based recommendation systems," in *Proceedings of the ACM Conference on Recommender systems*, 2009.
[7] J. Zhang and R. Cohen, "Evaluating the trustworthiness of advice about seller agents in e-marketplaces: A personalized approach," *Electronic Commerce Research and Applications*, vol. 7, no. 3, pp. 330–340, 2008.
[8] Y. Wang and M. P. Singh, "Formal trust model for multiagent systems," in *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI)*, 2007.