Identifying Fake Feedback for Effective Trust Management in Cloud Environments

Talal H. Noor[1], Quan Z. Sheng[1], Abdullah Alfazi[1], Jeriel Law[1], and Anne H.H. Ngu[2]

[1] School of Computer Science, The University of Adelaide, SA 5005, Australia
{talal,qsheng}@cs.adelaide.edu.au
[2] Department of Computer Science, Texas State University, TX 78666-4616, USA
angu@txstate.edu

Abstract. Managing trust in cloud environments is emerging as an important issue in recent years. The highly dynamic, distributed, and non-transparent nature of cloud services makes the trust management of these services difficult. Malicious users may collude to give multiple misleading trust feedback to disadvantage a cloud service, or create several accounts and then leave misleading trust feedback to trick users into trusting cloud services that are not actually trustworthy. In this paper, we propose techniques enabling the identification of fake trust feedbacks and thus provide significant improvement on trust management in cloud environments. In particular, we introduce a credibility model that not only identifies credible trust feedbacks from fake ones, but also preserves the privacy of cloud service consumers. The techniques have been validated by a prototype system implementation and experimental studies.

Keywords: Trust management, cloud computing, credentials, credibility, reputation, security, privacy.

1 Introduction

In recent years, cloud computing has been receiving much attention as a new computing paradigm for providing flexible and on-demand infrastructures, platforms and software as services. Unfortunately, trust management is still considered as one of the key challenges in the adoption of cloud computing. According to the researchers at UC Berkeley [2], trust management and security are ranked one of the top 10 obstacles for cloud computing. The highly dynamic, distributed, and non-transparent nature of cloud services makes the trust management in cloud environments even more challenging [2,10,14,15].

Several researchers have recognized the significance of trust management and proposed solutions to assess and manage trust based on feedbacks collected from participants [5,12,6,14]. In reality, it is not unusual that a trust management system experiences malicious behaviors (i.e., attacks) from its users. For instance, auction systems such as eBay, experience fake trust feedback problem where attackers create several accounts and quickly leave multiple trust feedbacks to

A. Ghose et al. (Eds.): ICSOC 2012, LNCS 7759, pp. 47–58, 2013.

boost their trust results at a glance [1]. Attackers trick users into trusting untrustworthy cloud services through creating several accounts, producing numerous transactions (e.g., creating multiple virtual machines for a short period of time), and leaving fake trust feedbacks.This paper focuses on improving trust management in cloud environments by proposing novel ways to identify fake feedbacks. In particular, we distinguish several key issues including i) *Feedback Collusion* where dynamic interactions of cloud services make the identification of credible trust feedbacks a difficult problem because new cloud service consumers join while others might leave around the clock, ii) *Multiplicity of Identities* where malicious cloud service consumers can use multiple identities [9] to give fake or misleading trust feedbacksor to be able to wipe off their negative historical trust records, and iii) *Cloud Service Consumers' Privacy* where interactions between the cloud service consumers and the trust management service usually involve sensitive information. The cloud service consumers might face sensitive information disclosure while dealing with the trust management service. For example, the accidental leaking of the trust participant's sensitive information such as user name, password, and postal address. is not possible.

In this paper, we overview the design and the implementation of a credibility model. Our model exploits techniques for fake feedback identification and thus provides significant improvement on trust management in cloud environments. In a nutshell, the salient features of our model are as follows:

- *Zero-Knowledge Credibility Proof Protocol.* To avoid the privacy breach problem when managing trust in cloud environments, we introduce the *Zero-Knowledge Credibility Proof Protocol* that not only preserves the cloud service consumers' privacy, but also enables the trust management service to prove the credibility of a particular cloud service consumer's feedbacks.
- *Feedback Density.* We propose this technique to tackle the feedback collusion issue by identifying credible trust feedbacks from fake ones (i.e., trust results manipulation by giving multiple trust feedbacks to a certain cloud service in a short period of time).
- *Multi-Identity Recognition.* We introduce this technique that addresses the multiplicity of identities challenge. This technique identifies fake trust feedbacks from malicious cloud service consumers who use multiple identities to manipulate trust results

The remainder of the paper is organized as follows. The design of the Zero-Knowledge Credibility Proof Protocol, assumptions and attack models are described in Section 2. Section 3 describes the details of our credibility model. Section 4 reports the trust management service implementation and several experimental evaluations for the proposed techniques. Finally, Section 5 overviews the related work and provides some concluding remarks.

2 Zero-Knowledge Credibility Proof Protocol (ZKC2P)

Since there is a strong relation between trust and identification as emphasized in [7], we propose that the *Identity Management Service* (IdM) can help the

Trust Management Service (TMS) in measuring the credibility of a cloud service consumer's feedback. However, processing IdM's information can breach the privacy of cloud service consumers. One way to preserve privacy is to use cryptographic encryption techniques but there is no efficient ways to process encrypted data [15]. Another way is to use anonymization techniques to process IdM's information without breaching the privacy of cloud service consumers. Thus, we propose a *Zero-Knowledge Credibility Proof Protocol* (ZKC2P) to allow TMS to process IdM's information (i.e., credentials) using the *Multi-Identity Recognition* factor (explained in detail in Section 3). TMS processes the credentials without including the sensitive information. Instead, anonymized information is used via consistent hashing (e.g., sha-256). The anonymization process covers all the credentials' attributes except the *Timestamps* attribute.

Identity Management Service (IdM). The cloud service consumers typically have to establish their identity for the first time they attempt to use TMS through registering their credentials at the *Trust Identity Registry*. These credentials typically involve sensitive information. The *Trust Identity Registry* stores an identity record denoted as $\mathcal{I}$ for each cloud service consumer. The identity record $\mathcal{I}$ is represented in a tuple that consists of the cloud service consumer's primary identity $\mathcal{C}$ (e.g., user name), the credentials' attributes (e.g., passwords, postal address, IP address) denoted by $\mathcal{C}_a$ and the timestamps $\mathcal{T}$ which is the cloud service consumer's registration time in TMS. The identity record is thus a tuple $\mathcal{I} = (\mathcal{C}, \mathcal{C}_a, \mathcal{T})$.

Trust Management Service (TMS). The trust behavior of a cloud service is represented by a collection of invocation history records denoted as $\mathcal{H}$. Each cloud service consumer c holds her point of view regarding the trustworthiness of a specific cloud service s in the invocation history record which is managed by TMS. Each invocation history record is represented in a tuple that consists of the cloud service consumer's primary identity $\mathcal{C}$, the cloud service's identity $\mathcal{S}$, a set of trust feedbacks $\mathcal{F}$ and the aggregated trust feedbacks weighted by the credibility $\mathcal{F}_c$ (i.e., $\mathcal{H} = (\mathcal{C}, \mathcal{S}, \mathcal{F}, \mathcal{F}_c)$). Each trust feedback in $\mathcal{F}$ is represented in numerical form with the range of [0, 1], where 0, 1, and 0.5 means *negative feedback*, *positive feedback*, and *neutral* respectively. Consequently, the trustworthiness of a certain cloud service s, the trust result, denoted as $\mathcal{T}_r(s)$, is calculated as follows:

$$\mathcal{T}_r(s) = \frac{\sum_{c=1}^{|\mathcal{V}(s)|} \mathcal{F}_c(c, s)}{|\mathcal{V}(s)|} \tag{1}$$

where $\mathcal{V}(s)$ denotes the trust feedbacks given to the cloud service s and $|\mathcal{V}(s)|$ represents the length of $\mathcal{V}(s)$ (i.e., the total number of trust feedbacks given to the cloud service s). $\mathcal{F}_c(c, s)$ are trust feedbacks from the c^{th} cloud service consumer weighted by the credibility. TMS distinguishes between credible trust feedbacks and malicious ones through assigning the credibility aggregated weights $\mathcal{C}_r(c, s)$ to trust feedbacks $\mathcal{F}(c, s)$ as shown in Equation 2, where the result $\mathcal{F}_c(c, s)$ is

held in the invocation history record h and updated in TMS. The details on how to calculate $\mathcal{C}_r(c, s)$ is described in Section 3.

$$\mathcal{F}_c(c, s) = \mathcal{F}(c, s) * \mathcal{C}_r(c, s) \tag{2}$$

In our credibility model, we assume that communications are secure since securing communications is not the focus of this paper. The attack such as *Man-in-the-Middle* (MITM) is therefore beyond the scope of this work. We also assume that IdM is managed by a trusted third party. The attacks that we consider are as follows:

- *Self-promoting Attack.* This attack arises when the malicious cloud service consumers attempt to increase their trust results [8] or their allies. This type of attack can occur either as an *Non-collusive Malicious Behavior* (i.e., when a malicious cloud service user gives numerous fake feedbacks to increase her trust results) or as a *Collusive Malicious Behavior* (i.e., when several users collaborate to give numerous fake feedbacks) also called *Feedback Collusion.*
- *Slandering Attack.* This attack is considered as the opposite of the *Self-promoting* attack that happens when malicious users try to decrease the trust results of certain cloud service [3] due to e.g., jealousy from its competitors. This type of attack can also happen either through *Non-collusive Malicious Behavior* or *Collusive Malicious Behavior.*
- *Sybil Attack.* This attack arises when malicious cloud service consumers use multiple identities [9,8] to give numerous misleading trust feedbacks to increase their allies' trust results or to decrease their competitors' trust results.
- *Whitewashing Attack.* This attack is similar to the *Sybil* attack in the use of multiple identities but differs in the purpose. The *Whitewashing* attack occurs when the malicious cloud service consumers seek new identities to clean their negative historical trust records [11].

3 The Credibility Model

Since the trust behavior of a cloud service is represented by a collection of invocation history records that contain cloud service consumers' trust feedbacks, there is a considerable possibility of TMS receiving *fake* trust feedbacks from vicious cloud service consumers. To overcome these issues, we propose a *credibility model*, which considers several factors including the *Feedback Density* and the *Multi-Identity Recognition.*

Feedback Density. Some malicious cloud service consumers may give numerous fake trust feedbacks to manipulate trust results for cloud services (i.e.,*Self-promoting* and *Slandering* attacks). Several online reputation-based systems such as eBay[1] have used the number of trusted feedbacks to help their consumers to overcome such attacks. The number of trusted feedbacks gives the evaluator a

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

hint in determining the feedback credibility [16]. However, the number of trust feedbacks is not enough in determining the credibility of trust feedbacks because a *Self-promoting* attack might have been performed on cloud services.

In order to overcome this problem, we introduce the concept of *Feedback Density* to support the determination of credible trust feedbacks. Specifically, we consider the total number of cloud service consumers who gave trust feedbacks to a particular cloud service as the *Feedback Mass*, the total number of trust feedbacks given to the cloud service as the *Feedback Volume*. The feedback volume is influenced by the *Feedback Volume Collusion* factor which is controlled by a specified volume collusion threshold. This factor regulates the multiple trust feedbacks extent that could collude the overall trust feedback volume. For instance, if the volume collusion threshold is set to 5 feedbacks, any cloud service consumer c who gives more than 5 feedbacks is considered to be suspicious of involving in a feedback volume collusion. The feedback density of a certain cloud service s, $\mathcal{D}(s)$, is calculated as follows:

$$\mathcal{D}(s) = \frac{\mathcal{M}(s)}{|\mathcal{V}(s)| * \mathcal{L}(s)} \tag{3}$$

where $\mathcal{M}(s)$ denotes the total number of cloud service consumers who gave trust feedbacks to the cloud service s (i.e., the *Feedback Mass*). $|\mathcal{V}(s)|$ represents the total number of trust feedbacks given to the cloud service s (i.e., the *Feedback Volume*). $\mathcal{L}(s)$ represents the *Feedback Volume Collusion* factor, calculated as follows:

$$\mathcal{L}(s) = 1 + \left(\frac{\sum_{h \in \mathcal{V}(s)} \left(\sum_{c=1}^{|\mathcal{V}_c(c,s)|} \left(\sum_{|\mathcal{V}_c(c,s)| > e_v(s)} |\mathcal{V}_c(c,s)| \right) \right)}{|\mathcal{V}(s)|} \right) \tag{4}$$

This factor is calculated as the ratio of the number of trust feedbacks given by cloud service consumers $|\mathcal{V}_c(c,s)|$ who give feedbacks more than the specified volume collusion threshold $e_v(s)$ over the total number of trust feedbacks received by the cloud service $|\mathcal{V}(s)|$. The idea is to reduce the value of the multiple trust feedbacks which are given diversely from the same cloud service consumer.

For instance, suppose there are two different cloud services x and y. Both cloud services have the same total number of trust feedbacks (i.e., $|\mathcal{V}(x)| = 150$ and $|\mathcal{V}(y)| = 150$) and very close aggregated feedbacks (e.g., x has 89% positive feedbacks and y has 92% positive feedbacks). However, the *Feedback Mass* of the cloud service x is higher than the cloud service y (i.e., $\mathcal{M}(x) = 20$ and $\mathcal{M}(y) = 5$). If the volume collusion threshold e_v is set to 10 feedbacks per cloud service consumer. Only 4 cloud service consumers gave more than 10 feedbacks to the cloud service x where the total number of their trust feedbacks $|\mathcal{V}_c(c,x)| = 60$ feedbacks; while 2 cloud service consumers gave more than 10 feedbacks to the cloud service y where the total number of their trust feedbacks $|\mathcal{V}_c(c,y)| == 136$ feedbacks. According to Equation 3, the *Feedback Density* of the cloud service x is higher than cloud service y (i.e., $\mathcal{D}(x) = 0.0953$ and $\mathcal{D}(y) = 0.0175$).

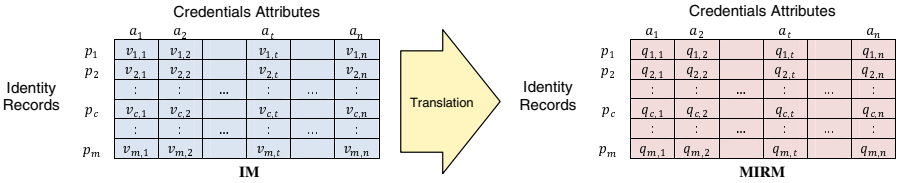


Fig. 1. IM Translation to MIRM

In other words, the higher the *Feedback Density*, the more credible the aggregated feedbacks are (i.e.,the higher possibility of collusion if vice versa).

Multi-Identity Recognition. Since the cloud service consumers have to register their credentials at the *Trust Identity Registry*, we believe that there is a possibility of a *Multi-Identity Recognition* by comparing the cloud service consumers' credentials attributes values from the identity records $\mathcal{I}$. The main goal in the multi-identity recognition factor is to protect TMS from malicious cloud service consumers who use multiple identities (i.e., *Whitewashing* and *Sybil* attacks) to manipulate the trust results. In a typical *Trust Identity Registry*, the entire identity records $\mathcal{I}$ are represented as a list of m cloud service consumers' primary identities $\mathcal{C}_p = \{p_1, p_2, ..., p_m\}$ (e.g., user name) and a list of n credentials' attributes $\mathcal{C}_a = \{a_1, a_2, ..., a_n\}$ (e.g., passwords, postal address, IP address, computer name, etc.). In other words, the entire $\mathcal{C}_p \times \mathcal{C}_a$ (Cloud Service Consumer's Primary Identity-Credentials' Attributes) Matrix, denoted as IM, covers all cloud service consumers who registered their credentials in TMS. The credential attribute value for a particular cloud service consumer $v_{c,t}$ is stored in TMS without including credentials with sensitive information using ZKC2P as mentioned earlier in Section 2.

We believe that TMS can identify patterns in cloud service consumers' anonymous credentials. There is a high possibility that malicious cloud service consumers use similar credentials in different identity records $\mathcal{I}$. Thus, we translate IM to the *Multi-Identity Recognition Matrix*, denoted as $MIRM$, which similarly covers the entire identity records $\mathcal{I}$ represented as the entire $\mathcal{C}_p \times \mathcal{C}_a$. However, the value for a particular cloud service consumer $q_{c,t}$ in the new matrix represents the frequency of the credential attribute value for the same particular cloud service consumer $v_{c,t}$ in the same credential attribute as shown in Figure 1.

The frequency $q_{c,t}$ of a particular credential attribute value $v_{c,t}$ is calculated as the times of appearance $\mathcal{A}_p$ that the credential value appears in the t^{th} credential attribute normalized by the total number of identity records (i.e., the length of a_t) as follows:

$$q_{c,t} = \frac{\sum_{c=1}^{c=m} (\mathcal{A}_p(v_{c,t}))}{|a_t|} \tag{5}$$

Then, the *Multi-Identity Recognition* factor $\mathcal{M}_{id}$ is calculated as the sum of frequencies of each credential attribute value for a particular cloud service consumer normalized by the total number of identity record as follows:

$$\mathcal{M}_{id}(c) = 1 - \left(\sum_{t=1}^{t=n} q_{c,t} \right) \tag{6}$$

where the sum of $q_{c,t}$ represents the similar credentials distributed over different identity records $\mathcal{I}$ and $\mathcal{M}_{id}(c)$ represents the opposite (i.e., at least that the cloud service consumer has fairly unique credentials).

Based on the specified trust feedback credibility factors (i.e., feedback density and multi-identity recognition), TMS distinguishes between credible trust feedbacks and fake ones through assigning the credibility aggregated weights $\mathcal{C}_r(c, s)$ to each trust feedback as shown in Equation 2. $\mathcal{C}_r(c, s)$ is calculated as follows:

$$\mathcal{C}_r(c, s) = \frac{\rho * \mathcal{D}(s) + \Omega * \mathcal{M}_{id}(c)}{\lambda} \tag{7}$$

where ρ and $\mathcal{D}(s)$ denote the *Feedback Density* factor's normalized weight (i.e., parameter) and the factor's value respectively. The second part of the equation represents the *Multi-Identity Recognition* factor where Ω denotes the factor's normalized weight and $\mathcal{M}_{id}(c)$ denotes the factor's value. λ represents the number of factors used to calculate $\mathcal{C}_r(c, s)$. For example, if we only consider feedback density, λ will be 1; if we consider both the feedback density and the multi-identity recognition, λ will be 2.

4 Implementation and Experimental Evaluation

In this section, we report the *Trust Management Service's* architecture and implementation. The trust management service's architecture evolved from our previous efforts in the *Trust as a Service* (TaaS) framework [14]. Figure 2 depicts the main components of the architecture. The *Providers* component represent the cloud service providers who provide cloud services such as Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS). The *Consumers* component represent the cloud service consumers who rent cloud services. The *Registry Service* component allows cloud service providers to advertise their services through the service registry and allows both the trust management service and cloud service consumers to access the service registry to discover cloud services. The *Identity Management Service* component manages cloud service consumers' identity records. The *Trust Management Service* component consists of four layers. The *Trust Data Provisioning* layer discovers the cloud services ID through the *Cloud Services ID Discoverer* module, collects cloud service consumers' trust feedbacks using the *Trust Feedbacks Collector* module and request cloud service consumers' credentials using the *Zero-Knowledge Credibility Proof Protocol* (ZKC2P) module. The *Trust Assessment Function* layer requests and translates anonymized credentials and recognizes multiple identities using the

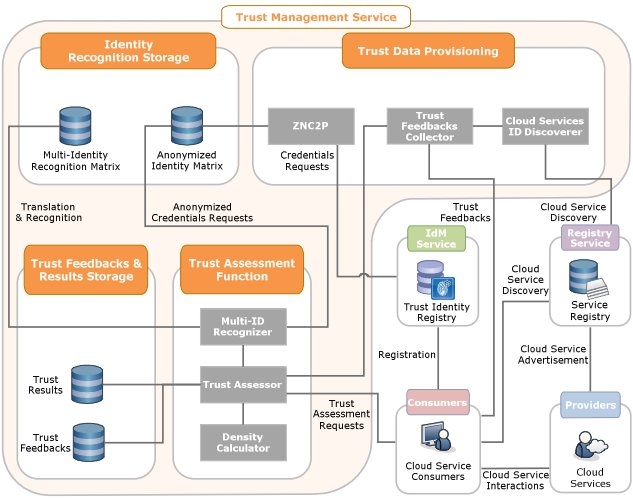


Fig. 2. System Architecture

Multi-ID Recognizer module, calculates the feedback density using the *Density Calculator* module and handle trust queries from cloud service consumers and assesses the trustworthiness of a particular cloud service using the *Trust Assessor* module. The *Identity Recognition Storage* layer stores the anonymized identity matrix and the Multi-Identity recognition matrix. The *Trust Feedbacks and Results Storage* layer stores trust feedbacks from cloud service consumers and trust results for cloud services.

Since it is hard to find some publicly available real-life trust data sets and user credentials, in our experiments, we used Epinions[2] rating data set which was collected by Massa and Avesani [13] and augmented a set of credentials for each corresponding user in Epinions rating data set. The data set has 49,290 users, 139,738 items and 664,824 trust feedbacks. We choose Epinions data set because it is similar in data structure (i.e., consumers' opinions on specific products and services) with our cloud service consumer trust feedbacks. In particular, we considered `user_id` in Epinions as the cloud service consumer's primary identity $\mathcal{C}$, `item_id` as the cloud service's identity $\mathcal{S}$ and we normalized the `rating_value` as the cloud service consumers' trust feedbacks $\mathcal{F}$ to scale [0, 1].

To validate the applicability of our approach, we have imported the Epinions data set for a set of randomly selected cloud services that we are intending to analyze and the number of cloud service consumers is set to one hundred. We evaluate the trust robustness of our credibility model against malicious behaviors of the cloud service consumers. In particular, we conducted experiments under two settings of malicious behaviors namely: the *Feedback Collusion Behavior* and the *Multiplicity of Identities Behavior*. In the *Feedback Collusion Behavior* setting, we aggregate trust results for the cloud services for 10 rounds by varying

[2] `http://www.trustlet.org/wiki/Downloaded_Epinions_dataset`

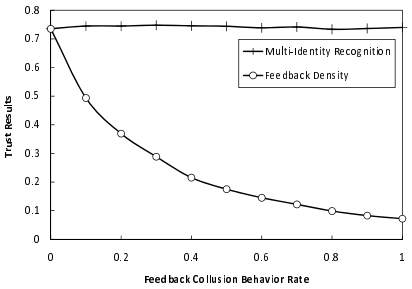
Table 1. Experiment Factors and Parameters Setup

Experiment Design	ρ	Ω	λ	$\mathcal{C}r(c,s)$
With Credibility factors	1	1	2	
Without Credibility factors				1
Feedback Density factor	1	0	1	
Multi-Identity Recognition factor	0	1	1	

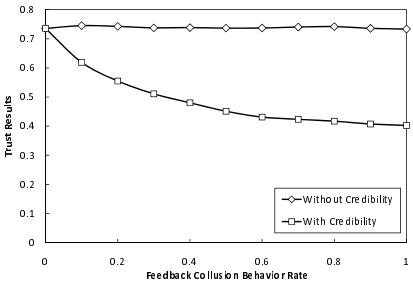
the number of feedbacks by 10% of each round (i.e., *Malicious Behavior Rate*) in which cloud service consumers act dishonestly to manipulate the trust result of the selected cloud services (i.e., the *Self-promoting* attack and the *Slandering* attack). Similarly, in the *Multiplicity of Identities Behavior* setting, we aggregate trust results for the cloud services for 10 rounds by varying the number of cloud service consumers by 10% of each round (i.e., *Multiplicity of Identities Behavior Rate*) in which cloud service consumers act dishonestly by using multiple identities to manipulate trust results of the selected cloud services (i.e., the *Whitewashing* attack and the *Sybil* attack).

To evaluate the trust robustness of our credibility model (i.e., with respect to *Malicious Behavior Rate* and *Multiplicity of Identities Behavior*), we use two experimental designs namely: i) measuring the trust result robustness for each factor in our credibility model including the *Feedback Density* and the *Multi-Identity Recognition*, and ii) measuring the trust result robustness with credibility factors and without the credibility factors (i.e., turning $\mathcal{C}_r(c,s)$ to 1 for all trust feedbacks). The parameters setup for each corresponding experiment factor is depicted in Table 1.

Feedback Collusion Behavior. Figure 3(a) shows the trust result robustness for each factor in our credibility model including the *Feedback Density* and the *Multi-Identity Recognition*. We can observe that the higher the feedback collusion behavior rate, the lower the trust results when considering to calculate the trust based on the feedback density factor only. On the other hand, the trust results show nearly no change to the feedback collusion behavior rate when considering to calculate the trust based on the multi-identity recognition factor. This is true because the malicious cloud service consumers manipulate trust results only through giving multiple fake trust feedbacks. Figure 3(b) depicts the trust result robustness with our proposed credibility factors and without the credibility factors. We note that the higher the feedback collusion behavior rate the lower trust results are when considering to calculate the trust with all credibility factors. We also note that trust results do not differ much when obtained without considering the credibility factors. This indicates that cloud service consumers may make bad decisions by choosing untrustworthy cloud services instead of trustworthy ones when not taking the credibility factors into account. As a result, our credibility model is robust and more sensitive in detecting the feedback collusion behaviors because of the feedback density factor.

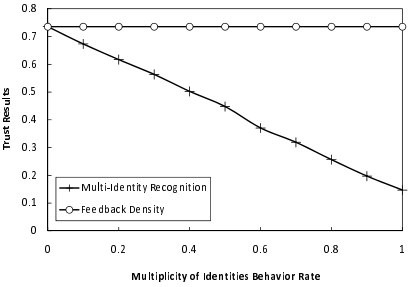


(a) Credibility Factors

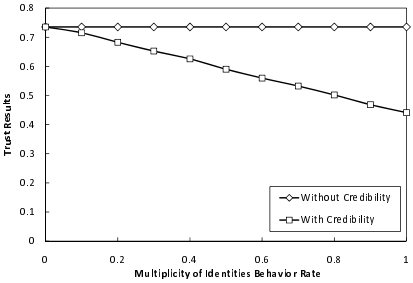


(b) With Vs. Without Credibility

Fig. 3. Feedback Collusion Behavior



(a) Credibility Factors



(b) With Vs. Without Credibility

Fig. 4. Multiplicity of Identities Behavior

Multiplicity of Identities Behavior. Figure 4(a) shows the trust result robustness for each factor in our credibility model including the *Feedback Density* and the *Multi-Identity Recognition*. We can see that trust results are not affected by the multiplicity of identities behavior rate when calculated based on the feedback density factor only. However, the trust results are very sensitive to the multiplicity of identities behavior rate and drops almost linearly when calculated based on the multi-identity recognition factor only. This is because the malicious cloud service consumers manipulate trust results only through using multiple identities. Figure 4(b) depicts the trust result robustness with credibility factors and without the credibility factors. We can observe that trust results response effectively to the multiplicity of identities behavior rate when calculating the trust using the proposed credibility factors, which are not the case when the proposed credibility factors are not considered. This means that without using our proposed credibility factors, a trust management system can be easily fooled by malicious cloud service consumers who create multiple accounts and quickly leave fake trust feedbacks. As a result, based on the experimental results, we can see that our proposed credibility model can effectively detect multiplicity of identities behaviors.

5 Discussions and Conclusion

Trust management is considered as one of the important issues in cloud computing and is becoming a very active research area in recent years. Several research works have proposed trust management techniques such as policy-based trust management. Brandic et al. [5] proposed a compliant management approach to help cloud service consumers to choose trustworthy cloud services. The approach is developed using a centralized architecture to establish trust between cloud service consumers and cloud service providers. Hwang et al. [10] proposed a security aware cloud architecture that assess the trust for both the cloud service provider and the cloud service consumers. To assess the trustworthiness of cloud service providers, they developed a trust negotiation and data coloring (integration) approach using fuzzy logic techniques. To assess the trustworthiness of cloud service consumers, they exploited the Distributed-Hash-Table (DHT)-based trust-overlay networks among several data centers to deploy a reputation-based trust management technique. Unlike previous works which did not consider the problem of fake trust feedbacks or require extensive computations and the trust participants' collaboration by rating trust feedbacks, we present a credibility model that include several metrics namely the *Feedback Density* and the *Multi-Identity Recognition* that assess cloud services' trustworthiness and identify credible trust feedbacks from fake ones.

Other research works have proposed trust management techniques such as reputation-based trust management. Malik and Bouguettaya [12] proposed several reputation metrics to assess the credibility of the rater such as personal experience for credibility evaluation, majority rating, past rating history, etc. The authors also proposed reputation assessment techniques for Web services based on the existing quality of service (QoS) parameters. Conner et al. [6] focused on assessing the trustworthiness of service requesters by proposing a trust management framework for the service-oriented architecture (SOA). This framework has a decentralized architecture that offers multiple trust evaluation metrics, allowing service providers to have customized evaluations. Unlike previous works, we were inspired by Xiong and Liu who differentiate between the credibility of a peer and the credibility of a feedback [16]. However, their approach is not applicable in cloud environments because peers supply and consume services and they are evaluated on that base. Our work was also inspired by Bertino et al. [4] in developing ZKC2P and the use of zero knowledge proofs but instead of using it for verification purposes, ZKC2P is used to measure the feedback credibility.

In the future, we plan to deal with more challenging problems in trust management such as occasional and periodic attacks detection. We also plan to identify new patterns in cloud service consumers' behaviors that can potentially help identify credible trust feedbacks from the fake ones. The performance optimization of the trust management service is another focus of our future work.

Acknowledgments. Talal H. Noor and Abdullah Alfazi work has been supported by King Abdullah's Postgraduate Scholarships, the Ministry of Higher Education: Kingdom of Saudi Arabia.

References

1. Akwagyiram, A.: How Do You Catch Online Auction Cheats? The British Broadcasting Corporation News (BBC) (July 2010), http://www.bbc.co.uk/news/10494724 (accessed May 16, 2012)
2. Armbrust, M., Fox, A., Griffith, R., Joseph, A., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I., Zaharia, M.: A View of Cloud Computing. Communications of the ACM 53(4), 50–58 (2010)
3. Ba, S., Pavlou, P.: Evidence of the Effect of Trust Building Technology in Electronic Markets: Price Premiums and Buyer Behavior. MIS Quarterly 26(3), 243–268 (2002)
4. Bertino, E., Paci, F., Ferrini, R., Shang, N.: Privacy-preserving Digital Identity Management for Cloud Computing. IEEE Data Eng. Bull. 32(1), 21–27 (2009)
5. Brandic, I., Dustdar, S., Anstett, T., Schumm, D., Leymann, F., Konrad, R.: Compliant Cloud Computing (C3): Architecture and Language Support for User-Driven Compliance Management in Clouds. In: Proc. of CLOUD 2010, Miami, Florida, USA (July 2010)
6. Conner, W., Iyengar, A., Mikalsen, T., Rouvellou, I., Nahrstedt, K.: A Trust Management Framework for Service-Oriented Environments. In: Proc. of WWW 2009, Madrid, Spain (April 2009)
7. David, O., Jaquet, C.: Trust and Identification in the Light of Virtual Persons (June 2009), http://www.fidis.net/resources/deliverables/identity-of-identity/ (accessed May 10, 2012)
8. Douceur, J.R.: The Sybil Attack. In: Druschel, P., Kaashoek, M.F., Rowstron, A. (eds.) IPTPS 2002. LNCS, vol. 2429, pp. 251–260. Springer, Heidelberg (2002)
9. Friedman, E., Resnick, P., Sami, R.: Manipulation-Resistant Reputation Systems. In: Algorithmic Game Theory, pp. 677–697. Cambridge University Press, New York, USA (2007)
10. Hwang, K., Li, D.: Trusted Cloud Computing with Secure Resources and Data Coloring. IEEE Internet Computing 14(5), 14–22 (2010)
11. Lai, K., Feldman, M., Stoica, I., Chuang, J.: Incentives for Cooperation in Peer-to-Peer Networks. In: Proc. of the 1st Workshop on Economics of Peer-to-Peer Systems, Berkeley, CA, USA (June 2003)
12. Malik, Z., Bouguettaya, A.: RATEWeb: Reputation Assessment for Trust Establishment Among Web Services. The VLDB Journal 18(4), 885–911 (2009)
13. Massa, P., Avesani, P.: Trust Metrics in Recommender Systems. In: Computing with Social Trust. Human-Computer Interaction Series. Springer
14. Noor, T.H., Sheng, Q.Z.: Trust as a Service: A Framework for Trust Management in Cloud Environments. In: Bouguettaya, A., Hauswirth, M., Liu, L. (eds.) WISE 2011. LNCS, vol. 6997, pp. 314–321. Springer, Heidelberg (2011)
15. Pearson, S., Benameur, A.: Privacy, Security and Trust Issues Arising From Cloud Computing. In: Proc. of CloudCom 2010, Indianapolis, Indiana, USA (November-December 2010)
16. Xiong, L., Liu, L.: Peertrust: Supporting Reputation-based Trust for Peer-to-Peer Electronic Communities. IEEE TKDE 16(7), 843–857 (2004)