International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 5, Issue 1, No 27, 2017 ISSN: 2348-6600
Reference ID: IJCS-290 PAGE NO: 1904-1917

Enhancement Supporting Reputation-Based Trust Management for Cloud Services

Stud. K.kavitha [#1] Prof. Mr.K.M.Subramanian M.E.,(Ph.D.), [#2]

[# 1 2] *Department of Computer Science,Erode Sengunthar Engineering College ,Thudupathi, Perundurai,Erode-638507,Tamilnadu,India.*
[1] kavitharajecse@gmail.com[2] rajemanik123@gmail.com

Abstract - Trust management is one of the most challenging issues for the adoption and growth of cloud computing. The highly dynamic, distributed, and non-transparent nature of cloud services introduces several challenging issues such as privacy, security, and availability. Preserving consumers' privacy is not an easy task due to the sensitive information involved in the interactions between consumers and the trust management service. Protecting cloud services against their malicious users (e.g., such users might give misleading feedback to disadvantage a particular cloud service) is a difficult problem. Guaranteeing the availability of the trust management service is another significant challenge because of the dynamic nature of cloud environments. In this article, we describe the design and implementation of Cloud Armor, a reputation-based trust management framework that provides a set of functionalities to deliver trust as a service (TaaS), which includes i) a novel protocol to prove the credibility of trust feedbacks and preserve users' privacy, ii) an adaptive and robust credibility model for measuring the credibility of trust feedbacks to protect cloud services from malicious users and to compare the trustworthiness of cloud services, and iii) an availability model to manage the availability of the decentralized implementation of the trust management service. The feasibility and benefits of our approach have been validated by a prototype and experimental studies using a collection of real-world trust feedbacks on cloud services. Index Terms—Cloud computing, trust management, reputation, credibility, credentials, security, privacy, availability

1 INTRODUCTION

THE highly dynamic, distributed, and non-transparent nature of cloud services make the trust management in cloud environments a significant challenge [1], [2], [3], [4]. According to researchers at Berkeley [5], trust and security are ranked one of the top 10 obstacles for the adoption of cloud computing. Indeed, service-level agreements (SLAs) alone are inadequate to establish trust between cloud consumers and providers because of its unclear and inconsistent. Consumers' feedback is a good source to assess the overall Trustworthiness of cloud services. Several researchers' have recognized the significance of trust management and proposed solutions to assess and manage trust based on feedbacks collected from participants [6], [7], [8], [9]. In reality, it is not unusual that a cloud service experiences malicious behaviors (e.g., collusion or Sybil attacks) from its users [6], [10].

This paper focuses on improving trust management in cloud environments by proposing novel ways to ensure the credibility of trust feedbacks. In particular, we distinguish the following key issues

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 5, Issue 1, No 27, 2017 ISSN: 2348-6600
Reference ID: IJCS-290 PAGE NO: 1904-1917

of the trust management in cloud environments: Consumers' privacy. The adoption of cloud computing raise privacy concerns [11]. Consumers can have dynamic interactions with cloud providers, which may involve sensitive information. There are several cases of privacy breaches such as leaks of sensitive information (e.g., date of birth and address) or behavioral information (e.g., with whom the consumer interacted, the kind of cloud services the consumer showed interest, etc.). Undoubtedly, services which involve consumers' data (e.g., interaction histories) should preserve their privacy [12]._ Cloud services protection. It is not unusual that a cloud service experiences attacks from its users. Attackers can disadvantage a cloud service by giving multiple misleading feedbacks (i.e., collusion attacks) or by creating several accounts (i.e., Sybil attacks). Indeed, the detection of such malicious behaviors poses several challenges. First, new users join the cloud environment and old users leave around the clock. This consumer dynamism makes the detection of malicious behaviors (e.g., feedback collusion) a significant challenge. Second, users may have multiple accounts for a particular cloud service, which makes it difficult to detect Sybil attacks [13]. Finally, it is difficult to predict when malicious behaviors occur (i.e., strategic vs. occasional behaviors) Trust management service's (TMS) availability. A trust management service provides an interface between users and cloud services for effective trust management. However, guaranteeing the availability of TMS is a difficult problem due to the unpredictable.

The Number of users and the highly dynamic nature of the cloud environment [6], [7], [10]. Approaches that require understanding of users' interests and capabilities through similarity measurements [15] or operational availability measurements [16] uptime to the total time) are inappropriate in cloud environments.

The TMS should be adaptive and highly scalable to be functional in cloud environments. Implementation of cloud consumers credibility Assessment & trust management of cloud services (Cloud Armor): a framework for reputation-based trust management in cloud environments. In Cloud Armor, trust is delivered as a service (TaaS) where TMS spans several distributed nodes to manage feedbacks in a decentralized way. Cloud Armor exploits techniques to identify credible feedbacks from malicious ones. In a nutshell, the salient features of Cloud Armor are: Zero knowledge credibility proof protocol (ZKC2P). We introduce ZKC2P that not only preserves the consumers 'privacy, but also enables the TMS to prove the credibility of a particular consumer's feedback. We propose that the identity management service (IdM) can help TMS in measuring the credibility of trust feedbacks without breaching consumers' privacy. Anonymization techniques are exploited to protect users from privacy breaches in users' identity or interactions.

A credibility model. The credibility of feedbacks plays an important role in the trust management service's performance. Therefore, we propose several metrics for the feedback collusion detection including the Feedback Density and Occasional Feedback Collusion. These metrics distinguish misleading feedbacks from malicious users. It also has the ability to detect strategic and occasional behaviors of collusion attacks (i.e., attackers who intend to manipulate the trust results by giving multiple trust feedbacks to a certain cloud service in a long or short period of time). In addition, we propose several metrics for the Sybil attacks detection including the multi-identity recognition and occasional sybil attacks. These metrics allow

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 5, Issue 1, No 27, 2017 ISSN: 2348-6600
Reference ID: IJCS-290 PAGE NO: 1904-1917

TMS to identify misleading feedbacks from Sybil attacks. an availability model. High availability is an important requirement to the trust management service.

Thus, we propose to spread several distributed nodes to manage feedbacks given by users in a decentralized way. Load balancing techniques are exploited to share the workload, thereby always maintaining desired availability level. The number of TMS nodes is determined through an operational power metric. Replication techniques are exploited to minimize the impact of inoperable TMS instances. The number of replicas for each node is determined through replication determination metric that we introduce. This metric exploits particle filtering techniques to precisely predict the availability of each node. The remainder of the paper is organized as follows. Section 2 briefly presents the design of Cloud Armor framework. Section 3 introduces the design of the Zero-Knowledge Credibility Proof Protocol, assumptions and attack models. Section 4 and Section 5 describe the details of our credibility model and availability model respectively.

Section 6 reports the implementation of Cloud Armor and the results of experimental evaluations. Finally, Section 7 overviews the related work and Section 8 provides some concluding remarks.

2 THE CLOUDARMOR FRAMEWORK

The Cloud Armor framework is based on the Service Oriented
Architecture (SOA), which delivers trust as a service. SOA and web services are one of the most important enabling technologies for cloud computing in the sense that resources (e.g., infrastructures, platforms, and software) are exposed in clouds as services [17], [18]. In particular, the trust management service spans several distributed nodes that expose interfaces so that users can give their feedbacks or inquire the trust results. Fig. 1 depicts the framework, which consists of three different layers, namely the Cloud Service Provider Layer, the Trust Management Service Layer, and the Cloud Service Consumer Layer.

The cloud service provider layer.

This layerconsistofdifferentcloud service providers who offer one or several cloud services, i.e., Infrastructure as a Service (IaaS), Platforms a Service(PaaS), and Software as a Service (SaaS), publicly on the web (more details about cloud services models and designs can be found in [19]). These cloud services are accessible through web portals and indexed on web search engines such as Google, Yahoo, and Baidu. Interactions for this layer are considered as cloud service interaction with users and TMS, and cloud services advertisements where providers are able to advertise their services on the web. The trust management service layer. This layer consists of several distributed TMS nodes which are hosted in multiple cloud environments in different geographical areas. These TMS nodes expose interfaces so that users can give their feedback or inquire the trust results in a decentralized way. Interactions for this layer include: i) cloud service interaction with cloud service providers, ii) service advertisement to advertise the trust as a service to users through the Internet,iii) cloud service discovery through the Internet to allow users to assess the trust of new cloud services, and iv) Zero knowledge credibility proof protocol interactions enabling TMS to prove the credibility of a particular consumer's feedback.

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 5, Issue 1, No 27, 2017 ISSN: 2348-6600
Reference ID: IJCS-290 PAGE NO: 1904-1917

The cloud service consumer layer. Finally, this layer consists of different users who use cloud services. For example, a new startup that has limited funding can consume cloud services (e.g., hosting their services in Amazon S3).

Interactions for this layer include: i) service discovery where users are able to discover new cloud services and other services through the Internet, ii) trust and service interactions where users are able to give their feedback or retrieve the trust results of a particular cloud service, and iii)registration where users establish their identity through registering their credentials in IdM before using TMS. Our framework also exploits a web crawling approach for automatic cloud services discovery, where cloud services are automatically discovered on the Internet and stored in a cloud services repository. Moreover, our framework contains an identity management service (see Fig. 1) which is responsible for the registration where users register their credentials before using TMS and proving the credibility of a particular consumer's feedback through ZKC2P.

3 ZERO-KNOWLEDGE CREDIBILITY PROOF PROTOCOL

Since there is a strong relation between trust and identification
as emphasized in [20], we propose to use the Identity Management Service to help TMS in measuring the credibility of a consumer's feedback. However, processing the IdM information can breach the privacy of users. One way to preserve privacy is to use cryptographic encryption techniques. However, there is no efficient way to process encrypted data [11]. Another way is to use anonymization techniques to process the IdM information without breaching the privacy of users. Clearly, there is a trade-off between high anonymity and utility. Full anonymization means better privacy, while full utility results in no privacy protection (e.g., using a de-identification anonymization technique still leak sensitive information through linking attacks [21]). Thus, we propose a Zero-Knowledge Credibility Proof Protocol to allow TMS to process IdM's information (i.e., credentials) using the Multi-Identity Recognition factor (see details in Section In other words, TMS will prove the users feedback credibility without knowing the users 'credentials. TMS processes credentials without including the sensitive information. Instead, anonym zed information is used via consistent hashing (e.g., sha-256). The anonymization process covers all the credentials' attributes except the Timestamps attribute.

3.1 Identity Management Service

Since trust and identification are closely related, as highlighted by David and Jaquet in [20], we believe that IdM can facilitate TMS in the detection of Sybil attacks against cloud services without breaching the privacy of users. When users attempt to use TMS for the first time, TMS requires them to register their credentials at the trust identity registry in IdM to establish their identities.

3.2 Trust Management Service

In a typical interaction of the reputation-based TMS, a user either gives feedback regarding the trustworthiness of a particular cloud service or requests the trust assessment of the service. From users' feedback, the trust behavior of a cloud service is actually a collection of invocation history records, represented by a tuple H = (C, S, F, T f),

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 5, Issue 1, No 27, 2017 ISSN: 2348-6600
Reference ID: IJCS-290 PAGE NO: 1904-1917

where C is the user's primary identity, S is the cloud service's identity, and F is a set of Quality of Service (QoS) feedbacks (i.e., the feedback represent several QoS parameters including availability, security, response time, accessibility, price). Each trust feedback in F is represented in numerical form with the range of [0, 1], where 0, 1, and 0.5 means negative, positive, and neutral feedback respectively. T f is the timestamps when the trust feedbacks are given. Whenever a user c requests trust assessment t for cloud service s, TMS calculates the trust result, denoted as T from the collected

3.3 Assumptions and Attack Models

In this paper, we assume that TMS is handled by a trusted third party. We also assume that TMS communications are secure because securing communications is not the focus of this paper. Attacks such as Man-In-The-Middle (MITM) are therefore beyond the scope of this work. We consider the following types of attacks: Collusion attacks. Also known as collusive malicious Feedback behaviors, such attacks occur when several vicious users collaborate together r to give numerous misleading feedbacks to increase the trust result of cloud services (i.e., a self-promoting attack [22]) or to decrease the trust result of cloud services (i.e., a slandering attack This type of malicious behavior can occur in a non-collusive Way where a particular malicious user gives multiple misleading feedbacks to conduct a self-promoting attack or a slandering attack._ Sybil attacks. Such an attack arises when malicious users exploit multiple identities to give numerous misleading feedbacks (e.g., producing a large number of transactions by creating multiple virtual machines for a short period of time to leave fake feedbacks) for a self-promoting or slandering attack. It is interesting to note that attackers can also use multiple identities to disguise their negative historical trust records whitewashing attacks.

4 THE CREDIBILITY MODEL

Our proposed credibility model is designed for i) the Feedback Collusion Detection including the feedback density and occasional feedback collusion, and ii) the Sybil Attacks Detection including the multi-identity recognition and occasional
Sybil attacks.

4.1 Feedback Collusion Detection

4.1.1 Feedback Density

Malicious users may give numerous fake feedbacks to manipulate trust results for cloud services (Self promoting and Slandering attacks). Some researchers suggest that the number of trusted feedbacks can help users to overcome such manipulation where the number of trusted feedbacks gives the evaluator a hint in determining the feedback credibility [25]. However, the number of feedbacks is not in determining the credibility of trust feedbacks. For instance, suppose there are two different cloud services and sy and the aggregated trust feedbacks of both Cloud services are high (i.e., sx has 89 percent positive feedbacks from 150 feedbacks, sy has 92 percent positive feedbacks from 150 feedbacks). Intuitively, users should proceed with the cloud service that has the higher aggregated trust feedbacks (e.g., sy in our case). However, a Self-promoting attack might have been performed on cloud service sy, which means sx should have been selected instead.

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 5, Issue 1, No 27, 2017 ISSN: 2348-6600
Reference ID: IJCS-290 PAGE NO: 1904-1917

4.1.2 Occasional Feedback Collusion

Since collusion attacks against cloud services occur sporadically
[14], we consider time as an important factor in detecting occasional and periodic collusion attacks (i.e., periodicity). In other words, we consider the total number of trust feedbacks jVðsÞj given to cloud service s during a period of time ½t0; t_. A sudden change in the feedback behavior indicates likely an occasional feedback collusion because the change of the number of trust feedbacks given to a cloud service happen abruptly in a short period of time. To detect such behavior, we measure the percentage of occasional change in the total number of feedbacks among the whole feedback behavior . where the first part of the numerator represents the whole area under the curve which represents the feedback behavior for the cloud service of the numerator represents the intersection between the area under the curve and the area under the cumulative mean of the total number of trust feedbacks (i.e., the area . The denominator represents the whole area under the curve.

As a result, the occasional collusion attacks detection is based on measuring the occasional change in the total number of trust feedbacks in a period of time. The higher the occasional change in the total number of trust feedbacks, the more likely that the cloud service has been affected by an occasional collusion attack.

4.2 Sybil Attacks Detection

4.2. 1 Multi-Identity Recognition

Since users have to register their credentials at the Trust Identity Registry, we believe that Multi-Identity Recognition is applicable by comparing the values of users' credential attributes from the identity records I. The main goal of this factor is to protect cloud services from malicious users who use multiple identities (i.e., Sybil attacks) to manipulate the trust results. In a typical Trust Identity Registry, the entire identity records I are represented as a list of m users' primary identities Cp ¼ fp1; p2; . . . ; pmg (e.g., user name) and a list of n credentials' attributes Ca ¼ fa1; a2; . . . ; ang (e.g., passwords, postal address, IP address s, computer name). In other words, the entire Cp _ Ca (Consumer's Primary Identity-Credentials' Attributes) Matrix, denoted as IM,covers all users who registered their credentials in TMS. The credential attribute value for a particular consumer vc;t is stored in TMS without including credentials with sensitive information using the ZKC2P .We argue that TMS can identify patterns in users' anonymous credentials. Malicious users can use similar credentials in different identity records I. Thus, we translate IM to the Multi-Identity Recognition Matrix, denoted as MIRM, which similarly covers the entire identity records I represented as the entire Cp _ Ca matrix.

However, the value for a particular consumer qc;t in the new matrix represents the frequency of the credential attribute value for the same particular consumer vc;t in the same credential attribute (i.e., attribute at). The frequency of a particular credential attribute value vc;t, denoted as qc;t, is calculated as the number of times of appearance (denoted asAp) that the credential value appears in the tth credential attribute normalized by the total number of identity records

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 5, Issue 1, No 27, 2017 ISSN: 2348-6600
Reference ID: IJCS-290 PAGE NO: 1904-1917

4.2.2 Occasional Sybil Attacks

Malicious users may manipulate trust results to disadvantage
particular cloud services by creating multiple accounts and giving misleading feedbacks in a short period of time(i.e., Sybil attacks). To overcome the occasional Sybil attacks, we consider the total number of established identities for users who give feedbacks to cloud service s during a period of time ½t0; t_. The sudden changes in the total number of established identities a possible occasional Sybil attack.
To detect such behavior, we measure the percentage of occasional change in the total number of established identities among the whole identity behavior. all established identities for users who gave feedback to a particular cloud service).

4.3 Feedback Credibility

Based on the proposed credibility metrics, TMS dilutes the influence of those misleading feedbacks by assigning the credibility aggregated weights where r and DðsÞ denote the Feedback Density factor's normalized weight and the factor's value respectively. f and Of ðs; t0; tÞ denote the parameter of
the occasional feedback collusion factor and the factor's value respectively . V denotes the Multi-identity Recognition normalized weight and MidðcÞ denotes the factor's value.
Denotes the occasional Sybil attacks' normalized weight and Oiðs; t0; tÞ denotes the factor's value. _ represents the number of factors used to calculate Crðc; s; t0; tÞ. If only feedback density is considered,_ will be 1. If all credibility factors are considered, _will be 4. All the metrics of the credibility model complement
each other in detecting malicious behaviors and their influence
can be adjusted using the above mentioned parameters.

5 IMPLEMENTATION AND EXPERIMENTAL EVALUATION

In this section, we report the implementation and experimental
results in validating the proposed approach. Our implementation and experiments were developed to validate and study the performance of both the credibility model and the availability model.

5.1 System Implementation

The trust management service's implementation is part of our
large research project, named CloudArmor,2 which offers a
platform for reputation-based trust management of cloud
services [9], [29], [30], [31]. The platform provides an environment. where users can give feedback and request trust assessment for a particular cloud service. Specifically, the trust
management service consists of two main components: the Trust Data Provisioning and the Trust Assessment Function. The trust data provisioning. This component is responsible for collecting cloud services and trust information. We developed the Cloud Services Crawler module based on the Open Source web Crawler for Java (crawler4j3) and extended it to allow the platform to automatically discover cloud services on the Internet. We implemented a set of functionalities to simplify the crawling process and made the crawled data more comprehensive (e.g.,

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 5, Issue 1, No 27, 2017 ISSN: 2348-6600
Reference ID: IJCS-290 PAGE NO: 1904-1917

addSeeds(), selectCrawlingDomain(), add Crawling Time()). In addition, we developed the Trust Feedbacks Collector module to collect feedbacks directly from users in the form of history records and stored them in the Trust Feedbacks Database. Indeed, users typically have to establish their identities for the first time they attempt to use the platform through registering their credentials at the Identity Management Service which stores the credentials in the Trust Identity Registry. Moreover, we developed the Identity Info Collector module to collect the total number of established identities among the whole identity behavior (i.e., all established identities for users who gave feedbacks to a particular cloud service).The trust assessment function. This function is responsible for handling trust assessment requests from users . the trustworthiness of cloud services are compared and the factors of trust feedbacks are calculated (i.e., the credibility factors). We developed the Factors Calculator for attacks detection based on a set of factors (more details on how the credibility factors are calculated can be found in Section 4). Moreover, we developed the Trust Assessor to compare the trustworthiness of cloud services through requesting the aggregated factors weights from the Factors Calculator to weigh feedbacks and then calculate the mean of all feedbacks given to each cloud service. The trust results for each cloud service and the factors' weights for trust feedbacks are stored in the Trust Results and Factors Weights Storage.

5.2 Experimental Evaluation

We particularly focused on validating and studying the robustness of the proposed credibility model against different malicious behaviors, namely collusion and Sybil attacks under several behaviors, as well as the performance of our availability model.

5.3 Credibility Model Experiments

We tested our credibility model using real-world trust feedbacks on cloud services. In particular, we crawled several review websites such as cloud - computing. find the best. com, cloud storage provider reviews.com, and Cloud Hosting Reviewer .com, and where users give their feedbacks on cloud services that they have used. The collected data is represented in a tuple H where the feedback represents several QoS parameters as mentioned earlier in Section 3.2 and augmented with a set of credentials for each corresponding consumer. We managed to collect 10,076 feedbacks given by 6,982 users to 113 real-world cloud services. The collected dataset has been released to the research community via the project website. For experimental purposes, the collected data was divided into six groups of cloud services, three of which were used to validate the credibility model against collusion attacks, and the other three groups were used to validate the model against Sybil attacks where each group consists of 100 users. Each cloud service group was used to represent a different attacking behavior model, namely: Waves, Uniform and Peaks as shown in Fig. 3. The behavior models represent the total number of malicious feedbacks introduced in a particular time instance (e.g., jVðsÞj ¼ 60 malicious feedbacks when T f ¼ 40, Fig. 3a) when experimenting against collusion attacks. The behavior models also represent the total number of identities established by attackers in a period of time (e.g., jIðsÞj ¼ 78 malicious identities when T

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 5, Issue 1, No 27, 2017 ISSN: 2348-6600
Reference ID: IJCS-290 PAGE NO: 1904-1917

i ¼ 20, Fig. 3c) where one malicious feedback is introduced per identity when experimenting against Sybil attacks. In collusion attacks, we simulated malicious feedback to increase trust results of cloud services (i.e., self-promoting attack) while in Sybil attacks we simulated malicious feedback to decrease trust results (i.e., slandering attack). To evaluate the robustness of our credibility model with respect to
malicious behaviors (i.e., collusion and Sybil attacks), we used two experimental settings: I) measuring the robustness
of the credibility model with a conventional model

In our experiments, TMS started rewarding cloud services that had been affected by malicious behaviors when the attacks percentage reached 25 percent (i.e., etðsÞ ¼ 25%), so the rewarding process would occur only when there was a significant damage in the trust result. We conducted 12 experiments where six of which were conducted to evaluate the robustness of our credibility model against collusion attacks and the rest for Sybil
attacks.

5.3.1 Robustness Against Collusion Attacks

For the collusion attacks, we simulated malicious users to
increase trust results of cloud services (i.e., self-promoting
attack) by giving feedback with the range of depicts the analysis of six experiments which were conducted to evaluate the robustness of our model with respect to collusion attacks. the trust result for experimental setting I, while A0, B0, and C0 depict the results for experimental setting II. We note that the closer to 100 the time instance is, the higher the trust results are when when the trust is calculated using the conventional model. This happens because malicious users are giving misleading feedback to increase the trust result for the cloud service. On the other hand, the trust results show nearly no change when calculated using the proposed credibility model (Figs. 4A, 4B, and 4C). Thisdemonstrates that our credibility model is sensitive to collusion attacks and is able to detect such malicious behaviors.
In addition, we can make an interesting observation that our
credibility model gives the best results in precision when
the Uniform behavior model is used (i.e., 0.51, see Fig. 4B0),while the highest recall score is recorded when the Waves behavior model is used (i.e., merely 0.9, see Fig. 4A0). Overall,
recall scores are fairly high when all behavior models are used which indicate that most of the detected attacks are actual attacks. This means that our model can successfully detect collusion attacks (i.e., whether the attack is strategic such as in Waves and Uniform behavior models or occasional such as in the Peaks behavior model) and TMS is able to dilute the increased trust results from self-promoting attacks using the proposed credibility factors.

5.3.2 Robustness Against Sybil Attacks

For the Sybil attacks experiments, we simulated malicious users to decrease trust results of cloud services (i.e., slandering attack) by establishing multiple identities and giving one malicious feedback with the range of [0, 0.2] per identity. Fig. 5 depicts the analysis of six experiments which were conducted to evaluate the robustness of our model with respect to Sybil attacks. In Figs. 5D, 5E, and 5F show the trust results for experimental setting I, while D0, E0, and F0 depict the results

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 5, Issue 1, No 27, 2017 ISSN: 2348-6600
Reference ID: IJCS-290 PAGE NO: 1904-1917

for experimental setting II. From Fig. 5, we can observe that trust results obtained by using the conventional model decrease when the time instance becomes closer to 100. This is because of malicious users who are giving misleading feedback to decrease the trust result for the cloud service. On the other hand, trust results obtained by using our proposed credibility model are higher than the ones obtained by using the conventional model (Figs. 5D, 5E, and 5F). This is because the cloud service was rewarded when the attacks occurred. We also can see some sharp drops in trust results obtained by considering our credibility model where the highest number of drops is recorded when the Peaks behavior model is used (i.e., we can see 5 drops in Fig. 5F which actually matches the drops in the Peaks behavior model in Fig. 3c). This happens because TMS will only reward the affected cloud services if the percentage of attacks during the same period of time has reached the threshold (i.e., which is set to 25 percent in this case). This means that TMS has rewarded the affected cloud service using the change rate of trust results factor. Moreover, from Figs. 5D0, 5E0, and 5F0, we can see that our credibility model gives the best results in precision when the Waves behavior model is used , while the highest recall score is recorded when the Uniform behavior model is used This indicates that our model can successfully detect Sybil attacks, either strategic attacks such as in Waves and Uniform behavior models or occasional attacks such as in the Peaks behavior model) and TMS is able to reward the affected cloud service using the change rate of trust results factor.

5.4 Availability Model Experiments

We tested our availability model using the same dataset we collected to validate the credibility model. However, for the availability experiments, we focused on validating the availability prediction accuracy, trust results caching accuracy, and reallocation performance of the availability model (i.e., to
validate the three proposed algorithms including Particle Filtering based Algorithm, Trust Results & Credibility Weights
Caching Algorithm, and Instances Management Algorithm).

5.4.1 Availability Prediction Accuracy

To measure the prediction accuracy of the availability model,
we simulated 500 nodes hosting TMS instances and set the failure probability for the nodes as 3.5 percent, which complies with the findings in [32]. The motivation of this experiment is to study the estimation accuracy of our approach. We simulated TMS nodes' availability fluctuation and tracked their fluctuation of availability for 100 time steps (each time step counted as an epoch). The actual availability of TMS nodes and corresponding estimated availability using our particle filter approach were collected and compared. Fig. 6a shows the result of one particular TMS node. From the figure, we can see that the estimated availability is very close to the actual availability of the TMS node. This means that our approach works well in tracing and predicting the availability of TMS nodes.

5.4.2 Trust Results Caching Accuracy

To measure the caching accuracy of the availability model, we varied the caching threshold to identify the optimal number of new trust feedbacks that TMS received to recalculate the trust result for a particular cloud service without having a

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 5, Issue 1, No 27, 2017 ISSN: 2348-6600
Reference ID: IJCS-290 PAGE NO: 1904-1917

significant error in the trust results. The trust result caching accuracy is measured by estimating the root-mean square error (RMSE) (denoted caching error) of the estimated trust result and the actual trust result of a particular cloud service. The lower the RMSE value means the higher accuracy in the trust result caching. Fig. 6b shows the trust result caching accuracy of one particular cloud service. From the figure, we can see that the caching error increases almost linearly when the caching threshold increases. The results allow us to choose the optimal caching threshold based on an acceptable caching error rate. For example, if 10 percent is an acceptable error margin, the caching threshold can be set to 50 feedbacks. It is worth mentioning that
the caching error was measured on real users' feedbacks on
real-world cloud services.

5.4.3 Reallocation Performance

To validate the reallocation performance of the availability model, we used two experimental settings: I) comparing the number of TMS nodes when using the reallocation of trust feedbacks and without reallocation while increasing the number of feedbacks (i.e., when the workload threshold ewðstmsÞ ¼ 25%); II) comparing the number of TMS nodes when using the reallocation of trust feedbacks and without reallocation while varying ewðstms). The lower the number of TMS nodes, the more cost efficient TMS is. Fig. 7a shows the results of experimental settings I. We can observe that the total number of TMS nodes when using the reallocation of trust feedbacks technique is fairly low and more stable than the total number of TMS nodes when reallocation is not used (i.e., even when the total number of feedbacks is high). Fig. 7b shows the results of experimental settings II. From the figure, we can see that the higher the workload threshold the lower the number of TMS nodes. However, the number of TMS nodes when using the reallocation of trust feedbacks technique is lower than the number of TMS
nodes when reallocation is not considered. This means that our approach has advantages in minimizing the ban

6 RELATED WORK

Over the past few years, trust management has been a hot topic in the area of cloud computing Some of the research efforts use policy-based trust management techniques. For example, Ko et al. [34] propose Trust Cloud framework for accountability and trust in cloud computing. In particular, Trust Cloud consists of five layers including workflow, data, system, policies and laws, and regulations layers to address accountability in the cloud environment. All of these layers maintain the cloud accountability life cycle which consists of seven phases including policy planning, sense and trace, logging, safe-keeping of logs, reporting and replaying, auditing, and optimizing and rectifying. Brandic et al. [7] propose a novel approach for compliance management in cloud environments to establish trust between different parties. The approach is developed using a centralized architecture and uses compliant management technique to establish trust between cloud service users and cloud service providers. Unlike previous works that use policy-based trust management techniques, we assess the trustworthiness of a cloud service using reputation-based trust management techniques. Reputation represents a high influence that cloud service users have over the trust management system [35], especially that the opinions of the various cloud service users can dramatically influence the

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 5, Issue 1, No 27, 2017 ISSN: 2348-6600
Reference ID: IJCS-290 PAGE NO: 1904-1917

reputation of a cloud service either positively or negatively. Some research efforts also consider the reputation-based trust management techniques. For instance, Habib et al. [6] propose a multi-faceted trust management (TM) system architecture for cloud computing to help the cloud service users to identify trustworthy cloud service providers. In particular, the architecture models uncertainty of trust information collected from multiple sources using a set of quality of service attributes such as security, latency, availability, and customer support. The architecture combines two different trust management techniques including reputation and recommendation where operators (e.g., AND, OR, and FUSION) are used. Hwang and Li [4] propose a security aware cloud architecture that assesses the trust for both cloud service providers and cloud service users. To assess the trustworthiness of cloud service providers, the authors propose the trust negotiation approach and the data coloring (integration) using fuzzy logic techniques. To assess the trustworthiness of cloud service users, they develop 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 do not consider the problem of unpredictable reputation attacks against cloud services, we present a credibility model that not only detects the misleading trust feedbacks from collusion and Sybil attacks, but also has the ability to adaptively adjust the trust results for cloud services that have been affected by malicious behaviors.

7 CONCLUSION

Given the highly dynamic, distributed, and non-transparent nature of cloud services, managing and establishing trust between cloud service users and cloud services remains a significant challenge. Cloud service users' feedback is a good source to assess the overall trustworthiness of cloud services. However, malicious users may collaborate together to i) disadvantage a cloud service by giving multiple misleading trust feedbacks (i.e., collusion attacks) or ii) trick users into trusting cloud services that are not trustworthy by creating several accounts and giving misleading trust feedbacks (i.e., Sybil attacks). In this paper, we have presented novel techniques that help in detecting reputationbased attacks and allowing users to effectively identify trustworthy cloud services. In particular, we introduce a credibility model that not only identifies misleading trust feedbacks from collusion attacks but also detects Sybil attacks no matter these attacks take place in a long or short period of time (i.e., strategic or occasional attacks respectively).

We also develop an availability model that maintains the trust management service at a desired level. We have collected a large number of consumer's trust feedbacks given on real-world cloud services (i.e., over 10,000 records) to evaluate our proposed techniques. The experimental results demonstrate the applicability of our approach and show the capability of detecting such malicious behaviors. There are a few directions for our future work. We plan to combine different trust management techniques such as reputation and recommendation to increase the trust results accuracy. Performance optimization of the trust management service is another focus of our future research work.

REFERENCES

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 5, Issue 1, No 27, 2017 ISSN: 2348-6600
Reference ID: IJCS-290 PAGE NO: 1904-1917

[1] S. M. Khan and K. W. Hamlen, "Hatman: Intra-cloud trust management
for Hadoop," in Proc. 5th Int. Conf. Cloud Comput., 2012, pp. 494–501.

[2] S. Pearson, "Privacy, security and trust in cloud computing," in Privacy and Security for Cloud Computing, ser. Computer Communications and Networks. New York, NY, USA: Springer, 2013, pp. 3–42.

[3] J. Huang and D. M. Nicol, "Trust mechanisms for cloud computing," J. Cloud Comput., vol. 2, no. 1, pp. 1–14, 2013.

[4] K. Hwang and D. Li, "Trusted cloud computing with secure resources and data coloring," IEEE Internet Comput., vol. 14, no. 5, pp. 14–22, Sep./Oct. 2010.

[5] M. Armbrust, A. Fox, R. Griffith, A. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A view of cloud computing," Commun. ACM, vol. 53, no. 4, pp. 50–58, 2010.

[6] S. Habib, S. Ries, and M. Muhlhauser, "Towards a trust management
system for cloud computing," in Proc. 10th Int. Conf. Trust, Security Privacy Comput. Commun., 2011, pp. 933–939.

[7] I. Brandic, S. Dustdar, T. Anstett, D. Schumm, F. Leymann, and R.
Konrad, "Compliant cloud computing (C3): Architecture and language
support for user-driven compliance management in clouds," in Proc. 3rd Int. Conf. Cloud Comput., 2010, pp. 244–251.

[8] W. Conner, A. Iyengar, T. Mikalsen, I. Rouvellou, and K. Nahrstedt, "A trust management framework for service-oriented environments," in Proc. 18th Int. Conf. World Wide Web, 2009, pp. 891–900.

[9] T. H. Noor, Q. Z. Sheng, and A. Alfazi, "Reputation attacks detection for effective trust assessment of cloud services," in Proc. [10] T. H. Noor, Q. Z. Sheng, S. Zeadally, and J. Yu, "Trust management of services in cloud environments: Obstacles and solutions,"ACM Comput. Surv., vol. 46, no. 1, pp. 12:1–12:30, 2013.

[11] S. Pearson and A. Benameur, "Privacy, security and trust issues arising from cloud computing," in Proc. 2nd Int. Conf. Cloud Comput., 2010, pp. 693–702.

[12] E. Bertino, F. Paci, R. Ferrini, and N. Shang, "Privacy-preserving digital identity management for cloud computing," IEEE Data Eng. Bull, vol. 32, no. 1, pp. 21–27, Mar. 2009. [13] E. Friedman, P. Resnick, and R. Sami, "Manipulation-resistant reputation systems," in Algorithmic Game Theory. New York, USA: Cambridge Univ. Press, 2007, pp. 677–697.

[14] K. Ren, C. Wang, and Q. Wang, "Security challenges for the public cloud," IEEE Internet Comput., vol. 16, no. 1, pp. 69–73, Jan./Feb. 2012.

[15] F. Skopik, D. Schall, and S. Dustdar, "Start trusting strangers? bootstrapping and prediction of trust," in Proc. 10th Int. Conf. Web Inf. Syst. Eng., 2009, pp. 275–289.

[16] H. Guo, J. Huai, Y. Li, and T. Deng, "KAF: Kalman filter based adaptive maintenance for

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 5, Issue 1, No 27, 2017 ISSN: 2348-6600
Reference ID: IJCS-290 PAGE NO: 1904-1917

dependability of composite services," in Proc. 20th Int. Conf. Adv. Inf. Syst. Eng., 2008, pp. 328–342.

[17] T. Dillon, C. Wu, and E. Chang, "Cloud computing: Issues and challenges," in Proc. IEEE 24th Int. Conf. Adv. Inf. Netw. Appl.,2010, pp. 27–33.

[18] Y. Wei and M. B. Blake, "Service-oriented computing and cloud computing: Challenges and opportunities," IEEE Internet Comput., vol. 14, no. 6, pp. 72–75, Nov./Dec. 2010.

[19] P. Mell and T. Grance. (2011, Sep.). The NIST definition of cloud computing [Online]. Available: http://csrc.nist.gov/ publications/drafts/800-145/Draft-SP-800-145_cloud-definition. Pdf

[20] O. David and C. Jaquet. (2009, Jun.). Trust and identification in the light of virtual persons pp. 1–103 [Online]. Available: http:// www.fidis.net/resources/deliverables/identity-of-identity/

[21] B. Fung, K. Wang, R. Chen, and P. Yu, "Privacy-preserving data publishing: A survey of recent developments," ACM Comput. Surv., vol. 42, no. 4, pp. 1–53, 2010.

[22] J. R. Douceur, "The sybil attack," in Proc. Revised Papers 1st Int. Workshop Peer-to-Peer Syst., 2002, pp. 251–260. [23] S. Ba and P. Pavlou, "Evidence of the effect of trust building technology in electronic markets: Price premiums and buyer behavior," MIS Quart., vol. 26, no. 3, pp. 243–268, 2002.