International Journal of Applied Sciences, Engineering and Management
ISSN 2320 – 3439, Vol. 04, No. 01, January 2015, pp. 20 – 24

Cloud Armor: A Trusty Supporting Reputation-based Management for Cloud Services

Kommineni Madhavi [1], M. Vijay Kumar [2]
[1]Student, Dept of CSE, Amrita Sai Institute of Science & Technology
[2]Assistant Professor, Dept of CSE, Amrita Sai Institute of Science & Technology
Email: kommineni.madhavi12@gmail.com, vijaymallarapu@gmail.com

Abstract: Trust management is a standout amongst the most difficult issue for the tackling and development of cloud computing. The highly dynamic, distributed, and non-transparent nature of cloud services leads to many challenging issues such as privacy, security, and availability. Saving consumers' privacy is not an easy task due to the confidential information involved in the interactions between customers and the trust management service. Protecting cloud services against their malicious clients (e.g., such clients may give misleading feedback to inconvenience a specific cloud service) is a complicated issue. Due to the dynamic nature of cloud environments, assuring the availability of the trust management service is a challenging issue. In this article, we describe the design and implementation of Cloud Armor, a reputation-based trust management system that gives an arrangement of functionalities to deliver Trust as a Service (TaaS), which includes i) a novel convention to demonstrate the credibility of trust inputs and save clients' security, ii) a versatile and robust credibility model for measuring the credibility of trust feedbacks to keep cloud services from malicious clients and to analyze the dependability of cloud services, and iii) an availability model to deal with the accessibility of the decentralized usage of the trust management service. The achievability and advantages of our methodology have been tried by a model and test studies utilizing a collection of true trust feedbacks on cloud services.
Keywords: *Cloud computing, trust management, reputation, credibility, security, privacy, availability.*

1. Introduction

Consumers' feedback is an excellent source to assess the overall trustworthiness of cloud services. Several researchers have known the significance of trust management and proposed solutions to assess and manage trust based on feedbacks collected from participants. In reality, it is not unusual that a cloud service experiences malicious behaviors e.g., collusion or Sybil attacks from its users. This paper focuses on improving trust management in cloud environments by presenting novel ways to ensure the credibility of trust feedbacks. In particular, 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. In particular, we distinguish the following key issues of the trust management in cloud environments**.** The adoption of cloud computing raises privacy concerns. Customers 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. It is not unusual that a cloud service experiences attacks from its users. Attackers can disadvantage a cloud service by giving multiple misleading feedbacks or by creating several accounts. Indeed, the detection of such malicious behaviors' poses several challenges. Firstly, new users join the cloud environment and old users leave around the clock. This consumer dynamism makes the detection of malicious behaviors a significant challenge. Secondly, users may contain multiple accounts for a particular cloud service, which makes it difficult to detect Sybil attacks. Finally, it is difficult to guess when malicious behaviors occur.

2. Literature Survey

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 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. 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, especially that the opinions of the various cloud service users can dramatically influence the reputation of a cloud service either positively or negatively

Layers: 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. 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 layer consists of different cloud service providers who offer one or several cloud services, i.e., IaaS (Infrastructure as a Service), PaaS (Platform as a Service), and SaaS (Software as a Service), publicly on the Web (more details about cloud services models and designs can be founding. 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 [from fig. 1].

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 (ZKC2P) interactions enabling TMS to prove the credibility of a particular consumer's feedback [from fig. 1].

ZERO-KNOWLEDGE CREDIBILITY PROOF PROTOCOL (ZKC2P)

Since there is a strong relation between trust and identification as emphasized in, we propose to use the *Identity Management Service* (IdM) 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. 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.

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 TM Sand proving the credibility of a particular consumer's feedback through ZKC2P. 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. In other words, TMS will prove the users' feedback credibility without knowing the users' credentials.

Attack Models:

Collusion Attacks: Also known as collusive malicious feedback behaviors, such attacks occur when several vicious users collaborate together to give numerous misleading feedbacks to increase the trust result of cloud services (i.e., a self-promoting attack) 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 (i.e., whitewashing attacks).

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.

Feedback Collusion Detection*:* Feedback Density Malicious users may give numerous fake feedbacks to manipulate trust results for cloud services (i.e., *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. However, the number of feedbacks is not enough in determining the credibility of trust feedbacks.

3. Proposed Approach

3.1 Implementation:

User Registration: Users are the important entities in our scheme. Initially the user wants to register their credentials in the corresponding system. These credentials are including some personal information about users like, name, date of birth, address, contact number etc... This personal information is stored in Identity Management Services. This acts like a database in this manner. After this registration only, the user can use all the services provide from cloud. But this information is stored very securely. We need to protect the user's privacy from unauthorized activities.

Upload Services: Cloud service provider is responsible for provide useful services to the user. Their services are classified into three categories. These are, Infrastructure As a service, Platform As a Service, Software As a Service. Under these three categories, the CSP upload the services for users. These details are stored in Identity management Service.

Send Feedback: After uploading the services, the user can use these services from the cloud. To use these services, the user needs to store their credentials in IDM services. Then the user can share their opinion to the cloud regarding to the services. This feedback also stored in the identity management Services. This IDM service sore the user details according to their feedback service.

Feedback Collusion Detection: Trust Management Service is the one, which use all the details stored in the IDM for check the user's credibility. Users have a limit to send the feedback for a service. There is a threshold value for that. If they cross the limit, we can identify if they are trying to increase/decrease the service rate. Suppose they cross the limit, the trust management service separate them from the users list. This process is called as feedback collusion detection.

Sybil Attack Detection: Some users are very brilliant. Because they know, if we cross the limit, we would catch. So they use the different accounts for increase/decrease the service rate. In our system, their credentials also stored in identity management service, this record are viewed by trust management service. Our TMS service cross checking the user's credentials. Some credential of user's are cannot to change like date of birth, mobile number, mail id. Using that similarity, the TMS can found the unauthorized users. This is called as Sybil attack detection.

3.2 INPUT DESIGN

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple.

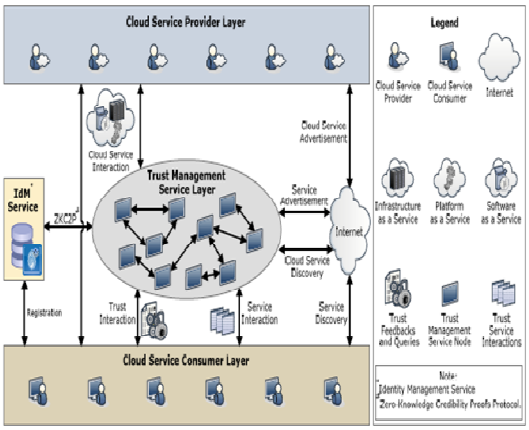
3.3 OBJECTIVES

Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities. When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow.

3.4 OUTPUT DESIGN:

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements..Select methods for presenting information..Create document, report, or other

formats that contain information produced by the system. The output form of an information system should accomplish one or more of the following objectives. Convey information about past activities, current status or projections of the Future. Signal important events, opportunities, problems, or



warnings. Trigger an action. Confirm an action.

3.5 SYSTEM ARCHITECTURE

Figure 1: Architecture of the Cloud Armor Trust Management Framework

1. The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.
2. The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.
3. DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.
4. DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail.

4. Conclusion

Given the highly dynamic, distributed, and nontransparent nature of cloud services, managing and creating trust between cloud service users and cloud services remains a large challenge. Cloud service users' feedback is a good source to levy the overall trustworthiness of cloud services. though, malicious users may rally 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 sending misleading trust feedbacks (i.e., Sybil attacks). In this paper, we have presented novel techniques that help in detecting reputation based attacks and allowing users to effectively to find trustworthy cloud services. In finicky, we establish 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 unruffled a large number of customer's trust feedbacks given on real-world cloud services (i.e., over 10,000 records) to evaluate our proposed techniques. The experimental results express the applicability of our approach and show the capability of finding such malicious behaviors.

5. References

[1] S. M. Khan and K. W. Hamlen, "Hatman: Intra-Cloud Trust Management for Hadoop," in Proc. CLOUD'12, 2012.

[2] S. Pearson, "Privacy, Security and Trust in Cloud Computing," in Privacy and Security for Cloud Computing, ser. Computer Communications and Networks, 2013, pp. 3–42.

[3] J. Huang and D. M. Nicol, "Trust Mechanisms for Cloud Computing," Journal of Cloud Computing, 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 Computing, vol. 14, no. 5, pp. 14–22, 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," Communications of the 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. of TrustCom'11, 2011.

[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. of CLOUD'10, 2010.

[8] W. Conner, A. Iyengar, T. Mikalsen, I. Rouvellou, and K. Nahrstedt, "A Trust Management Framework for Service-Oriented Environments," in Proc. of WWW'09, 2009.

[9] T. H. Noor, Q. Z. Sheng, and A. Alfazi, "Reputation Attacks Detection for Effective Trust Assessment of Cloud Services," in Proc. of TrustCom'13, 2013.

[10] T. H. Noor, Q. Z. Sheng, S. Zeadally, and J. Yu, "Trust Management of Services in Cloud Environments: Obstacles and Solutions," ACM

Computing Surveys, 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. CloudCom’10, 2010.
[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, 2009.

About the Authors

Ms. K. Madhavi is a student of Amrita Sai Institute of Science And Technology, paritala, Krishna Dt, Andhra Pradesh, presently she is pursuing M. Tech (C.S.E) and her areas of interest are Cloud Computing and Data Structures.

Mr. M Vijay Kumar is working as an Assistant Professor in Department of Computer Science and Engineering of Amrita Sai Institute of Science And Technology. Having 7 years of teaching experience and areas of interests are Network Security and Computer Networks.