A Novel Approach for Handling Security in Cloud Computing Services

Sahar Mohammed Abduljalil
Student
Department of Information Systems, Faculty of Computers and Information.
Cairo University, Cairo, Egypt.

Osman Hegazy
Professor
Department of Information Systems, Faculty of Computers and Information.
Cairo University, Cairo, Egypt.

Ehab E. Hassanein
Associate Professor.
Department of Information Systems, Faculty of Computers and Information.
Cairo University, Cairo, Egypt.

ABSTRACT
The rapid advances in cloud computing has introduced a variety of security issues; each requires certain skills and knowledge. While developing business services requires conducting analysis and design for the business activities that does not include common security functions for these services. Currently there is no framework existing from the logical level for securely orchestrating of cloud services in a heterogeneous environment. We are addressing in the paper a clear separation of concerns between the "business logic" and the "security logic" in order for any service implementing the proposed security service to be considered a high level secured service in terms of access and communication in order for it to be widely used and acceptable. A development model is proposed to write secured services without burdening the developer of continuously rewriting security routines.

General Terms
Cloud Computing, Security, Cloud services, Identity Access Management, Security Assertion Markup Language, Java authentication authorization Service.

1. INTRODUCTION
Cloud computing is set of resources and services offered through the Internet. Cloud services are delivered from data centers located throughout the world. Clearly, security is a serious concern in such a big environment. Cloud services can be implemented using different approaches and technologies which need to be secured at the different stages of the request / response cycle between clients (relying parties such as users or applications) and service providers (companies or divisions within a company exposing those services). Several security layers are defined between clients and services providers. The first security layer, also known as "perimeter security" or "first line of defense," is referred to as the demilitarized zone or DMZ. The second security layer, or "green zone" to continue with the military analogy, is located behind the inner firewall of the DMZ. In some cases, the green zone may include several security sub-layers designed to further filter access to web services. Finally, the last security layer which we are focusing at, or "last-mile security," is provided by agents co-located with the web services or applications to be protected.

Cloud computing technology has revolutionized the IT industry like never before, and is seen to be offering the most promising future for the computing world. Cloud technology is now in the stage of unleashing its tremendous capabilities and most of its flavours are being explored around the world by IT companies, big and small alike. Cloud Computing has almost everything to offer to its customers, whether it is software as pay per use (Software as a Service model), or development platforms and tolls being offered and accessed through a web browser (Platform as a Service model) or provides customers with highly scalable and on demand computing resources (Infrastructure as a Service) [1]. The following section explores some of the major security issues that cloud computing faces today: [2, 3, 4, 5, 6, 7].

Table 1. Security Issues

Security Issue	Explanation
Data Security	Encryption, fine grained authorization.
Network Security	All data flow over the network needs to be secured in order to prevent leakage of sensitive information. Traditional network security issues: Man in the middle IP spoofing Port scanning Packet sniffing Encryption techniques such as: Secure Socket layer [SSL] Transport Layer Security [TLS]
Data locality	Due to compliance and data privacy laws in various countries, location of data is of utmost importance in many enterprise architecture
Data integrity	-Data integrity is easily achieved in a standalone system with a single database. - Data integrity in such a system is maintained via database constraints and transactions. Transactions should follow ACID (atomicity, consistency, isolation and durability) properties to ensure data integrity. -In distributed environment, there should be

	central global transaction manager. -Can be achieved by 2 phase commit protocol.
Data Segregation	As a result of multi-tenancy multiple users can store their data using the applications provided by SaaS. In such a situation, data of *various users* will reside at the *same location*, so Intrusion of data of one user by another becomes possible in this environment. This intrusion can be done either by ***hacking through the loop holes*** in the application or by ***injecting client code*** into the SaaS system. A client can write a masked code and inject into the application. If the application executes this code without verification, then there is a high potential of intrusion into other's data. - handcrafting parameters that bypass security checks and access sensitive data of other tenants. A SaaS model should therefore ensure a clear boundary for each user's data. The boundary must be ensured not only at the physical level but also at the application level. The service should be intelligent enough to segregate the data from different users. [3]
Data Access	Data access issue is mainly related to security policies provided to the users while accessing the data. The SaaS model must be flexible enough to incorporate the specific policies put forward by the organization (e.g policies based on access rights of employees). The model must also be able to provide organizational boundary within the cloud because multiple organization will be deploying their business processes within a single cloud environment.
Data Breaches	Breaching into cloud environment where various users and business organizations lie together.
Web application security	Solutions such as network firewall, intrusion detection systems and prevention systems do not adequately address security challenges in SaaS applications, because when we talk about SaaS security we are not only concerned with introducing security risks at the network level, but application level as well.
Virtualization Vulnerability	Major tasks in virtualizations are: -Ensuring isolation of different instances running on the same physical machine [Current VMMs (Virtual Machine Monitor) do not offer perfect isolation]. -Controlling host and guest operating system by the administrator. -VMM should be root secured, i.e no privilege within the guest virtualized environment permits interface with the host system. Vulnerabilities have been found in virtualization softwares which can be exploited by malicious where they bypass certain security restrictions and gain privileges. -Example: -*Microsoft Vulnerability*: A guest OS can run a code on the host or another guest OS. *Xen Vulnerability*: Input validation error where root users of the guest domain can execute commands via special crafted entries when guest system is booted.
Availability	SaaS applications need to provide it's service around the clock and this will be ensured by 2 ways: -Making some architectural changes at the application and infrastructural level for scalability and availability. -Adopting a multitier architecture supported by a load balancer farm. -Resilience to hardware and software failures and denial of service attacks. -Considering an appropriate action plan for business continuity and disaster recovery. -Mitigation techniques for distributed denial of service. -Automatically locking user accounts after successive incorrect credentials, but incorrect configuration and implementation of some features can be used by malicious users and do denial of service.
Backup	Sensitive enterprise data should be backed up for recovery in case of disasters. Using strong encryption techniques to protect backup data. -Data at rest stored in *S3* in *Amazon* are not encrypted by default.
Identity Management and sign on process.	The pure identity paradigm. The user access (log-on) paradigm: The service paradigm. Models for Identity management and sign on services: -Independent IdM stack: All data (user account, passwords) is maintained with the SaaS vendor. -Credential Sychronization: Users do not need to remember multiple passwords. -Federated IdM : Users do not need to remember multiple passwords . -No separate integration with enterprise directory. -Low security risk value as compared to credential synch

Lack of Trust management and privacy in cloud: As it is seen in the cloud environment do not have adequate trust and privacy management facilities established well in place to mitigate the fear of cloud users in moving their critical IT business and data to cloud. Consumers and cloud service providers are forced to trust among themselves without much knowledge about the vendor's availability, back up, job service efficiency, security controls and so on. On the other

hand, the service provider has to trust the customers' data assuming that the user is a legitimate one without any malicious intent. Any malicious user can put the service providers' name and reputation at stake.

2. MOTIVATION FOR THE WORK:

The problem of duplication of security logic across multiple cloud services gets even more aggravated when a user decides to use multiple cloud service providers, so his data get duplicated and stored across multiple cloud vendors. [8] So every cloud service, the customer needs to exchange his/her authentication information with each cloud service provider. These redundant actions can introduce vulnerability. This is a security concern because identification and credentials information are used to uniquely identify a user which can help in targeting attacks against this specific user. This in turn can be used to infringe on the privacy of the customers which have greater significance. So our model has been designed to offer a comprehensive and single point of reliance for all the security needs. In other words, gathering security functionality as Security Management service to allow them to be located and used as needed by more than one service, and allow security to be adapted without having to change the service logic itself. Both the cloud service users and providers can avail the services as per demand through an account created with our proposed security management model. This problem is clarified in figure 1.

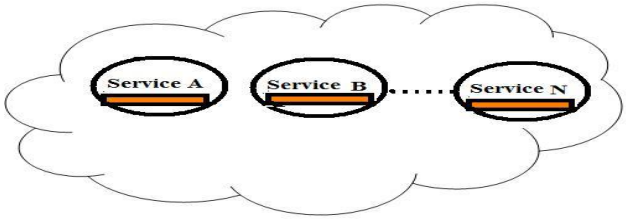

Figure 1: Security logic code is repeated in each service.

Publishes service security policy so that clients can discover dynamically what credentials and mechanisms are needed to establish trust with the service.

The rest of the paper is organized as follows. In section 2, various related approaches and works in this direction have been looked at, and in section 3 a detailed explanation of the proposed architecture and working of the model have been explained. In section 4, the relative merits of the scheme have been mentioned. Finally, implementation and evaluation in section 5 followed by conclusion and future work.

3. RELATED WORK

Table 2. Analysis of scholars work.

Paper Name	Problem focused at	Limitation
Cloud Computing Security Management [9]	It focused on one data security aspect which is Identity and access management (IAM) in the cloud by discussing Security Assertion Markup Language (SAML) and OAuth protocol. Their main concern was to discuss some of the security IAM protocols used to protect cloud users and to conclude which of these protocols will be best for organizations which are moving in the direction of consuming the cloud Services. They concluded that it is very difficult to choose one protocol that will be better than another, where it is totally dependent on the organizational behavior toward their business goals	The work did not address accountability and auditing issues in the cloud, nor does it focus on trust and privacy.
CTES based Secure approach for Authenticatin and Authorization of Resource and Service in Clouds. [10]	-Addresses the associated concerns through an authentication and authorization model for a cloud computing paradigm. -Improvement over traditional Kerberos protocol to authenticate the users and to access the services and resources in cloud.	The scholars have not discussed about the implementation in a public cloud with heterogeneous users and providers.
Trust as a Service: A Framework for Trust Management in Cloud Environments [11].	This paper has focused on trust management on cloud computing. They have introduced an adaptive credible model that distinguish between credible trust feedback and malicious feedback by considering cloud service consumers' capability and majority consensus of their feedback.	Cloud Service providers feedback about the cloud consumers have not been considered.
A Secured Cost-effective Multi-Cloud Storage in Cloud Computing.	The model proposed distributes the data pieces of a user among more than one service providers, in such a way that no one of the SPs can	No effective data distribution techniques.

[12]	retrieve any meaningful information from the pieces of data stored on its servers. So it aims to remove centralization in storing data in the cloud. In case there was an intrusion to a network of one of the providers, they won't get any meaning full information from the pieces of stored data.	
TrustCloud: A Framework for Accountability and Trust in Cloud Computing. [13]	The author bring out the urgent need for research in cloud accountability and the various challenges in achieving a trusted cloud. They further discuss the policy based and technical approaches that can be used for establishing an accountable cloud. Their work focuses on a technique called provenance logging.	No implementation for the technique.

4. PROPOSED Model FOR CLLOUD SECURITY MANAGEMENT

4.1 Overview

Secure operation requires that applications [14] and services be capable of supporting a variety of security functionality, such as authentication, authorization, credential conversion, auditing, and delegation. Interaction between services requires having a range of security requirements and mechanisms. These mechanisms and requirements are likely to evolve over time as new mechanisms are developed or policies change. According to the needs of the cloud users, they can appropriately choose the solutions available for their identity and access requirements, trust and privacy needs. Similarly, cloud providers can register to the proposed model and ensure their security requirements. The following section explains in details how the model works and renders the appropriate service to both the cloud service consumer and provider.

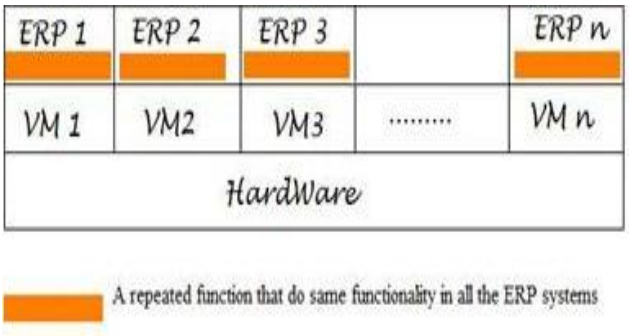

Figure 2: Before proposed model was applied.

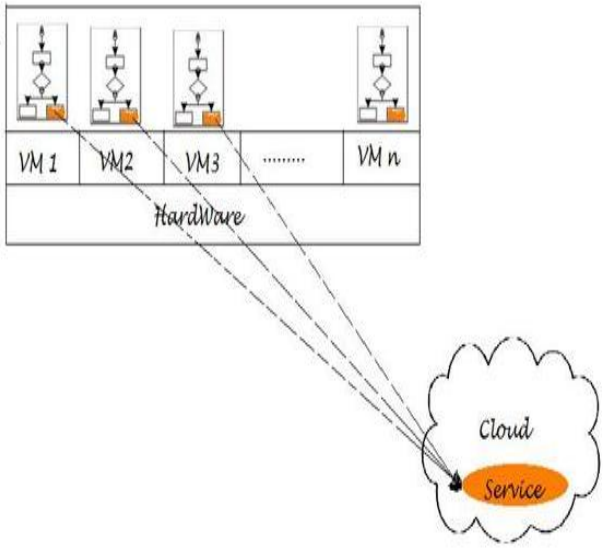

Figure 3: This Figure shows that security logic which was duplicated in all ERP systems is gathered to one service.

4.2 Architecture and working of the proposed model

The working of the cloud security manager model involves four phases: Enrolment phase, Credential processing service phase, authorization service phase using Java Authentication Authorization Service (JAAS), and finally Service rendering phase.

-Enrolment phase: Users need to enrol themselves with the Cloud Security Service, this is the cloud service consumer, and cloud service provider, but the enrolment procedure for both are totally different with respect to the data collected. For example a cloud service provider that provides a cloud storage service is generally priced on two factors: how much data is to be stored on the cloud service provider's server, and how long this data will be stored for some period of time [12]. To sum up, each service provider is associated with a service level of agreement that a service consumer adheres to. This service level agreement in the example is the cost of providing storage service per unit of stored data. If the user is a service provider, then the SAAS services provided by this provider need to be registered in the service registry this helps mediate

the brokering process, as well as applying human readable naming conventions such as ID or URI to easily identify a particular service. While for the cloud service user, login credentials data are collected, and a passphrase is needed for every user, moreover, all this login data will be stored encrypted by a generated one time public and private key, using RSA algorithm.

-Credential Processing Service: This service validates the details of processing and validates authenticated tokens. Different authentication mechanisms can be used classified as cryptography or biometric based, but as for the implementation part I have used cryptography.

-Authorization Service using JAAS: A service that evaluates policy rules regarding the decision to allow the attempted actions based on information about the requestor (identity, attributes, etc.), the target or service accessed (identity, policy, attributes, etc.), and details of the request. The Java Authentication and Authorization Service (JAAS) is a set of application program interfaces (APIs) that can determine the identity of a user or computer attempting to run Java code and ensure that the entity has the right to execute the functions requested. In this context, authentication is the process of determining whether or not an entity is who or what it declares itself to be; authorization is the process of giving an entity permission to do, use, or obtain something according to credential based mechanism which use trustworthy information being held by a subject.

-Service rendering: The service rendering phase explains how the cloud users and CPSs are significantly leveraging the benefits of the Security Management Service for cloud security. The registered users can avail the service by entering the passphrase they have chosen during the enrolment phase.

The high level architecture of the proposed scenario is clarified in figure 4, as far as services need to be added {S1, S2, S3, ...Sn} , those services need to implement the Security Abstract Service which act like a controller, this service has two registries, one for the services registered associated with its providers, and the other for the consumers invoking and binding the service. The Security Abstract Service uses a Security Manager Service to provide the security functionality.

5. IMPLEMENTATION

Experimental evaluation has been done on Eclipse Juno

Plan A: Requesting AES symmetric Service

Plan B: Requesting RSA Asymmetric Service

Plan C: Requesting MD5 hashing algorithm

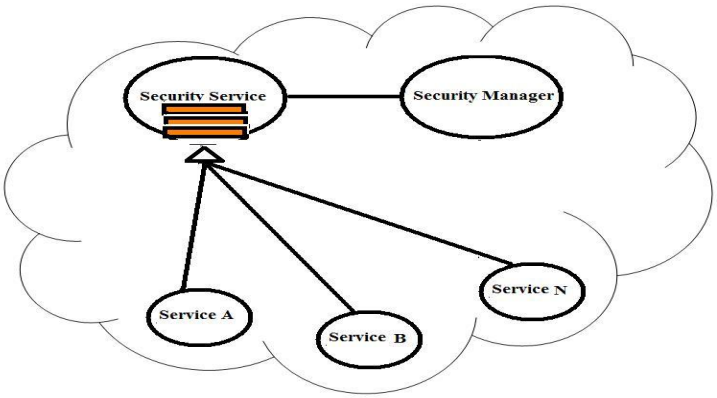

Figure 4: Proposed model.

We assumed the following scenario that a service consumer wants to store the data

1. So the system will first check if this user is an authenticated one.
2. Next what security services are authorized to be given to this service consumer.
3. A query will be sent to the database to retrieve authorization information.
4. Let's assume that this user is authorized to encrypt data before storing it, moreover to do digital signature step, in order to validate message integrity and ensure that the sender has sent the message, and this message have not been tampered at by an attacker at the middle.
5. Save encrypted message.

Where on the three plans authentication and authorization using JAAS is called, however encryption algorithm is varying in each case, and processing time was noted, in order for a comparison to be done on which encrypting algorithm will take the minimum time in developed in cloud platform.

6. MERITS OF THE PROPOSED MODEL

-Multiple security mechanisms can be added and updated in the security management service directly.

-Dynamic creation of services. Users must be able to create new services (e.g., "resources") dynamically without administrator intervention. These services must be coordinated and must interact securely with other services. Thus, we must be able to name the service with an assertable identity and to grant rights to that identity without contradicting the governing local policy.

7. CONCLUSION:

Since many of the major hurdles for adopting cloud computing are related to security concerns, security firms need to provide their software on a more agile development cycle and build their software to natively work with multitenant systems with reducing slow processing in the

cloud, so it is unreasonable and inefficient to require that security software be installed in every virtual machine because that bogs down the cloud services' servers.

In the paper, we have proposed a model, where security workload is totally separated from the service logic. Our research found that it is possible to separate security functions from business activities, encapsulate them into services, a reusable secured service, and a business service, then combine or merge them to achieve secured services without burdening the developer from coding the security code over and over again, so the developer of the service will concentrate on the business logic of the service itself not the security issues. To sum up, one time security service is done, but used several times with many services independent from each other. So proposed model is satisfying and targeting developers who are building cloud-based services where they can implement directly the security logic in order to get a secured cloud services. However, as the number of users using the security service increase, accessing this centralized service can experience delays.

8. REFERENCES

[1] Deepa Krishnan, M. C. (2012). Cloud Security Management Suite- Security as a Service. *IEEE* .

[2] Mathisen, E. (2011). Security Challenges and Solutions in Cloud Computing. *IEEE* .

[3] Subashini, S., & Kavitha, V. (2011). A survey on security issues in service delivery models of cloud computing. *Journal of Network and Computer Applications* , 1-11.

[4] Rohit Bhadauria, R. C. (2011). A Survey on Security Issues in Cloud Computing. *IEEE* .

[5] Qaisar, S., & Khawaja, K. F. (2012). Cloud Computing: Network/Security Threats And Countermeasures. *Interdisciplinary Journal Od Contemporary Research in Business.*

[6] Bernstein, D., & Vij, D. (2010). Intercloud Security Considerations. *IEEE International Conference on Cloud Computing Technology and Science* .

[7] F. Lombardi, D. P. (2010). Transparent security for cloud. *ACM Symposium on Applied Computing Sierre* , 414--415.

[8] Ayesha Malik, M. M. (2012). Security Framework for Cloud Computing Environment: A Review. *Journal of Emerging Trends in Computing and Information Sciences* .

[9] Sameera Abdulrahman Almulla, C. Y. (2010). Cloud Computing Security Management. *International Conference on the Engineering Systems Management and its application* .

[10] Sanjeev Kumar Pippal, A. K. (2011). CTES based Secure approach for Authentication and Authorization of Resource and Service in Clouds. *International Conference on Computer & Communication Technology* .

[11] Talal H. Noor, Q. Z. (2011). Trust as a Service: A Framework for Trust Management in Cloud Environments. *WISE'11 Proceedings , ACM* , 314–321.

[12] Yashaswi Singh, F. K. (2011). A Secured Cost-effective Multi-Cloud Storage in Cloud Computing. *IEEE INFOCOM* .

[13] Ryan K L Ko, P. J. (2011). TrustCloud: A Framework for Accountability and Trust in Cloud Computing. *IEEE World Congress on Services* .

[14] Jean Bacon, D. E. (2010). Enforcing End-to-end Application Security in the Cloud. *International Conference on Middleware*, (pp. 293-312).