ISSN:2348-2079

Volume-6 Issue-2

International Journal of Intellectual Advancements and Research in Engineering Computations

Providing Data Security & Integrity in Cloud using CCAF

[1]Mr. C.Mani, M.C.A., M.Phil., M.E., Associate Professor,
[2]Mr. G.Gowri Shankar,FinalMCA.,
Department of MCA,Nandha Engineering College(Autonomous), Erode-52.
E-Mail ID: cmanimca@gmail.com, sankarsank555@gmail.com

***Abstract*-This project mainly focus on providing data security to the cloud users. Security is essential factor in the cloud services. For this purpose, we need a clear framework which provides security to both cloud users and cloud providers. The Cloud Computing Adoption Framework (CCAF) is used for adopting and applying cloud security principles systematically. This framework has key features includes identification, data integrity, privacy and durability. The CCAF has three layers of security such as firewall and access control, identity management and intrusion prevention and convergent encryption. The Security Manager stores metadata which includes block signatures, encrypted key and process identity management check. In the convergent encryption layer, the data undergoes a security test which uses the hash of plain text to work out the encryption key. These three layers employs on providing security to the cloud data.**

***Keywords*- CCAF, Cloud Computing,CE,DI,Security framework.**

I. INTRODUCTION

This current challenges facing cloud community on cloud security is enormous. Therefore, we need a clear framework, which provides an integrated approach to study cloud service performances before the implementation, the one that supports clear implementation of cloud security attributes at the implementation level, and the one that can be adopted by both cloud users and cloud providers.

The CCAF is a comprehensive model for adopting and applying cloud security principles systematically. The outcome of each activity is shown inside the parenthesis. These best practice techniques will keep grow as the framework has been in various applications. Their approach is heavily focused on the use of XML to transfer and interpret data through their security mechanism. Framework is an appropriate method provided with careful and clear explanations. Security is the most important attribute for any system. Providing secure experience is one of the key principles in the process of gaining customer confidence for a system. Now days, almost all the websites are asking to store user's personal information in servers to understand the customer and serve better. It's the responsibility of an organization to confirm that customer's data is safe and accessed in a secured manner. In any

Technology stack, there are number of different approaches to develop a system. Choosing an approach depends on various factors such as budget, resource, timeline and expertise in the cloud technology. In addition to those factors, data security is unavoidable factor in a system design and every developer should have enough knowledge on possible security attacks on the system and follow the best practices in development to protect the application from those attacks.

SQL Injection is one of the most common attacks that can be triggered against any application that talks to database. Cross Site Scripting is one of the most common application level attacks that hackers use to sneak into web applications today, and one of the most dangerous. It is an attack on the privacy of clients of a particular web site which can lead to a total breach of security when customer details are stolen or manipulated. Unfortunately, as outlined in this paper, this is often done without the knowledge of either the client or the organization being attacked. In order to prevent this malicious vulnerability, it is critical that an organization implement both an online and offline security strategy.

This includes using an automated application vulnerability assessment tool, like App Scan from Sanctum, which can test for all the common web vulnerabilities, and application specific vulnerabilities (like cross site scripting) on a

site. And for a full online defense, installing an application firewall, like AppShield from Sanctum, that can detect and defend against any type of manipulation to the code and content sitting on and behind the web servers. In this project proposed system core of a traditional CSS attack lies a vulnerable script in the vulnerable site. This script reads part of the HTTP request (usually the parameters, but sometimes also HTTP headers or path) and echoes it back to the response page, in full or in part, without first sanitizing it i.e. making sure it doesn't contain Javascript code and/or HTML tags

II. RELATED WORKS

A. A DYNAMIC SECURE GROUP SHARING FRAMEWORK IN PUBLIC CLOUD COMPUTING

With the popularity of group data sharing in public cloud computing, the privacy and security of group sharing data have become two major issues. The cloud provider cannot be treated as a trusted third party because of its semi-trust nature, and thus the traditional security models cannot be straightforwardly generalized into cloud based group sharing frameworks. In this paper, we propose a novel secure group sharing framework for public cloud, which can effectively take advantage of the Cloud Servers' help but have no sensitive data being exposed to attackers and the cloud provider. The framework combines proxy signature, enhanced *TGDH* and proxy re-encryption together into a protocol. By applying the proxy signature technique, the group leader can effectively grant the privilege of group management to one or more chosen group members. The enhanced *TGDH* scheme enables the group to negotiate and update the group key pairs with the help of Cloud Servers, which does not require all of the group members been online all the time. By adopting proxy re-encryption, most computationally intensive operations can be delegated to Cloud Servers without disclosing any private information. Extensive security and performance analysis shows that our proposed scheme is highly efficient and satisfies the security requirements for public cloud based secure group sharing.

B.ADDRESSING CLOUD COMPUTING SECURITY ISSUES

The recent emergence of cloud computing has drastically altered everyone's perception of infrastructure architectures, software delivery and development models. Projecting as an evolutionary step, following the transition from mainframe computers to client/server deployment models, cloud computing encompasses elements from grid computing, utility computing and autonomic computing, into an innovative deployment architecture. This rapid transition towards the clouds, has fuelled concerns on a critical issue for the success of information systems, communication and information security. From a security perspective, a number of unchartered risks and challenges have been introduced from this relocation to the clouds, deteriorating much of the effectiveness of traditional protection mechanisms. As a result the aim of this paper is twofold; firstly to evaluate cloud security by identifying unique security requirements and secondly to attempt to present a viable solution that eliminates these potential threats. This paper proposes introducing a Trusted Third Party, tasked with assuring specific security characteristics within a cloud environment. The proposed solution calls upon cryptography, specifically Public Key Infrastructure operating in concert with SSO and LDAP, to ensure the authentication, integrity and confidentiality of involved data and communications. The solution, presents a horizontal level of service, available to all implicated entities, that realizes a security mesh, within which essential trust is maintained.

B. CLOUD STORAGE AND BIOINFORMATICS IN A PRIVATE CLOUD DEPLOYMENT

This paper describes service portability for a private cloud deployment, including a detailed case study about Cloud Storage and bioinformatics services developed as part of the Cloud Computing Adoption Framework (CCAF). Our Cloud Storage design and deployment is based on Storage Area Network (SAN) technologies, details of which include functionalities, technical implementation, architecture and user support. Experiments for data services (backup automation, data recovery and data migration) are performed and results confirm backup automation is completed swiftly and is reliable for data-intensive research. The data recovery result confirms that execution time is in proportion to quantity of recovered data, but the failure rate increases in an exponential manner. The data migration result confirms execution time is in proportion to disk volume of migrated data, but again the failure rate increases in an exponential manner. In addition, benefits of CCAF are illustrated using several bioinformatics examples such as tumor modeling, brain imaging, insulin molecules and simulations for medical training. Our Cloud Storage solution described here offers cost reduction, time-saving and user friendliness.

D.TOWARD SECURE AND DEPENDABLE STORAGE SERVICES IN CLOUD COMPUTING

The Cloud storage enables users to remotely store their data and enjoy the on-demand high quality cloud applications without the burden of local hardware and software management. Though the benefits are clear, such a service is also relinquishing users' physical possession of their outsourced data, which inevitably poses new security risks toward the correctness of the data in cloud. In order to address this new problem and further achieve a secure and dependable cloud storage service, we propose in this paper a flexible distributed storage integrity auditing mechanism, utilizing the homomorphic token and distributed erasure-coded data. The proposed design allows users to audit the cloud storage with very lightweight communication and computation cost. The auditing result not only ensures strong cloud storage correctness guarantee, but also simultaneously achieves fast data error localization, i.e., the identification of misbehaving server. Considering the cloud data are dynamic in nature, the proposed design further supports secure and efficient dynamic operations on outsourced data, including block modification, deletion, and append. Analysis shows the proposed scheme is highly efficient and resilient against Byzantine failure, malicious data modification attack, and even server colluding attacks.

III. METHODOLOGY

A. NETWORK SECURITY

In this module it describes the intrusion protection used in cloud computing adaptation framework to ensure that all data is safeguarded all the times. The Intrusion Prevention System is used with the core syntax includes:
Syntax:

- crypto key pubkey-chain rsa` named-key realm-cisco.pub signature key-string.

Example:

An encrypted key-string is generated to protect the data from potential malicious hack. The key-string may look like this: B199ABCB D34ED0F9 085FADC1 359C189E F30AF10A C0EFB624 7E0764BF 3E53053E.

C. ADOPTIVE ENCRYPTION SCHEME

In this module, the process is implemented with the users, cloud computing adaptation server and the security options. The Users can encrypt each key from each block and own key. They can split files into blocks, encrypt them with the key, followed by signing the resulting encrypted blocks and creating the storage request. For each file, this key will be used to decrypt and rebuild the original file during the retrieval phase.

The user also uses single sign-on to access each block with a compact signature scheme. Cloud Server roles are offered by the server. First, it can authenticate users during the storage/retrieval phase. Second, it can access control. Third, it can encrypt or decrypt data between users and their cloud. The data can be further encrypted to prevent dictionary attacks before being forwarded to the metadata manager (MM). Each user has a pair of keys (pk,sk) which is used in the asymmetric encryption algorithm, and pk needs to be negotiated with the group manager on the condition that no Certificate Authorities and security channels are involved in. The key is the private key of the user and is used for data sharing in the scheme.

Blocks are decrypted and the server verifies the signature of each block with the user's public key during the retrieval phase. The security module stores metadata which include block signatures, encrypted keys and process identity management check. While SM checks and verifies the right identity, the security proceeds to convergent encryption, which serves as the third layer of security. After the identity management phase, all data has to undergo the security test offered by Convergent encryption (CE), which uses the hash of plaintext to work out the encryption key (K)

D. PREVENT SCRIPTING ATTACK

This module covers Non-persistent types of XSS checking. In non-persistent type, if any URL is provided by the end user to the server, the query string is parsed such that it contains any java script and then it is eliminated. The new URL with new query string is displayed to the user. The user if send that URL then it is processed. Likewise, if the user during uploading the content sends any html tag data then it is also found out and warned.

In this module, the user uploads a page and our coding will check the contents is containing such that if it contains navigation to other unwanted sites and that will lead to download an executable, malware or spyware. In this module, cross site scripting is major problem in website attack. In order to test this attack, all the html page and posts are checked to find the attack and prevent the website from malware.

D.*PREVENTION OF SQL INJECTION ATTACK*

In this module, the end user enters the data in the text box. The web page after clicking the submit button, parses the words into tokens. Then it

checks the syntaxes of the submitted query such that it contains comment characters. If it so then it is removed. Then Non equal number of single quotes is checked. If it contains odd number of single quotes, then it rejects the input.

Then the words are checked such that it matches with column names in table presented in 'table' of the given query. If it contains any column name then the query is rejected. It also matched the stored procedures names in the database and if found, query is rejected. 5

After elimination of unwanted contents such as field names or stored procedure names / comments, the query is recreated and displayed to the user. If the user submits the new displayed query, then it is executed as it will not harm the database.

F. SQL INJECTION ATTACK SCENARIO

In this module, the end user enters the data in the text box. The web page after clicking the submit button, parses the words into tokens. Then it checks the syntaxes of the submitted query such that it contains comment characters. If it so then it is removed. Then Non equal number of single quotes is checked. If it contains odd number of single quotes, then it rejects the input.

Then the words are checked such that it matches with column names in table presented in 'table' of the given query. If it contains any column name then the query is rejected. It also matched the stored procedures names in the database and if found, query is rejected. In the previous after the elimination of unwanted contents such as field names or stored procedure names / comments, the query is recreated and displayed to the user. If the user submits the new displayed query, then it is executed as it will not harm the database.

G. GUOJUN WANG, QIN LIU AND JIE WU

Cloud computing, as an emerging computing paradigm, enables users to remotely store their data into a cloud so as to enjoy scalable services on-demand. Especially for small and medium-sized enterprises with limited budgets, they can achieve cost savings and productivity enhancements by using cloud-based services to manage projects, to make collaborations, and the like. However, allowing cloud service providers (CSPs), which are not in the same trusted domains as enterprise users, to take care of confidential data, may raise potential security and privacy issues. To keep the sensitive user data confidential against un trusted CSPs, a natural way is to apply cryptographic approaches, by disclosing decryption keys only to authorized users.

However, when enterprise users outsource confidential data for sharing on cloud servers, the adopted encryption system should not only supp ort fine-grained access control, but also provide high performance, full delegation, and scalability, so as to best serve the needs of accessing data anytime and anywhere, delegating within enterprises, and achieving a dynamic set of users. In this paper [16], we propose a scheme to help enterprises to efficiently share confidential data on cloud servers. We achieve this goal by first combining the hierarchical identity-based encryption (HIBE) system and the cipher ext-policy attribute-based encryption (CP-ABE) system, and then making a performance-expressivity tradeoff, finally applying proxy re-encryption and lazy re-encryption to our scheme.

IV. CONCLUSION AND FUTURE WORKS

In this project analysis a periodic scan service, essentially simulating an attack from their server to a client's in order to check if the attack is successful. If the attack succeeds, the client receives detailed information on how it was performed and thus has a chance to fix the issues before the same attack is attempted by someone else. For sites that require complete mitigation of XSS vulnerabilities, assessment techniques like manual code review are necessary. Additionally, if Javascript is executing on the page, the seal could be overwritten with a static copy of the seal (so, in theory, such a service alone is likely not sufficient to eliminate XSS risk completely).

The analysis fragment part of the location/URL object, in which case the server does not receive the payload at all, because the browser typically does not send this part of the URL.

To improve the referrer object, in which case the server receives the payload in the Refer header.

REFERENCES

[1] K. Xue and P. Hong, "A dynamic secure group sharing framework in public cloud computing," IEEE Transactions on Cloud Computing, vol. 2, no. 4, pp. 459–470, 2014.

[2] D. Zissis and D. Lekkas, "Addressing cloud computing security issues," Future Generation Computer Systems, vol. 28, no. 3, pp. 583–592, 2012.

[3] V. Chang, R. J. Walters, and G. Wills, Cloud Storage and Bioinformatics in a Private Cloud Deployment: Lessons for Data Intensive Research. New York, NY, USA: Springer CLOSER 2012, CCIS 367, pp. 245–264, 2013.

[4] C. Wang, Q. Wang, K. Ren, N. Cao, and W. Lou, "Toward secure and dependable storage services in cloud computing," IEEE Transactions on Services Computing, vol. 5, no. 2, pp. 220–232, 2012.