Grenze International Journal of Engineering and Technology, Special issue

Multi-Factor Protection Mechanism for Secure Data Sharing in Cloud

C.Gobinath[1], M.Aarthi[2] , K.Babyshalini[3] and R.Rengashree[4]
[1]Assistant Professor, Department of Information Technology, Kongunadu College of Engineering and Technology, Trichy, Tamilnadu.
Email: gobinath0119@gmail.com
[2-4]Final year, Department of Information Technology, Kongunadu College of Engineering and Technology, Trichy, Tamilnadu.
Email: {aarthibby123, shreeravindran98, babyshalinik123}@gmail.com

***Abstract*—Cloud computing is a shared pool of configurable computer structure resources and high level services can be rapidly provisioned with minimal supervision effort, often over the internet in everywhere at any time. In cloud storage, there should be lot of sanctuary problems to access the data so we propose the multi-factor authentication for sheltered data access in cloud such as attribute based authentication, certificate generation, key generation. In two-factor protection method, the cryptographic key is protected by two factors. They are Attribute based encryption and security device based authentication. In security device based authentication, secret code is kept in user's system as well as in device like pen drive. If gadget is lost or stolen, we couldn't access the data from the cloud. The complications are how the old security device is revoked and how the new security device can do the decryption properly. To avoid the security problem in cloud, we provide multi factor authentication such as certificate generation between machine to machine, attribute based authentication and key generation for secure data sharing in cloud. The purpose of certificate generation is used to verify that user sending a message is who he/she claims to be and to provide the receiver with the means to encode a reply.**

***Index Terms*— Attribute based authentication, certificate exchange, key generation, secure encryption.**

I. INTRODUCTION

Cloud computing is definitely a promising model for business computing. It's describes important infrastructure to have an up-and coming type of service provision which includes the benefit of dropping expense by sharing computing and storage sources. Currently, Cloud Computing is really a huge knowledge that is exceeding all of the earlier technologies of computing of this competitive and demanding Information technology industry.
Cloud computing is consistently growing and there are many main cloud computing providers including Amazon, Google, Microsoft, Yahoo and many others who are offering solutions including Software-as-a-Service(SaaS), Platform-as-a-Service(PaaS), Storage-as-a- Service and Infrastructure-as-a-Service(IaaS). In addition, considering the possibility to substantially minimizing costs by optimization and also maximizing operating as we l as economic helpfulness, cloud computing is an excellent technology. Furthermore, cloud

Grenze ID: 01.GIJET.5.1.

computing can tremendously advance its cooperation, speed, and also range, thus empowering a totally worldwide computing model on the internet infrastructure. On top of that, the cloud computing has recompense in delivering bonus scalable, fault tolerant services.

Cloud computing handle resource supervision in a better way since the user no longer needs to be accountable for identifying wherewithal for storage. If a user wants to store more data they request it from the cloud provider and once they are finished they can either release the storage by simply stopping the use of it, or move the data to a long-term lower-cost storage resource. This further allows the user to effectively use more dynamic resources because they no longer need to concern themselves with storage and cost that accompany new and old resources.

The subsequent cloud computing categories have been notorious and defined in the progression of cloud development:

A. Infrastructure as Service (IaaS)

It provides virtual machines and other inattentive hardware and operating systems which may be controlled through a service Application Programming Interface (API). IaaS includes the entire infrastructure resource stack from the facilities to the hardware platforms that reside in them. It incorporates the capability to abstract resources as well as deliver physical and logical connectivity to those resources. IaaS provides a set of APIs which allow executive and other forms of interaction with the infrastructure by consumers.

B. Platform as a Service (PaaS)

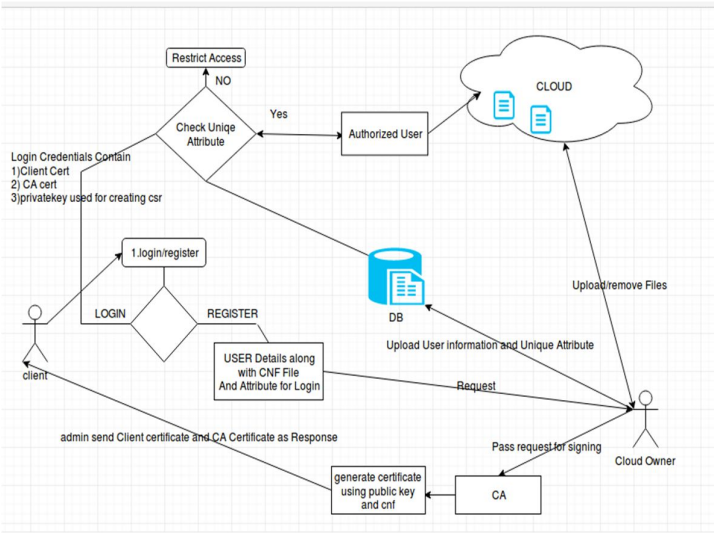
It allows customers to develop new applications using APIs, implemented and operated remotely. The platforms offered include development tools, configuration management and deployment platforms. PaaS is positioned over IaaS and adds an additional layer of integration with application development frameworks and gathering such as database, messaging and queuing that allow developers to put up applications for the platform with programming languages and tools are supported by the stack.

C. Software as a Service (SaaS)

Is software offered by a third party provider, available on demand, usually through a Web browser, operating in a remote manner. Examples include online word processing and spreadsheet tools, CRM services and Web content delivery services. SaaS in turn is built upon the underlying IaaS and PaaS stacks and provides a self-contained operating environment used to deliver the intact user experience plus the content, its presentation, the applications and executive capabilities.

II. System Design

To avoid the security problem in cloud, we provide multi factor authentication such as certificate generation between machine to machine, attribute based authentication and key generation for secure data contribution in cloud. The purpose of certificate generation is used to substantiate that user sending a message is who he/she claims to be and to endow with the receiver with the means to encode a counter.



III. SYSTEM TESTING

Software testing is a method of assessing the functionality of a software program. There are many diverse types of software testing but the two focal categories are dynamic testing and static testing. Dynamic testing is an assessment that is conducted while the program is executed; static testing, on the other hand, is an examination of the program's code and coupled documentation. Dynamic and static methods are often used together.
Testing is a set activity that can be deliberate and conducted systematically. Testing begins at the module level and work towards the amalgamation of entire computers based system. Nothing is absolute without testing, as it is vital success of the scheme

A. Testing Objectives

There are several convention that can serve as testing objectives, they are

1. Testing is a process of executing a program with the intent of finding an error
2. A good test case is one that has high probability of finding an undiscovered error.
3. A successful test is one that uncovers an undiscovered error.

If testing is conducted lucratively according to the objectives as stated above, it would uncover errors in the software. Also testing demonstrates that software functions materialize to the working according to the specification, that performance requirements appear to have been met.
There are three ways to test a program

1. For Correctness
2. For Implementation efficiency
3. For Computational Complexity.

Tests for correctness are made-up to verify that a program does unerringly what it was designed to do. This is greatly more thorny than it may at first appear, especially for large programs.
Tests used for implementation efficiency attempt to find ways to make a correct program faster or use less storage. It is a code-refining process, which reexamines the implementation phase of algorithm development. Tests for computational complexity amount to an experimental analysis of the complexity of an algorithm or an experimental comparison of two or more algorithms, which solve the same problem.
The data is entered in all forms separately and whenever an error occurred, it is corrected immediately. A quality team deputed by the management verified all the necessary documents and tested the Software while entering the data at all levels. The development process involves various types of testing. Each test type addresses a specific testing requirement. The most common types of testing involved in the development process are:

- Unit Test.
- System Test
- Integration Test
- Functional Test

B. Unit Testing

The first test in the progress process is the unit test. The source code is normally divided into modules, which in turn are divided into smaller units called units. These units have specific behavior. The test done on these units of code is called unit test. Unit test depends upon the language on which the project is developed. Unit tests ensure that each unique path of the project performs accurately to the documented specifications and contains clearly defined inputs and expected results.

C. Functional Testing

Functional test can be defined as testing two or more modules together with the intent of finding defects, demonstrating that defects are not present, verifying that the module performs its intended functions as stated in the specification and establishing confidence that a program does what it is supposed to do.

D. Integration Testing

In integration testing modules are combined and tested as a group. Modules are typically code modules, individual applications, source and destination applications on a network, etc. Integration Testing follows unit testing and precedes system testing. Testing after the product is code complete. Betas are often widely distributed or even distributed to the public at large in hopes that they will buy the final product when it is released.

E. White Box Testing

Testing based on an analysis of internal workings and structure of a piece of software. This testing can be done sing the percentage value of load and energy. The tester should know what exactly is done in the internal program. It includes techniques such as Branch Testing and Path Testing. White box testing also called as Structural Testing or Glass Box Testing.

F. Black Box Testing

In block box testing exclusive of knowledge of the internal workings of the item being tested. Tests are usually functional. This testing can be done by the user who has no familiarity of how the shortest path is found.

IV. MODULES

A. Login module

In this module, user can login to their webpage through registration process. It should provide authentication by conveyance key to specific user using key generation based mechanism.

B. Certificate Generation

In this module the certificate is exchange between cloud owner and user by CA (i.e.,In server side CA can generate certificate based on the user request using public key and cnf)

C. Secure file access

User can access the files in locked manner by using login credentials such as client certificate, CA certificate and user private key.

V. CONCLUSION

In this paper we anticipated a multifactor protection mechanism for cloud storage. The multifactor is realized by separating security key in two ways. One is authenticated through key generation and other is done by attribute based authentication. Furthermore with the help of certificate substitute we obtained secure access control of data from cloud.

REFERENCES

[1] J. K. Liu and D. S. Wong, “Solutions to key exposure problem in ring signature.” IJ Network Security, vol. 6, no. 2, pp. 170–180, 2008

[2] “Attribute-based encryption supporting direct/indirect revocation modes,” in Cryptography and Coding. Springer, 2009, pp. 278–300.

[3] V. Goyal, O. Pandey, A. Sahai, and B. Waters, “Attribute-based encryption for fine-grained access control of encrypted data,” in Proceedings of the 13th ACM conference on Computer and communications security.

[4] J. K. Liu, M. H. Au, X. Huang, R. Lu, and J. Li, “Fine-grained two factor access control for web-based cloud computing services,” IEEE Transactions on Information Forensics and Security, vol. 11, no. 3, pp. 484–497, 2016.

[5] C. Zuo, J. Shao, G. Wei, M. Xie, and M. Ji, “Chosen ciphertext secure attribute-based encryption with outsourced decryption,” in Australasian Conference on Information Security and Privacy. Springer, 2016, pp. 495–508.

[6] M. Pirretti, P. Traynor, P. McDaniel, and B. Waters, “Secure attribute- based systems,” in Proceedings of the 13th ACM conference on Computer and communications security. ACM, 2006, pp. 99–112.

[7] F. Zhang, Q. Li, and H. Xiong, “Efficient revocable key-policy attribute Based encryption with full security,” in Computational Intelligence and Security (CIS), 2012 Eighth International Conference on. IEEE, 2012,pp. 477–481.