ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 9, September 2014

A Novel Approach to Provide Dynamic Authentication & Data Integrity in Public Cloud Environment: Using MD5, RSA and Enhanced OTP

Priyanka Nema

Master of Engineering (CSE), Shri Ram Institute of Technology, Jabalpur, Madhya Pradesh, India

ABSTRACT*:* The cloud computing stage gives populace the chance for sharing information resources and services along with the people through internet. In private cloud structure, data and information is shared amongst the individuals who are in that cloud only. In this paper we have projected new advanced security design and architecture for cloud computing platform. This makes certain secure communication system and hiding information from others. In this model message digest based file encryption system and secure public-key encryption system using RSA for exchanging data is included. This model also includes onetime password (OTP) system for user authentication process. This structure can be easily applied with all cloud computing layers, e.g. PaaS, SaaS and IaaS. Our work primarily deals with the cloud security system and authentication of use in whole cloud computing environment.

KEYWORDS*:* Cloud computing; Security architecture; AES; MD5 Hashing; RSA; One-time Password (OTP).

I. INTRODUCTION

At the present world of networking system, Cloud computing [1] is one the most important and developing concept for both the developers and the users. Persons who are interrelated with the networking environment, cloud computing is a preferable platform for them. Therefore in recent days providing security has become a major challenging issue in cloud computing.

In the cloud environment, resources are shared among all of the servers, users and individuals. As a result files or data stored in the cloud become open to all. Therefore, data or files of an individual can be handled by all other users of the cloud. [2, 3] Thus the data or files become more vulnerable to attack. As a result it is very easy for an intruder to access, misuse and destroy the original form of data. An intruder can also interrupt the communication. Besides, cloud service providers provide different types of applications which are of very critical nature. Hence, it is extremely essential for the cloud to be secure [4]. Another problem with the cloud system is that an individual may not have control over the place where the data needed to be stored. A cloud user has to use the resource allocation and scheduling, provided by the cloud service provider. Thus, it is also necessary to protect the data or files in the midst of unsecured processing. In order to solve this problem we need to apply security in cloud computing platforms. In our proposed security model we have tried to take into account the various security breaches as much as possible.

Besides, hardware encryption is helpful only for the database system, not for other security issues. Authenticated user detection technique is currently very important thing. But, this technique is rarely discussed in the recently used models for ensuring security in cloud computing. In this paper we have proposed new security architecture for cloud computing platform. In this model high ranked security algorithms are used for giving secured communication process. Here files are encrypted with AES algorithm in which keys are generated randomly by the system.

The RSA algorithm is used for secured communication between the users and the servers. This paper is formatted in the following way: - section II describes related work of this paper work, section III describes proposed architecture and its working steps, section IV describes the experimental environment, results in different aspects and advantages of the propose model, and section V describes the future aspect

II.RELATED WORK

Numerous research on security in cloud computing has already been proposed and done in recent times. Identification based cloud computing security model have been worked out by different researchers [12]. But only identifying the actual user does not all the time prevent data hacking or data intruding in the database of cloud environment. Yao's Garbled Circuit is used for secure data saving in cloud servers [13, 14].. AES based file encryption system is used in some of these models [15, 16]. But these models keep both the encryption key and encrypted file in one database server. Only one successful malicious attack in the server may open the whole information files to the hacker, which is not desirable.

Some other models and secured architectures are proposed for ensuring security in cloud computing environment [17, 18]. Although these models ensures secured communication between users and servers, but they do not encrypt the loaded information. For best security ensuring process, the uploaded information needs to be encrypted so that none can know about the information and its location. Recently some other secured models for cloud computing environment are also being researched [19, 20]. But, these models also fail to ensure all criteria of cloud computing security issues [21]

III.PROPOSED MODEL

At present ensuring security in cloud computing platform has become one of the most significant concerns for the researchers. We have undertaken these problems in our research, to provide some solution correlated with security. We have proposed the following security model for cloud computing data storage shown in Figure 1. In this model, all the users irrespective of new or existing member, needs to pass through a secured channel which is connected to the main system computer. System server computer has relation with other data storage system. system. RSA algorithm using the system's public key is used for the encryption. Whenever the user requests for a file the system sends it by encrypting it via RSA encryption algorithm

The password is used to keep the user account secure and secret from the unauthorized user. But the user defined password can be compromised. To overcome this difficulty one time password is used in the proposed security model. Thus whenever a user login in the system, he/she will be provided with a new password for using it in the next login. This is usually provided by the system itself. This password will be generated randomly. Each time a new password is created for a user, the previous password for that user will be erased from the system. New password will be updated for that particular user. A single password will be used for login only once. The password will be sent to the users authorized mail account. Therefore at a same time a check to determine the validity of the user is also performed. As a result only authorized user with a valid mail account will be able to connect to the cloud system. By this system, existence of unauthorized user or a user with an invalid mail account will be pointed out. The newly generated password is restored in the system after md5 hashing. The main purpose of MD5 hashing is that this method is a one way system and unbreakable. Therefore it will be difficult for an unauthorized or unknown party for retrieving the password for a selected user even if gained access to the system database. After connecting with the system a user can upload or download the file(s). For the first time when connected with the system the user can only upload file(s). After that users can both upload and download their files. When a file is uploaded by an user the system server encrypts the file using AES encryption algorithm.

In the proposed security model 128 bit key is used for AES encryption. 192 bit or 256 bit can also be used for this purpose. Here the 128 bit key is generated randomly by the system server. A single key is used only once. That particular key is used for encrypting and decrypting a file of a user for that instance. This key is not further used in any instance later. The key is kept in the database table of the system server along with the user account name. Before inserting the user account name it is also hashed using md5 hashing. This insures that unauthorized person cannot retrieve the key to decrypt a particular file for a particular user by simply gaining access and observing the database table of the system server. As a result the key for a particular file becomes hidden and safe. Again when the encrypted file is uploaded for storing to the storage server, the path of the encrypted file along with user account is kept and maintained in the database table on the storage server. Here user name is used for synchronization between the database tables of main system server and the storage server. The encrypted files on the storage server are inserted not serially. We have developed a hash table for determining where to insert a file into the database table. The algorithm for generating the hash table is described later in this section.

Login into the main system is compulsory when a user wants to download a previously stored file. When the user selects a file to download, the system automatically retrieves the key for the requested file from the main system server.

The system matches user account name saved in its database table with that saved in the storage server after hashing it using md5 hashing. The path of the encrypted file from the storage server is found by using the user account name and the hash table input for the requested file. In this model, the encryption key for a particular file of a particular user is only known to the main system server.

At last, we propose hardware encryption for making the databases fully secured from the attackers and other unauthorized persons. Figure 2 is the Pictorial representation of the proposed cloud security architecture. Here, single user and server represent n users and n servers.An algorithm is developed, which is used for inserting the file in the main server (System), and in the database table where the encrypted file is kept. This is saturated from the system server for the cloud computing platform. . In file saving server, the file is inserted in a random order which becomes the output of the algorithm. The relations between the system server table and database server tables can be thought as disjoint sets. The pseudo code of the algorithm used is described in table.

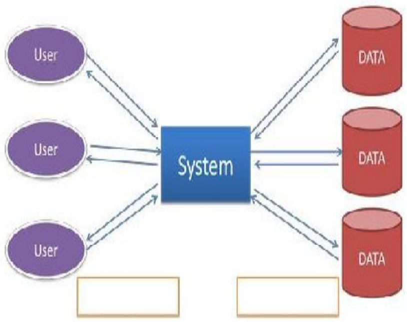

Figure1:-proposed security model

.

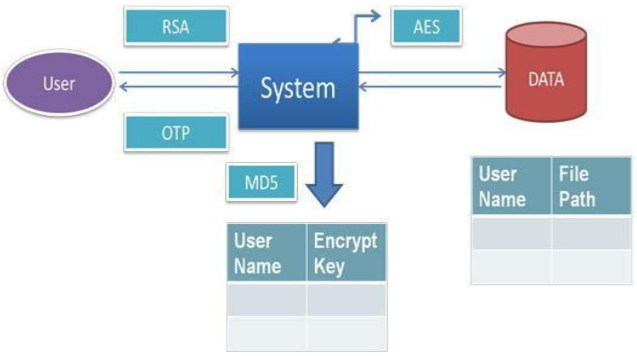

Figure2:Structure

The algorithm for generating the hash table which is used for inserting a file in the database table of the storage server is described below:-**Step 1:** - Select a seed for generating the hash table which is equal to the block size of the table. Block size means with how many positions of files will be taken from a series of execution**Step 2:** - Compute the position where to insert a file. Position = 2.Where represents the no. of file and represents the seed value.**Step 3**:- a) if Position is empty, then insert the file in that Position b) Else, increment the Position and set Offset. Repeat step .A sample hash table with seed S = 100 is shown in table II:

File No In System Server	**Position Of File In Database Server**	**Offset**
1	1	0
2	4	0
: :	: :	: :
5	25	0
: :	: :	: :
15	26	1
: :	: :	: :

TABLE I Synchronization of files in two servers

IV. EXPERIMENTAL RESULTS

In the lab we have worked with about 100 users and also with their files for studying and prove the efficiency of the proposed model. We have tried to find out different execution results which helped us to demonstrate our model with better result. Different conditions and positions were observed during the working and execution time of this proposed model

A. Lab Setup

Platform: Visual Studio 2010 (asp.net)

Processor: Core 2 Duo (2.93 GHz),

RAM: 2 GB

In this environment, the whole model took average of 5 seconds for executing all the steps. This hardware configuration takes highest 2 seconds to encrypt about a 10 KB file. This model is fast enough and can be applied to current cloud computing environments

B. Case Studies

Working with the model in Lab at different times and with different user and their individual files, which are different from each other in size, contents, extension, etc. take different times for executing the overall model. Depending on the file size, program execution time varies from person to person. Among the 100 users result, 10 of them are shown in table III and table I

TABLE II Execution time for Uploading File of 10 People

Person No	File Size	Time Required for file Upload (Full Process)	Person No	File Size	Time Required for file Upload (Full Process)
1	1 KB	3 sec	6	17 KB	10 sec
2	4 KB	5 sec	7	15 KB	10 sec
3	14 KB	9 sec	8	5 KB	5 sec
4	7 KB	6.5 sec	9	2 KB	3 sec
5	9 KB	8	10	8 KB	8 sec

V. CONCLUSION

In this paper we have projected a novel security formation for cloud computing environment which comprises AES, md5, OTP and RSA. The AES is used for file encryption system, RSA system is used for secure communication, Onetime password (OTP) is used to authenticate users in cloud environment and MD5 hashing method is used for hiding information. This model ensures authentication and security for complete cloud computing system.

The execution time is not consequently high because implementation of every algorithm is done in diverse servers. In this proposed system, an intruder cannot effortlessly acquire information and upload the files because he needs to take be in charge of over all the servers, which is quite difficult. In our proposed model we have used RSA encryption system which is deterministic. For this reason, it becomes brittle in long run process. But the other algorithms like AES, MD5 and OTP makes the model highly secured. In future we want to work with certifying protected communication system among users and systems and user to user. In future it can also possible that encryption algorithms will get weak, so we want to work with encryption algorithms to find out more secure encryption system for secured file information protected system.

REFERENCES

1. Yashpal Kadam, “Security Issues in Cloud Computing A Transparent View”, International Journal of Computer Science Emerging Technology, Vol-2 No 5 October, 2011 , 316-322
2. Rohit Bhadauria, Rituparna Chaki, Nabendu Chaki, Sugata Sanyal, “A Survey on Security Issues in Cloud Computing”, 2011
3. Mladen A. Vouk, “Cloud Computing – Issues, Research and Implementations”, Journal of Computing and Information Technology - CIT 16, 2008, 4, 235–246
4. Ye Hu, Johnny Wong, Gabriel Iszlai, Marin Litoiu, “Resource Provisioning for Cloud Computing”, IBM Canada Ltd., 2009
5. Daniele Catteddu, Giles Hogben, “Cloud Computing:- Benefits, risks and recommendations for information security”, November, 2009

6. “Cloud Computing: Silver Lining or Storm Ahead?”, Volume 13 Number 2, Spring 2010
7. NGONGANG GUY MOLLET, “CLOUD COMPUTING SECURITY” , Thesis Paper, April 11, 2011
8. Gunasekar Kumar, Anirudh Chelikani, “Analysis of security issues in cloud based e-learning”, Master’s thesis, 2011
9. Jiyi Wu, Qianli Shen, Tong Wang, Ji Zhu, Jianlin Zhang “Recent Advances in Cloud Security”, JOURNAL OF COMPUTERS, VOL. 6, NO. 10, OCTOBER 2011
10. Ahmad-Reza Sadeghi, Thomas Schneider, and Marcel Winandy, “Token - Based Cloud Computing Secure Outsourcing of Data and Arbitrary Computations with Lower Latency”, TRUST 2010, LNCS6101, pp . 417–429, 2010.
11. Trusted Computing Group, “Solving the Data Security Dilemma with Self-Encrypting Drives”, May 2010
12. Hongwei Li, Yuanshun Dai, Ling Tian and Haomiao Yang, “Identity-Based Authentication for Cloud Computing”, CloudCom 2009, LNCS 5931, pp. 157–166, 2009
13. Sven Bugiel, Stefan Nurnberger, Ahmad-Reza Sadeghi, Thomas Schneider, “Twin Clouds: Secure Cloud Computing with Low Latency”, CASED, Germany, 2011
14. Sven Bugiel, Stefan Nurnberger, Ahmad-Reza Sadeghi, Thomas Schneider, “Twin Clouds: Secure Cloud Computing with Low Latency”-Extended Abstract, CASED, Germany, 2011
15. Luis M. Vaquero, Luis Rodero-Merino, Daniel Morán, “Locking the sky: a survey on IaaS cloud security”, Computing (2011) 91:93–118
16. Yang Tang, Patrick P. C. Lee, John C. S. Lui, and Radia Perlman, “FADE: Secure Overlay Cloud Storage with File Assured Deletion”, 2010
17. Thuy D. Nguyen, Mark A. Gondree, David J. Shifflett, Jean Khosalim, Timothy E. Levin, Cynthia E. Irvine, “A Cloud-Oriented Cross-Domain Security Architecture”, The 2010 Military Communications Conference, U.S. Govt.
18. Cong Wang, Qian Wang, and Kui Ren, Wenjing Lou, “Ensuring Data Storage Security in Cloud Computing”, US National Science Foundation under grant CNS-0831963, CNS-0626601, CNS-0716306, and CNS-0831628, 2009
19. Vaibhav Khadilkar, Anuj Gupta, Murat Kantarcioglu, Latifur Khan, Bhavani Thuraisingham, “Secure Data Storage and Retrieval in the Cloud”, University of Texas, 2011
20. John Harauz, Lori M. Kaufman, Bruce Potter, “data Security in the World of Cloud Computing”, The IEEE Computer SOCIETIES, August, 2009
21. Kevin Hamlen, Murat Kantarcioglu, Latifur Khan, Bhavani Thuraisingham, “Security Issues for cloud computing”, International Journal of Information Security and Privacy, 4(2), 39-51, April-June 2010
22. R.L. Rivest, A. Shamir, and L. Adleman, “A Method for Obtaining Digital Signatures and Public-Key Cryptosystems”, Laboratory for Computer Science, Massachusetts Institute of Technology, Cam-bridge, November, 1977
23. Burt Kaliski, The Mathematics of the RSA Public-Key Cryptosystem, RSA Laboratories
24. Joan Daemen, Vincent Rijmen, “AES Proposal: Rijndael”, 1999
25. Joan Daemen, Vincent Rijmen, “Announcing the ADVANCED ENCRYPTION STANDARD (AES)”, Federal Information Processing Standards Publication 197, November 26, 2001
26. Joshua Holden, Mohammad Musa, Edward Schaefer, and Stephen Wedig, “A Simplified AES Algorithm”, January 2010
27. Ronald Rivest, “MD5 Message-Digest Algorithm”, rfc 1321, April 1992
28. Neil M.Haller, “THE S/KEY ONE-TIME PASSWORD SYSTEM”, 1993
29. Neil Haller, “A One-Time Password System”, October 23, 1995
30. “Securing Data at Rest: Developing a Database Encryption Strategy”- A White Paper for Developers, e-Business Managers and IT
31. Ulf T. Mattsson, “Database Encryption - How to Balance Security with Performance”, 2004