International Journal of Scientific Engineering and Applied Science (IJSEAS) – Volume-2, Issue-12, December 2016
ISSN: 2395-3470
www.ijseas.com

SECURING THE AUTOMATION PROCESS BY PREDICTION OF USER CLASSIFICATION AND MFA

Dr. M. Parvathy[1], M.Kumudha[2], J.Dinesh[3]
Prof/ CSE[1] , AP/CSE[2], AP/CSE[3]
[1,2,3]Department of Computer Science and Engineering
[1,2,3]KLN College of Information Technology,Pottapalayam, Tamilnadu, India
parvathydurai2002@gmail.com[1], kumuvani@gmail.com[2] , yuvadinesh@gmail.com [3]

Abstract

Cloud computing is an emerging data interactive paradigm to fulfill user's requirements based on the demand. The existing system focuses more on Threat reduction by using MFA which is more secure but reduces the chances of availability. The Objective of the proposed work is to bridge the gap between Security and Quality of Service. Here the security directives are made complex for the malicious users, at the same time easy availability of data by valid users. The better protection mechanisms are provided to safeguard the system. The system identifies the type of users and avail accessibility accordingly. Multifactor Authentication is provided for protecting automation process of workload sharing through server virtualization. Experimental simulation mechanisms prove better results.

Index terms: *Multifactor Authentication, OAuth Token, Instantly generated Token, Dynamic Mixed Concatenation*

1. Introduction

Cloud is very flexible, more reliable and also elastic that is on demand by most of the users. The resource allocation in cloud needs to be more systematic and the data in cloud needs to be accessible only by those who are authenticated. Security directives are key challenges, but are difficult to achieve. The traditional methods of managing security aren't scaling to the growth of the threat landscape. Also implementing more security mechanisms leads to complexity and therefore provides less availability even to the authorized users.

Data Integrity, Confidentiality and Authentication and timely delivery of data has to be ensured. It is the responsibility of any system to distribute, balance the load for speeding up the response and to reduce the work of the server. Hence automation mechanisms and security directives are essential so that the server may not be responsible for all the tasks at all the time.

2. Literature survey

Sumathi M, Sharvani G.S , Dinesha H A (2013). Implementation Of Multi Factor Authentication System For Accessing Cloud Service, International Journal of Scientific and Research Publications, 3, 1849-1855.

The system deals about implementing Multi-factor Authentication in the cloud environment that allows more layers of security for protecting the resources and make it complex for the malicious user to get access to the resources. The multiple layers include Organization/Service Authentication Passwords that accepts image feature as input and more storage is essential. This is followed by Team Authentication Password and the User Authentication Password. The limitations are the Server needs to store the seeds of the portfolio images of each user in plain text and Selecting a set of pictures from the picture database is tedious and time consuming.

Niharika Gupta, Rama Rani (2015) Implementing High Grade Security in Cloud using Multifactor Authentication and Cryptography. International journal of Web & Semantic Technology,6, 9-17.

The paper implements multiple security layers using cryptographic mechanisms. Hashing is been done on the data for protecting the password. It deals with multiple hash functions like SHA1, MD5, and SHA512. Password will be hashed in a different way and will be less prone to reverse engineering. Different key is used every time for each user so that it adds more to the security. The limitation here is the avalanche effect. If there is a change in one bit then it is reflected in multiple bits.

Prachi Soni, Monali Sahoo (2015) Multi-Factor Authentication Security Framework in Cloud Computing, International Journal of Advanced Research in Computer Science and Software Engineering, 5, 1065-1071.

The key used for authentication is splitted into two halves. One half is sent as email id provided by the client in the process of registration and the second half is send as SMS to the mobile phone no. provided at the time of registration of the user. The system deals with issues due to static passwords and also other security solutions

The drawbacks here is it requires simultaneous access of mobile and email; sometimes it may not be possible and splitting key value may sometimes cause confusions

3. Objectives

The system is intended to reduce the load of the server by using Virtualization Technology and automates the entire process. This allocates data center resources dynamically based on application demands and optimizes the number of servers in use thereby achieving Green computing. It also achieves greater security for protection of the above task that can never be reengineered. A system needs to be developed where the security directives are made complex for the malicious users, at the same time easy availability of data by valid users. The better protection mechanisms are provided to safeguard the system. Data availability for the authenticated users is provided in less time with less security verification. The system also develops a set of heuristics that prevent overload in the system effectively while saving energy used.

4. Proposed Work

The system proposes an attractive model for implementation of Multifactor Authentication techniques. The System identifies and classifies the users. A threshold value is assigned. For example the rank is assigned for each individual user based on the activities and depending upon the value of rank the system assigns to each user. If the value exceeds the threshold, the user is considered as malicious, then an iterative model of security mechanism is directed to the user. If the value lies below threshold, the user is considered to be authenticated and security mechanisms are designed with less complexity for easy availability. Various layers of VM's are used and mapped to some particular server such that Server1 →VM1, VM2…VMn; Server2 → VM1, VM2…VMn; Server3 → VM1, VM2… VMn. The load distribution process by the server on the VM's is automated by Ultrasonic Protocol and all the VM's are evaluated for Resource allocation.

MFA increases the security levels which will become tedious for availing data. The level of security for the authenticated and unauthenticated users varies based on the classification. The system concentrates on securing the automation process of workload distribution. Multifactor Authentication (MFA) at different levels is implemented that allows multiple security levels based on what type of user the system predicts. The above is implemented such that the security mechanisms are very difficult to reengineer. Also the VM's of one machine can be operated in parallel from the other Machine.

5. Module Description

- Multifactor Authentication
- Workload sharing

5.1 Multilevel oAuth Authentication

A secure Token/Key integrated with a Dynamic code apart from traditional password system. The above is the combination of knowledge factor and possession factor. These two factors are

concatenated in different ways within the specified time limit. If the concatenation of both static and dynamic pin is right, then the user is permitted to next step. The three layers of security is

Instant Token Generation
oAuth Token Generation
Mixed Concatenation Technique

The first level is to use the UserID and RID generated during registration for logging in. if the login is successful the user will have to proceed to Instant token generation phase.

Instant Token Generation
(4 digit registered token + instantly generated token)

RID is a unique token of four digits which is a server generated random number. The Instant token will be prevailing only for specific time period. The user has to provide the right combination of input within time. The input will not be accepted after the time exceeds. The token generated is of random cycles. Other set of alphanumeric characters are generated if the given input is wrong or not given within the time given. The Instance for the generated token will be running on the server. The system authenticates if the combinations of input is right.

1st 10 seconds = 4 digit Registered Token + Instance Number1

2nd 10 Seconds = 4 digit Registered Token + Instance Number2 … etc

oAuth Token Generation

(Open Authentication→ 1st 2 digit of registered token + OTP (1st half) token

oAuth token is the concatenation of the same Registration Id and the OTP. Here the combination is different from Instant token generation but the input has to be given within the timestamp. The 1st two characters of RID, 1st two characters of the OTP, next two characters of the RID, and the last two characters of the OTP is the combination followed for this security module.

RID: ABCD
OTP:1234
Combination: AB12CD34

Dynamic Mixed concatenation

If the system identifies the user as malicious or if the system is not able to predict the user then it redirects to the following layer of security mechanism. Here each dynamic combination of input has to be provided by the user. Each complete input is given within the mentioned time duration in cyclic combinations. (RID+OTP)

RID: ABCDEFGH; OTP: 12345678
Time: 20 Seconds

DC1: ABCDEFGH12345678; r(DC1)
DC2: 12345678ABCDEFGH; r(DC2)

DC3: ABCD1234EFGH5678; r(DC3)
DC4: 1234ABCD5678EFGH; r(DC4)
DC5: DCBA1234HGFE5678; r(DC5)
DC6: DCBA4321HGFE8765; r(DC6)
DC7: 1234DCBA5678HGFE; r(DC7)
DC8: 4321DCBA8765HGFE; r(DC8)

DC9: AB12CD34EF56GH78; r(DC9)
DC10:12AB34CD56EF78GH; r(DC10)
DC11:BA12DC34FE56HG78; r(DC11)
DC12:BA21DC43FE65HG87; r(DC12)
DC13:12BA34DC56FE78HG; r(DC13)
DC14:21BA43DC65FE87HG; r(DC14)

DC(15): A1B2C3D4E5F6G7H8; r(DC15)
DC(16): 1A2BC3D4E5F6G7H8; r(DC16)
DC(17): H1G2F3E4D5C6B7A8; r(DC17)
DC(18): H8G7F6E5D4C3B2A1; r(DC18)
DC(19): 1H2G3F4E5D6C7B8A; r(DC19)
DC(20): 8H7G6F5E4D3C2B1A; r(DC20)

DC = Dynamic Concatenation;
r(DC) = Reverse of DC

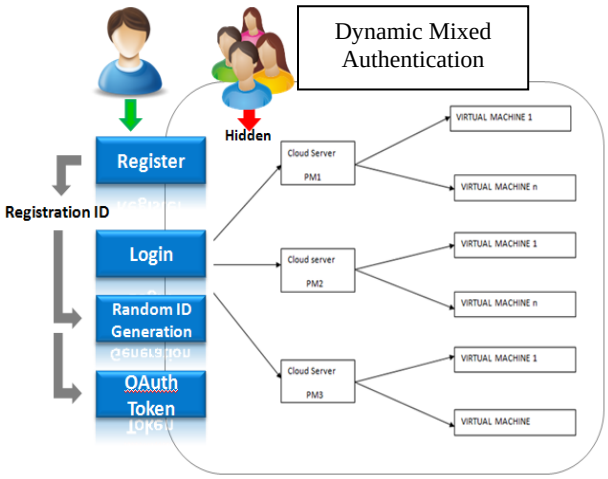
5.2 Workload sharing

The module deals with the user requesting to download the required file. The request will be stored and processed by the server to respond the

user. It checks the appropriate sub server which is nothing but the Virtual Machines to assign the task. VM's are created and connected with all servers to perform the user requested tasks. Sharing the server data to VM's i.e., Sub-servers located as layers is been performed. Task is been performed by Virtual Machines (VM's) and the control is been provides by Physical Machines (PM's).

6. Experimental work

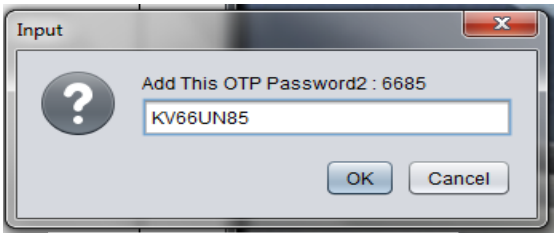
The system is multi-step process that focuses on Security, data structure software architecture, procedural details, (algorithms etc.) and interface between modules. The workload sharing and security is aggregated into two separate modules. Multiple files of different formats were collected and stored for processing by multiple VM's. The data set used for extension of security includes alphanumeric characters. The dynamic mixed concatenation uses a combination of 8 characters and 8 numbers in different ways. A total of 40 combinations are provided to improvise the security.

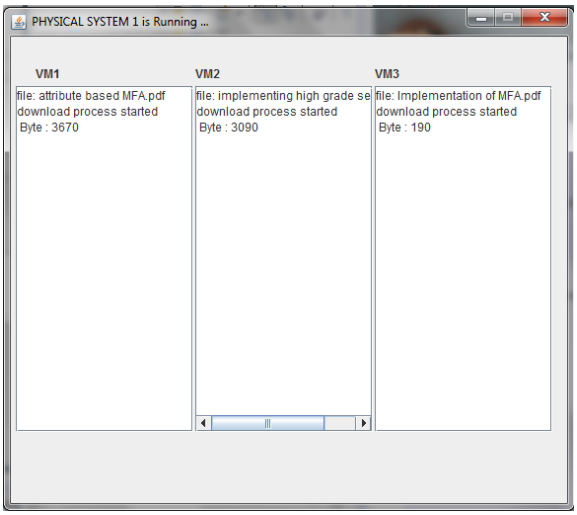


Architecture

7. Results and Implementation

Instant Token





pname	ipaddress	port	lname	status
psystem 1	127.0.0.1	2222	vm1	1
psystem 1	127.0.0.1	2223	vm2	1
psystem 1	127.0.0.1	2225	vm3	0
psystem 2	127.0.0.1	3333	vm1	0
psystem 2	127.0.0.1	3335	vm2	0
psystem 2	127.0.0.1	3336	vm3	0
psystem 3	127.0.0.1	5551	vm1	0
psystem 3	127.0.0.1	5553	vm2	0
psystem 3	127.0.0.1	5559	vm3	0
NULL	NULL	NULL	NULL	NULL

Database

Conclusi **Task assignment**

The system bridges the gap between Security and QoS. The system is also scalable so that it adapts multiple VM's. Security is a critical parameter. Load balancing integrated with security will be more efficient for developed and developing applications. The system meets the need i.e., fulfills the demand and supply of the resources and implements strong security by trying different

concatenation. The future work is to try different concatenation within the specific time. It is not possible for the hacker to try different combination and concatenation of input key within the mentioned timestamp.

References

[1] Ms.M.Kumudha, Dr.M.Arunachalam: "Automating resource management of sub servers in cloud computing environment", International Journal of Scientific Engineering and Applied Science, vol.2, No.4, 2016

[2] Niharika Gupta, Rama Rani: "Implementing High Grade Security in Cloud using Multifactor Authentication and Cryptography", International Journal of Web & Semantic Technology (IJWesT) Vol.6, No.2, April 2015

[3] Sumathi M, Sharvani G.S, Dinesha H A: "Implementation of Multi Factor Authentication System For Accessing Cloud Service", International Journal of Scientific and Research Publications, Volume 3, Issue 6, June 2013

[4] Deepa Panse, P. Haritha: "Multi-Factor Authentication In Cloud Computing For Data Storage Security", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 4, Issue 8, August 2014

[5] T.Lakshmi Praveena, V.Ramachandran, CH. Rupa: "Attribute based Multifactor Authentication for Cloud Applications", International Journal of Computer Applications (0975 – 8887)Volume 80 – No 17, October 2013

[6] P. Mell and T. Grance, "Draft NIST Working Definition of Cloud Computing," Nat'l Inst. of Standards and Technology, 2009.

[7] A. Mishra, R. Jain, and A. Durresi, "Cloud Computing: Networking and Communication Challenges," IEEE Comm. Magazine, vol. 50, no. 9, pp. 24-25, Sept. 2012.

[8] R. Moreno-Vozmediano, R.S. Montero, and I.M. Llorente, "Key Challenges in Cloud Computing to Enable the Future Internet of Services,"IEEEInternet Computing,vol.17,no. 4,pp. 18-25July/Aug.2013.

[9] K. Hwang and D. Li, "Trusted Cloud Computing with Secure Resources and Data Coloring," IEEE Internet Computing, vol. 14, no. 5, pp. 14-22, Sept./Oct. 2010.

[10] J. Chen, Y. Wang, and X. Wang, "On-Demand Security Architecture for Cloud Computing," Computer, vol. 45, no. 7, pp. 73-78, 2012.

[11] Y. Zhu, H. Hu, G. Ahn, and M. Yu, "Cooperative Provable Data Possession for Integrity Verification in Multi-Cloud Storage," IEEE Trans. Parallel and Distributed Systems, vol. 23, no. 12, pp. 2231-2244, Dec. 2012.

[12] H. Wang, "Proxy Provable Data Possession in Public Clouds," IEEE Trans. Services Computing, vol. 6, no. 4, pp. 551-559, Oct.-Dec. 2012

[13] K. Yang and X. Jia, "An Efficient and Secure Dynamic Auditing Protocol for Data Storage in Cloud Computing," IEEE Trans. Parallel and Distributed Systems, vol. 24, no. 9, pp. 1717-1726, Sept. 2013.