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

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 4, April 2017

Implementation of Secure Authentication Scheme and Access Control in Cloud Computing

Abhishek Patel, Prof. Ashok Verma

Research Scholar, Dept. of Computer Science, Gyan Ganga Institute of Technology and Science, Jabalpur, M.P, India

Associate Professor & HOD, Dept. of Computer Science, Gyan Ganga Institute of Technology and Science, Jabalpur, M.P, India

ABSTRACT: Cloud computing provides problem solving services, software services, data access services and storage accommodations that don't require end-users information of the materialistic positioning and hardware topology of the system. This also leads to the security threats in authentication and data access. To overcome the security threats during authentication we will implement a Remote Authentication Dial-In User Service (RADIUS) system, which will use One Time Password (OTP), username and password for login into cloud from outside of the network

KEYWORDS*:* Cloud computing, OTP, PKI, SMTP, OPENSTACK

I. INTRODUCTION

Cloud computing technology is seen because the assortment of web primarily based services for higher utilizing the resources and services. It's the new utility that provides virtualization, parallel and distributed computing into single unit. It implies the sharing of resources to handle applications with reduces capital and low maintenance price. It offers magnified quantifiable and easy access feature with low quality. There can be three main ways in which cloud services are utilized; they are Software as a Service (SaaS), (Platform as a Service) PaaS and Infrastructure as a Service (IaaS). These services can be deployed by Private, Public, Hybrid or Community cloud.

The data present in the cloud is easily susceptible to many kind of attack. The data which is present in the cloud should not get served to any user without knowing the user details, that the user is eligible to access this data is not, hence this gives rise to the strong authentication requirement for cloud based resource. Even if the authentication requirement is matched then there should be some mechanism which should take care access rights of the users, hence this gives rise to develop the access control mechanism which should take care whether the resource is being accessed to proper user or not, or in short we can say "Who can access what".

Here, enhanced security model has been proposed to overcome the gap of RADIUS system. It engages mobile phone instead of security token or public key infrastructure for strong association/interaction of user during authentication process. Use of mobile based OTP is more practicable than security token because no need to carry extra device for authentication purpose. Subsequently, Public Key

Infrastructure (PKI) suffers with certificate issue and poor speed along with absence of user involvement. Certificate issue can be resolved but poor speed of authentication can lead of susceptible time period. Attacker may attempt to capture the packet and reformat it after fabricating to get entry into system. Default authentication scheme can be dangerous for public system which may login without permission of user.

DOI: 10.15680/IJIRCCE.2017. 0504396

In order to differentiate public and private cloud access, user authentication in private cloud has been implemented using user id & password and IP based authentication system. Afterwards, Along with user id and password, the OTP verification has been done for public cloud login purpose. Involvement of mobile phone required user presence at time of login where PKI does not. For implementing access control this work classified users into three categories can be written as role to make access standard and transparent. The details of user-roles and assigned privileges are explained in next sections.

The first section of this paper serves the Introduction part which is followed by the related work section in which described the research work which has already been done corresponding to cloud computing. It also helps to explore the analysis of present system and gap observations which lead to find problem statement. Problem statement and methodology has been described in third and fourth section followed by system architecture and enhanced security model for cloud computing.

In fifth section we have calculated the result for two services, the first is upload file in which we have calculated total time to upload a file to server and file size after encryption process, the second service is download file in which we have calculated download time and file size after decryption and download process. In the sixth section the final conclusion of the paper is presented.

II. RELATED WORK

Xiong et al. showed that the cloud computing services can serve the user with computer hardware and software with ease [1]. As we have already seen that we can access cloud services via public, private and hybrid cloud based deployment model. The first deployment model we know is public cloud, which was very thoughtful approach by Amazon, which offers high performance and bulk data volume storage. But public cloud is susceptible to privacy, threat and other security threat like man in the middle attack.

Smoot et al. presented their thoughts on the private cloud based deployment model. According to them, most of the organization/software companies are using this model for providing the services. These services can be provided only to the users who are allowed to work from inside the organization/company. The private cloud service is very expensive method and it has many restrictions regarding access of the resource accessibility [2].

According to Bicer et al. hybrid cloud deployment makes the person to access/utilize the public and private resource at an ease at an optimal cost and minimum time [3]. The model of hybrid cloud must agree, guarantee and follow the rules and regulations of the Network, Database and security aspect of sharing public and private cloud. Hybrid cloud service must understand and take care of the weak points of the public and private cloud services and should provide services without any security issues.

According to Zhang et al. Hybrid computing development, however, is hampered by privacy concerns. A significant amount of organizational computing workload at least partially involves sensitive data and therefore cannot be directly outsourced to the public cloud [4]. Hence there is a big threat to security. Prasadreddy et al and Varadharaj et al. concluded that there is the need of secure authentication system for hybrid cloud service [5, 6].

B. Prasanalakshmi, et al. worked on the same secure authentication in cloud and provides the solution to this problem by implementing additional biometric authentication factor. They use three biometric traits, the face, the fingers and palm veins to develop a secure authentication system [7]. Though this could be the most accurate authentication scheme but the main disadvantage of biometric authentication system is that it adds additional periphery which requires additional cost that standard user is not willing to pay.

In April 2013, Alina et al provided the Hybrid Text-Image based authentication for cloud services [8]. In this approach, for authenticating the user, the first point of validation is user text based authentication (ID + Password) and second point of validation is Image based authentication. Every time when the user will be asked to provide his/her identity, a form for each image including the photo will be listed. The user will have to remember the secret code for each image and to carefully introduce in the form. The main drawback of this approach is that if user forgets the code for the image then he/she will face lots of trouble to access the cloud based resources.

Jin-Mook Kim and Jeong-Kyung Moon [9] proposed a RADIUS server based authentication scheme to maintain security service during resource sharing. Sharing is important because of computer resource reuse, server device cost, and space problems. Private cloud can be used to solve the issue of authentication and confidentiality but can't be feasible for various applications. They proposed a solution with involvement of security token and RADIUS server for authentication purpose to make a difference between public and private cloud login.

III. PROBLEM STATEMENT

As a problem statement, the system proposed by Jin-Mook Kim and Jeyong-Kyung Moon [9] does not maintain privacy of content during communication and database storage. Database keeps all information in plain text format. It does not imply any access control feature or algorithm, which means anybody, can access anything once authentication happens successfully. System is Vulnerable for interception and fabrication attacks. Existing model gives security solution for hybrid clouds. Hence this security model cannot be applied to private and public clouds as well. There is no distinctive algorithmic rule for OTP generation. The security threat/problems in cloud computing includes Data security, Identity and access management, Key management, Virtual machine security.

Few other threats which are identified in cloud computing are listed in the Table 1.

Table 1: Security Threats in Cloud Environment

Attack	Description
Tampering	Attacker may alter or fabricate information
Eavesdropping Information Disclosure	Attacker may listen or read the information
Repudiation	Attacker may Refuse the validity or claim of information or service
Man-in-the-Middle Attack	Attacker may intercept the communication and deploy third party involvement
Replay Attack	Attacker may hold and resend the packet information after a time delay.
Identity Spoofing	Attacker may kill or misuse the identity of node, server or client.
Viruses and Worms	Attacker may use certain bad source code to compromise

DOI: 10.15680/IJIRCCE.2017. 0504396

IV. SOLUTION STATEMENT

As a solution we have proposed a security model to enhance the authentication and access control which will also maintain level of security in terms of confidentiality and integrity too for private, public and hybrid cloud.

Here we have used asymmetric key based RSA algorithm to make complete communication secure and encrypted by converting all the plain text information into cipher text information by using 2048 bit size key. Here the question arise that why we have used 2048 bit size key, the answer is that according to U.S. National Institute of Standards and Technology, 768 bit is the largest key size which has been cracked, also it has been estimated that 1024 bit key size might get break in next 5 years, hence we have selected 2048 bit as the key size. As we know that key size is directly proportional to the encryption strength, hence more is the key size the more is the encryption strength and more is the data security. Subsequently, as per RSA standards size of plain text that can be encrypted can't be more than 1/8th value of plain text.

For implementing access control issues we have used Role based access control (RBAC) technique, where we classify each user on the basis of three role i.e. admin, manager, customer and assign different services as per their roles and responsibilities. The admin has the highest privileges and then comes the manager finally in the last with least privilege the customer comes. Table 2 describes the same thing.

Table 2: The below table shows the level of the user

User Role	Privilege	Level
Customer User / Client	Limited Access	1 [Low]
Manager	Advance Access	2 [Medium]
Administrator	Full Control with Transparency	3 [High]

Implementation Flow of Proposed Solution:

1. User Registration

2. Login

    2.1 User verification (Username & Password method)

    2.2 IP Verification for Private Cloud

    2.3 OTP Verification (For public cloud)

3. Perform Services

    3.1 Upload File

        3.1.1 Calculate MD5 of respective file and transmit stored into key management table with file name.

        3.1.2 Divide the complete file into multiple chunks.

3.1.3 Perform RSA Encryption Algorithm.

3.1.4 Upload file to Cloud Server.

3.2 Download File

3.2.1 Retrieve all connected chunks.

3.2.2 Decrypt and Integrate into single file.

3.2.3 Calculate new MD5 and compare with the previous value, in case of different value it retries for download else perform download operation.

4. Logout

Following steps are involved to implement the above described solutions.

1. Initially customer is registered by filling registration form. As soon as registration gets completed a request is generated for Manager to approve the customer. On manager's approval, the customer will now be able to perform and use the services which are bind to him.

2. If customer wants to access the services from private cloud, then the registered IP address of the customer system, Username and Password based authentication will take place.

3. If customer wants to access the services from outside private cloud i.e. from public cloud then that IP address of the system is not registered, hence OTP based authentication gets added with username and password.

4. There can be three level of registered user/customer. The highest level is admin level which will have all the authority and access. The next level is manager which has medium authority and access. Manager can keep an eye on user/customer's requests. The lowest level is for User/Customer. The Table 3 will list all the roles with their privileges for the proposed system.

Table 3: List of User Role with assigned privileges

User Role	Privileged Services
General User/Customer	Login & Logout
	Upload the file
	Download File [Individual]
	View File Information[Individual]
	Delete File[Individual]

DOI: 10.15680/IJIRCCE.2017. 0504396

	Update Password
	Update Base Computer IP Address
	Update Profile
Manager	All Features of General User at Individual level
	Search & View Permitted Details of user (Except Password & Mobile Number)
	Sanction User access
	Sanction Update IP Address Permission of user
	Request to Delete User
Administrator	Create Manager
	Search & View Permitted Details of user (Except Password & Mobile Number)
	Sanction Update IP Address Permission of Manager
	Delete User / Manager
	View Total Storage Consumed [User Wise]

For achieving the confidentiality in the system we have used RSA algorithm. We have used 2048 bit key for maximum security. For message authenticity we have used MD5 algorithm which will generate 32 bit of digest. This digest will be used for validating the file content at the time of decryption process and hence maintaining the authenticity of the files in the system. As an authentication scheme if user logins from private cloud then we have used IP along withUsername and password for authentication. If user logins from outside of the organization then we have added an extra OTP based authentication scheme. For implementing access control we have used RBAC scheme which is explained in the earlier paragraphs. Table 4 explains the implementation of security principle and suitable algorithm

Table 4: Used Algorithm and principle

Confidentiality	RSA
Authentication	1. User Id-Password Mechanism 2. IP-Based Authentication [**Private Cloud Authentication**] 3. OTP Based Authentication **[Public Cloud Authentication**]

Integrity	MD5

V. RESULTS

As we have already seen that for adding security in authentication biometric method has been used, tokens generated from tokenize method has been used, but it all requires additional hardware. Use of OTP in authentication will use just one message on user’s cell phone and authentication will be done. Hence the use of OTP sending on user’s cell phone to access resource and services from outside cloud is best suited method for authentication.

Step 5.1 Remotely Access OpenStack Dashboard

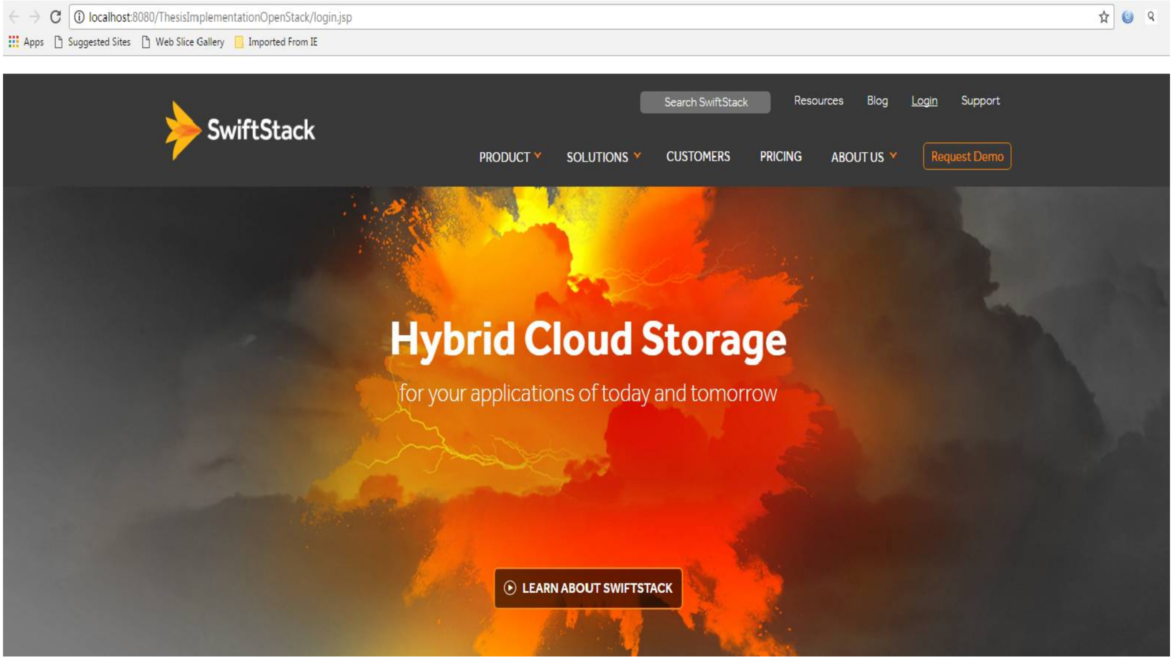
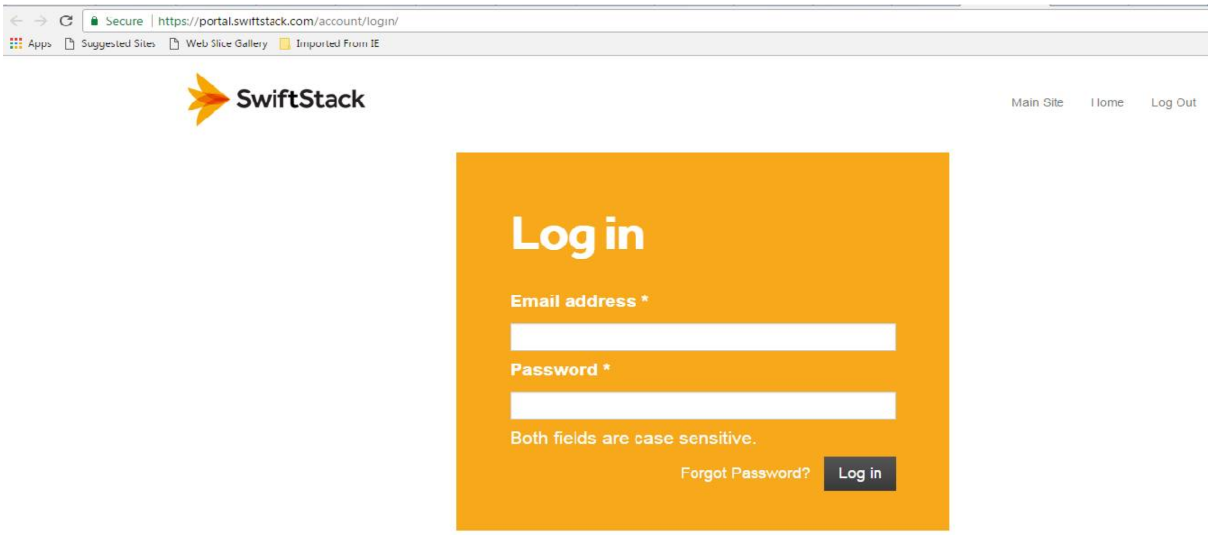


Figure 5.1 OpenStack Dashboards



Log in

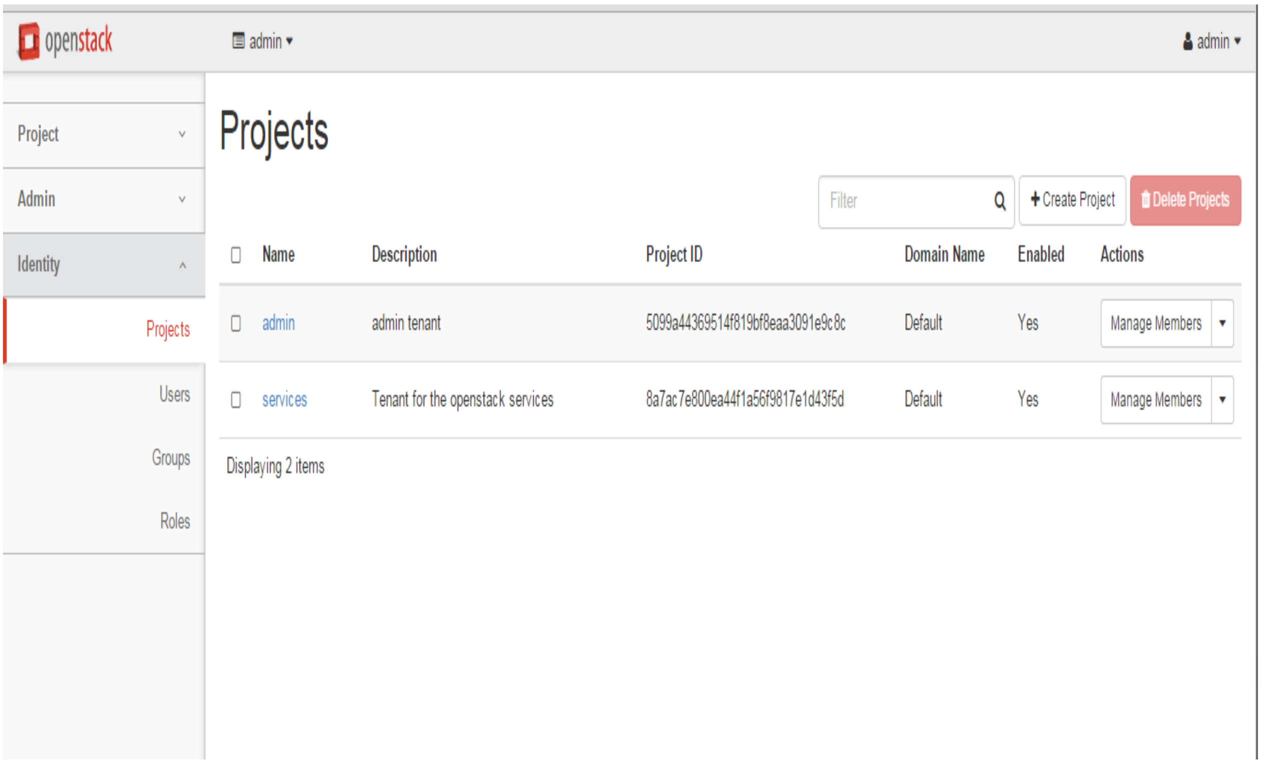
Email address *

Password *

Both fields are case sensitive.

Figure 5.1.1 OpenStack Login Dashboards

OpenStack is a group of open source project that use shared virtual resources to build private and public clouds. Projects handle the core cloud-computing services of compute, networking, and storage. Projects can be bundle together to create deployable clouds.



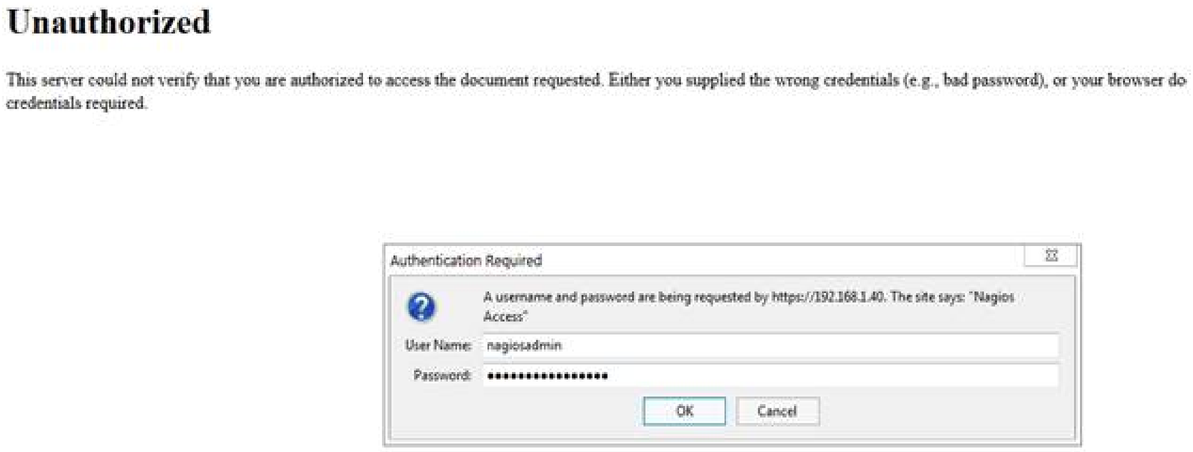
Projects

□	Name	Description	Project ID	Domain Name	Enabled	Actions
□	admin	admin tenant	5099a44369514f819bf8eaa3091e9c8c	Default	Yes	Manage Members ▾
□	services	Tenant for the openstack services	8a7ac7e800ea44f1a56f9817e1d43f5d	Default	Yes	Manage Members ▾

Displaying 2 items

Figure 5.2 Openstack Projects

on the other hand, if we opt to install **Nagios** component for **OpenStack.**



Unauthorized

This server could not verify that you are authorized to access the document requested. Either you supplied the wrong credentials (e.g., bad password), or your browser do
credentials required.

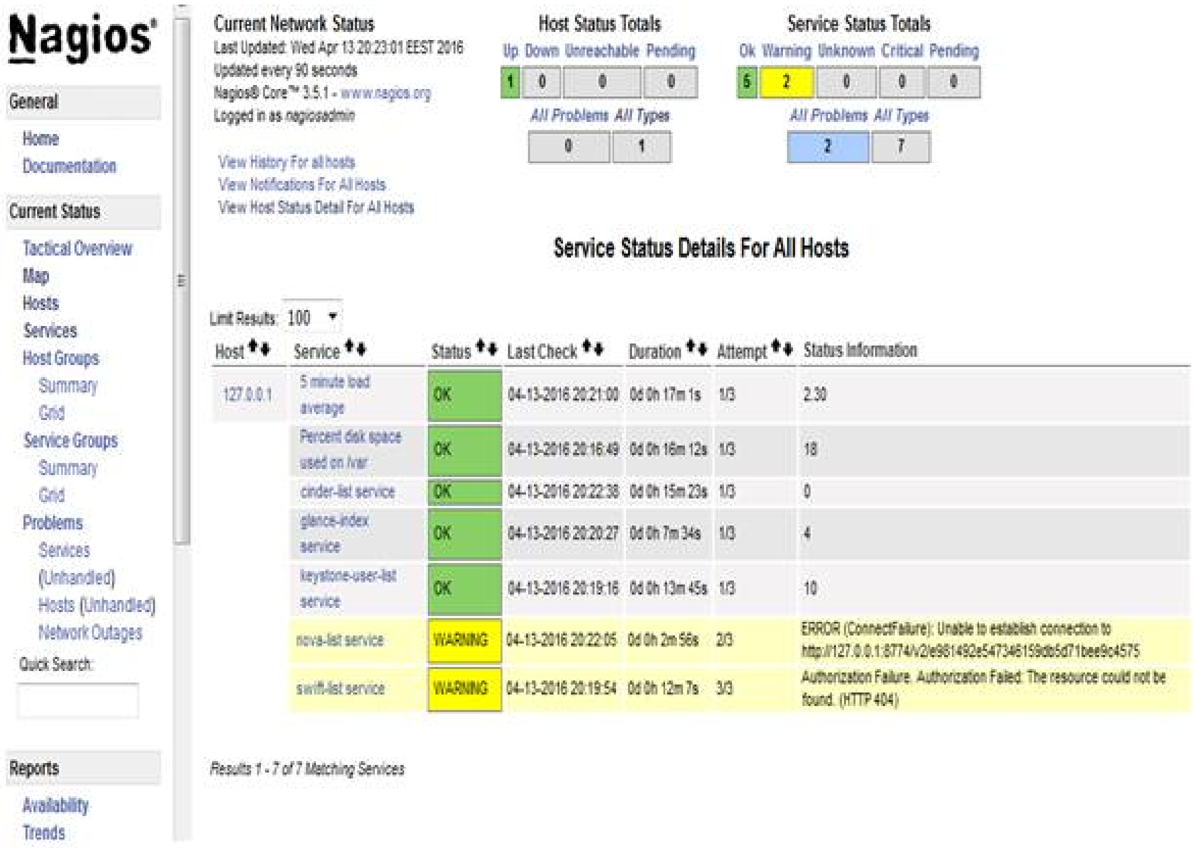
Authentication Required

A username and password are being requested by https://192.168.1.40. The site says: "Nagios Access"

User Name: nagiosadmin

Password: ••••••••••••••••

Figure 4.3 Nagios Login Dashboard



Current Network Status
Last Updated: Wed Apr 13 20:23:01 EEST 2016
Updated every 90 seconds
Nagios® Core™ 3.5.1 - www.nagios.org
Logged in as nagiosadmin

View History For all hosts
View Notifications For All Hosts
View Host Status Detail For All Hosts

Host Status Totals

Up	Down	Unreachable	Pending
1	0	0	0

All Problems	All Types
0	1

Service Status Totals

Ok	Warning	Unknown	Critical	Pending
5	2	0	0	0

All Problems	All Types
2	7

Service Status Details For All Hosts

Limit Results: 100

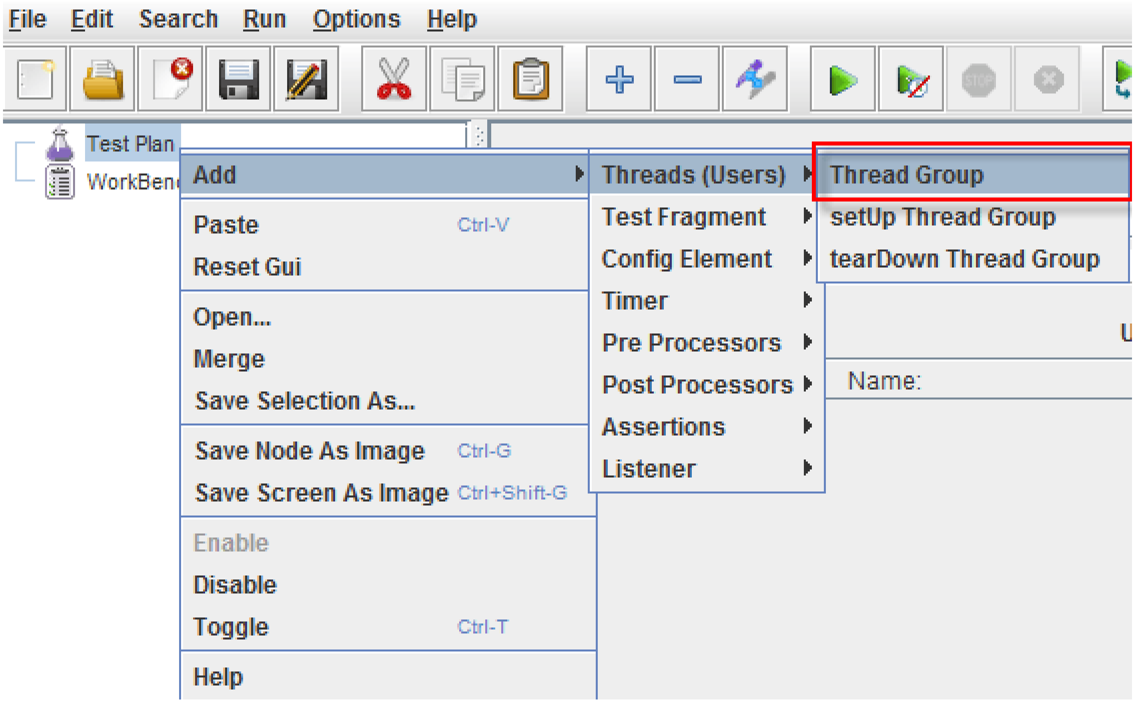
Host	Service	Status	Last Check	Duration	Attempt	Status Information
127.0.0.1	5 minute load average	OK	04-13-2016 20:21:00	0d 0h 17m 1s	1/3	2.30
	Percent disk space used on /var	OK	04-13-2016 20:16:49	0d 0h 16m 12s	1/3	18
	cinder-list service	OK	04-13-2016 20:22:38	0d 0h 15m 23s	1/3	0
	glance-index service	OK	04-13-2016 20:20:27	0d 0h 7m 34s	1/3	4
	keystone-user-list service	OK	04-13-2016 20:19:16	0d 0h 13m 45s	1/3	10
	nova-list service	WARNING	04-13-2016 20:22:05	0d 0h 2m 56s	2/3	ERROR (ConnectFailure): Unable to establish connection to http://127.0.0.1:8774/v2/e981492e547346159db5d71bee9c4575
	swift-list service	WARNING	04-13-2016 20:19:54	0d 0h 12m 7s	3/3	Authorization Failure. Authorization Failed: The resource could not be found. (HTTP 404)

Results 1 - 7 of 7 Matching Services

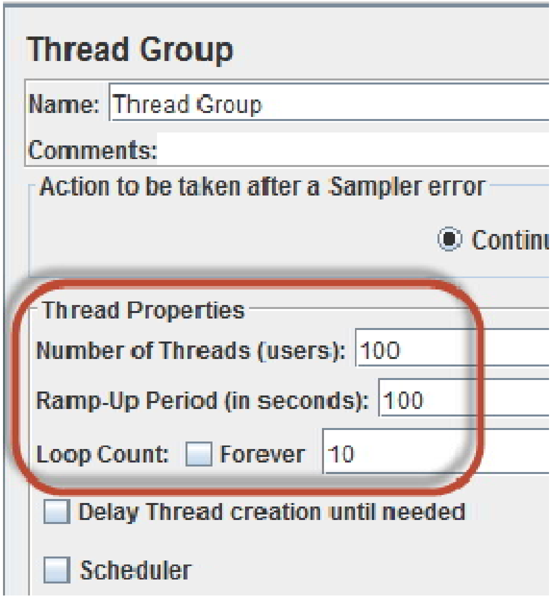
Figure 5.4 Nagios Linux Monitoring Interface

Performance Testing is crucial to determine that the web application under test will satisfy **high load** requirements. It can be used to analyze overall server performance under heavy load.

Add Thread Group
> Thread Group



In Thread Group control panel, enter Thread Properties as following:


Thread Group

Name: Thread Group

Comments:

Action to be taken after a Sampler error

● Continu

Thread Properties

Number of Threads (users): 100

Ramp-Up Period (in seconds): 100

Loop Count: □ Forever 10

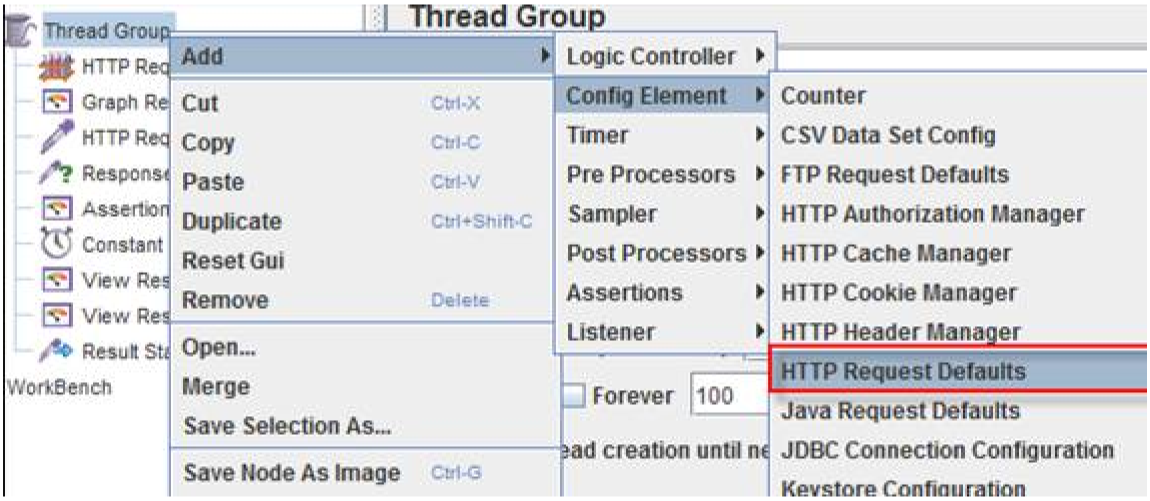
□ Delay Thread creation until needed

□ Scheduler

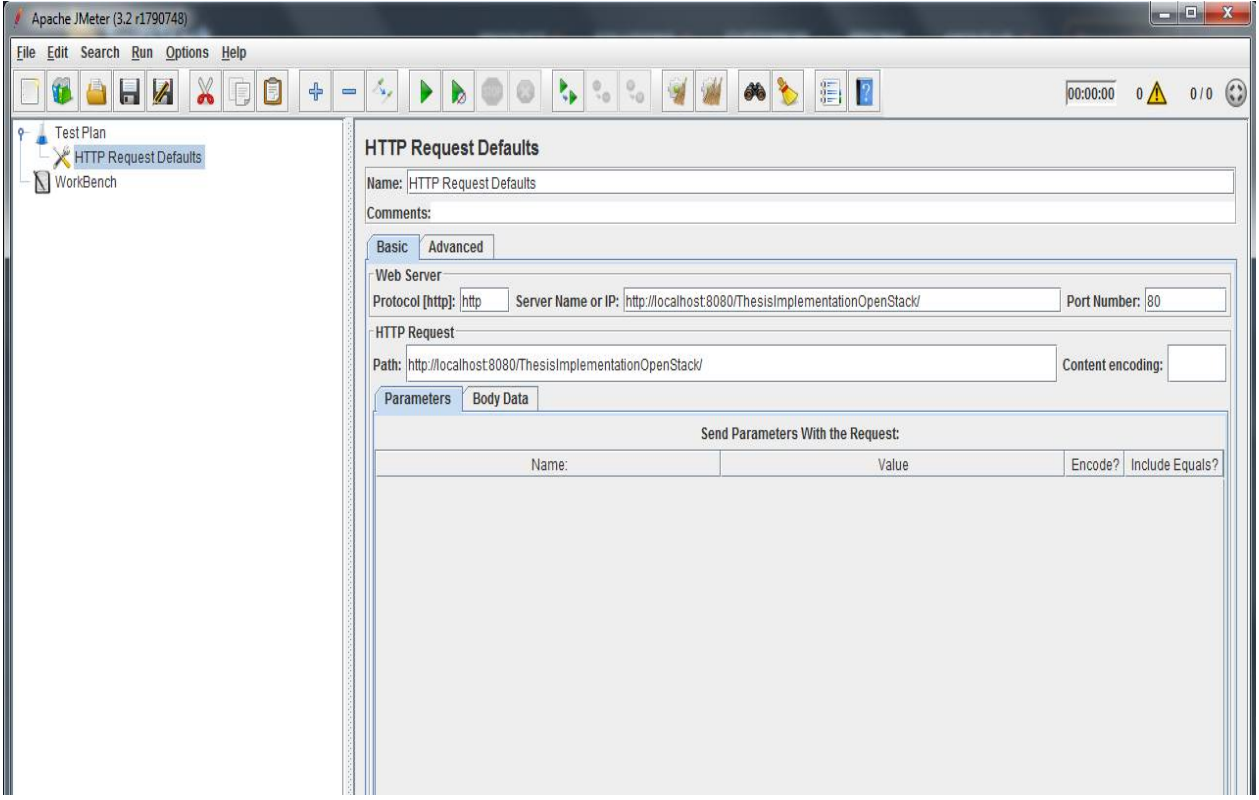
- **Number of Threads**: 100 (Number of users connects to target website: 100)
- **Loop Count**: 10 (Number of time to execute testing)
- **Ramp-Up Period**: 100

Step 2) Adding JMeter elements

This element can be added by right-clicking on the Thread Group and selecting: **Add -> Config Element -> HTTP Request Defaults.**



In the HTTP Request Defaults control panel, enter the Website name under test (**http://localhost:8080/ThesisImplementationOpenStack/**)



HTTP Request Defaults

Name: HTTP Request Defaults

Comments:

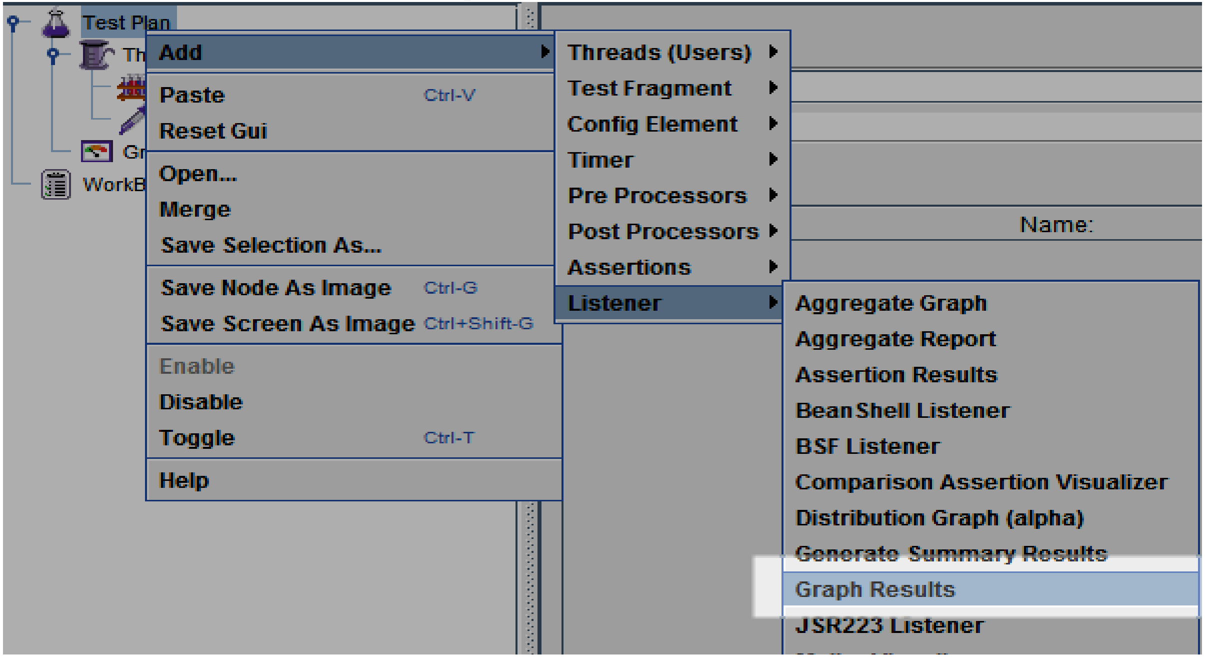
Protocol [http]: http
Server Name or IP: http://localhost:8080/ThesisImplementationOpenStack/
Port Number: 80

Path: http://localhost:8080/ThesisImplementationOpenStack/
Content encoding:

Step 3) Adding Graph result

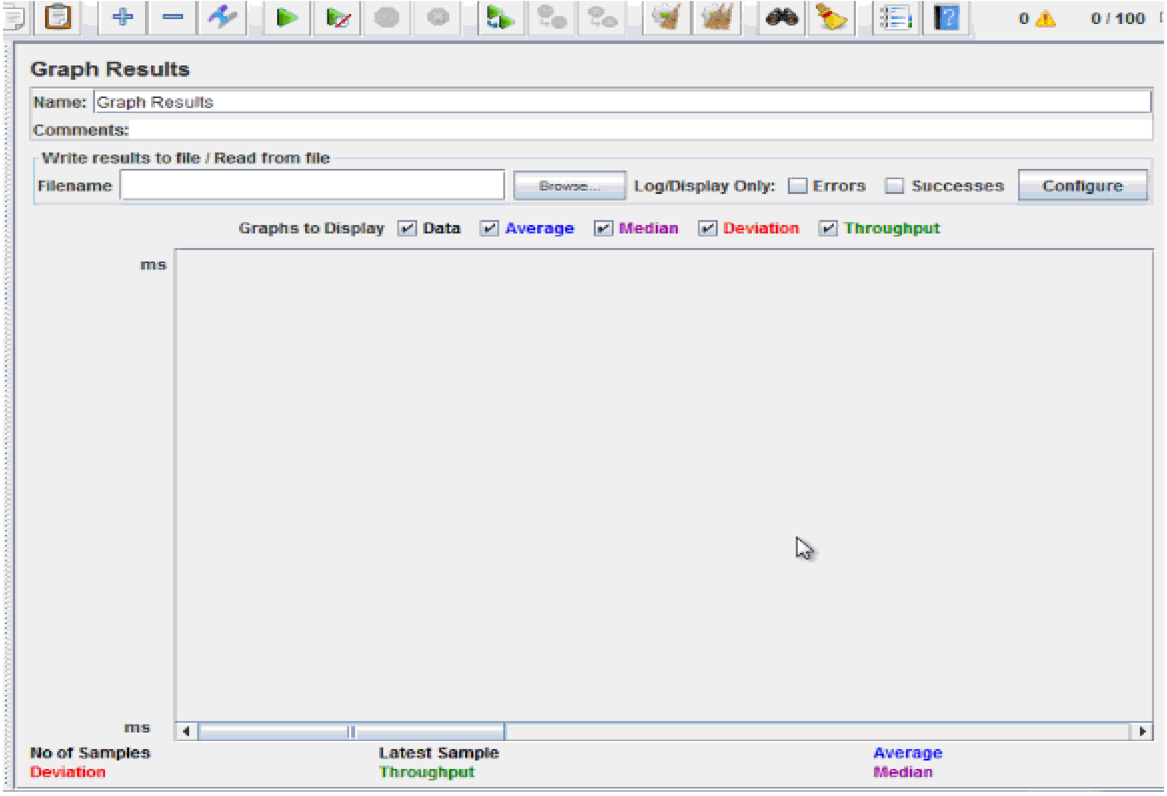
JMeter can show the test result in Graph format.

Right click Test Plan, **Add** -> **Listener** -> **Graph Results**



Step 4) Run Test and get the test result

The picture below presents a graph of a test plan, where we simulated 100 users who accessed on website http://localhost:8080/ThesisImplementationOpenStack/



Graph Results

Name: Graph Results

Comments:

Write results to file / Read from file

Filename

Log/Display Only: □ Errors □ Successes

Graphs to Display ☑ Data ☑ Average ☑ Median ☑ Deviation ☑ Throughput

No of Samples
Deviation
Latest Sample
Throughput
Average
Median

At the bottom of the picture, there are the following statistics, represented in colors:

- Black: The total number of current samples sent.
- Blue: The current average of all samples sent.
- Red: The current standard deviation.
- Green: Throughput rate that represents the number of requests per minute the server handled

Let analyze the performance of http://localhost:8080/ThesisImplementationOpenStack/ in below figure.

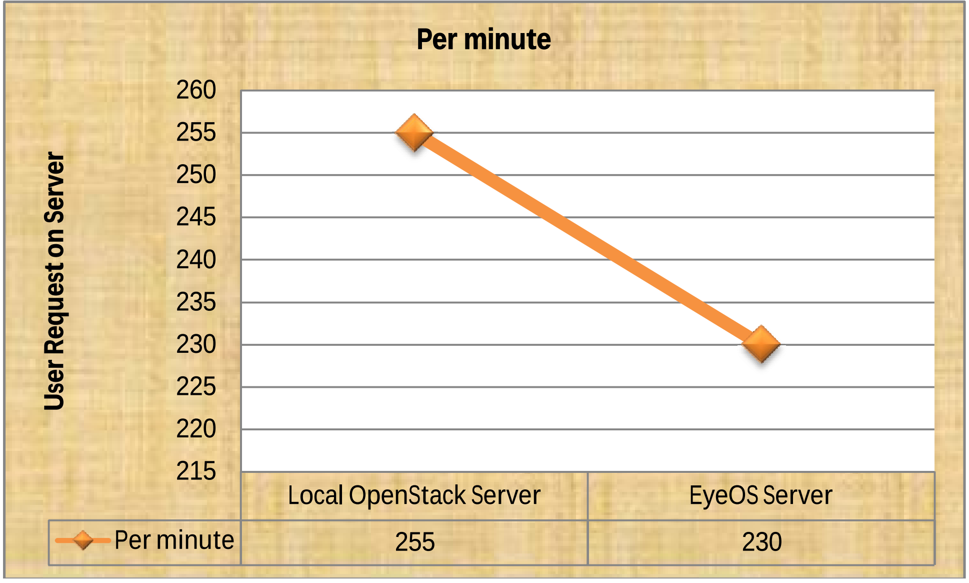
To analyze the performance of the web server under test, we have to focus on 2 parameters

- Throughput
- Deviation

The **Throughput** is the most important parameter. It represents the ability of the server to handle heavy load. The **higher** the Throughput is, the **better** is the server performance.

In this test, the throughput of http://localhost:8080/ThesisImplementationOpenStack/ is 255 /minute. It means Google server can handle 255 requests per minute. This value is quiet high so we can conclude that http://localhost:8080/ThesisImplementationOpenStack/ server has good performance

User Request On Server	Local OpenStack Server	EyeOS Server
Per minute	255	230



The throughput of website under test **eyeos** server is 230/minutes. It means this server handle 230 requests per minute, lower than **http://localhost:8080/ThesisImplementationOpenStack/.**

In comparison with previous work, few enhancements have been observed which are cited below;

1. Previous system by Kim et. al. [9] doesn’t consider confidentiality during storage. They stored file and all user information into plain text format which can be lead for information compromization or interception attack. Here in our work RSA algorithm has been integrated to achieve confidentiality during file storage and information insertion in database.

2. Kim et. al. [9] don't integrate access control model. Here in our work RBAC model has been used to maintain access control and service classification.

3. Kim et. al. [9] don't show the evalution stastics, so actual performance measurement can't be observed. Our work completely consider multiple size file input to observe computation time and memory complexity for both encryption and decryption process.

4. Priyanka Nema has shown the implementation of RSA for communication purpose [15], which increase the tranmission time and complexty of data tranfer. Here our work uses https protocol to make communication secure and RSA to make storage secure.

5. Kim et. al. [9] has used hash algorithm for integrity purpose. Here in our work we replace it by MD5 to enhance the strength of integrity and performance of propsoed solution.

VI. CONCLUSION

The implementation of this project work is to enhance the authentication and access control process in cloud environment. By implementing the OTP as an additional authentication parameter when one user wants to access cloud resource from external location or from public cloud the authetication system is enhanced. By implementing Role Based Access and Control (RBAC) we have provided different privileges to different users, in this way we have implemented a much secure access conrol mechanism. Integrity of the file, which user is uploading on the server is calculated and maintained using MD5 algorith. The confidentiality of the message is maintained by using RSA algorithm. The database keeps all the informaion in encrypted form, hence it will be very hard to intercept the data stored in database.

The complete study concludes that, the existing model not only enhances the security to the public and private cloud but it also provides the same security in the hybrid cloud environment. By implementing the security principle and algorithm we tried to enhance the security of the overall cloud based system.

REFERENCES

[1] Jinbo Xiong, Ximeng Liu, Zhiqiang Yao, Jianfeng Ma, Qi Li, Kui Geng, Patrick S. Chen, A Secure Data Self-Destructing Scheme in Cloud Computing, IEEE Transactions on Cloud Computing, Volume: 2, Issue: 4, Oct.-Dec. 1 2014.

[2] Stephen R. Smoot, Nam K. Tan, Private Cloud Computing: Consolidation, Virtualization, and Service-Oriented Infrastructure, 1st edition, Morgan Kaufmann Publishers Inc.

[3] Tekin Bicer, David Chiu, Gagan Agrawal, Time and Cost Sensitive Data-Intensive Computing on Hybrid Clouds, 2012 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing.

[4] Kehuan Zhang, Xiaoyong Zhou, Yangyi Chen, XiaoFeng Wang, Yaoping Ruan, Sedic: privacy-aware data intensive computing on hybrid clouds, CCS 11, October 17/21, 2011, Chicago, Illinois, USA

[5] P. V. G. D. Prasadreddy, T. Srinivasa Rao, S. Phani Venkat, A Threat Free Architecture for Privacy Assurance in Cloud Computing, Proceeding SERVICES '11 Proceedings of the 2011 IEEE World Congress on Services Pages 564-568

[6] Vijay Varadharajan, Department of Computing, Macquarie University, Sydney, Australia, Security and trust in the web, Proceeding APWeb'12 Proceedings of the 14th Asia-Pacific international conference on Web Technologies and Applications

[7] B.Prasanalakshmi, A.Kannammal, Secure Credential Federation for Hybrid Cloud Environment with SAML Enabled Multifactor Authentication using Biometrics, International Journal of Computer Applications (0975 – 8887) Volume 53– No.18, September 2012

[8] D.E. Popescu, A.M. Lonea, An Hybrid Text-Image Based Authentication for Cloud Services, INT J COMPUT COMMUN, ISSN 1841-9836 8(2):263-274, April, 2013.

[9] Jin-Mook Kim and Jeong-Kyung Moon, Secure Authentication system for hybrid cloud service in mobile communication environment, Hindawi Publishing Corporation International Journal of Distributed Sensor Networks Volume 2014, Article ID 828092, 7 pages http://dx.doi.org/10.1155/2014/828092

[10] Sang-Ho Na, Jun-Young Park, Eui-Nam Huh, Personal Cloud Computing Security Framework, Proceeding APSCC '10 Proceedings of the 2010 IEEE Asia-Pacific Services Computing Conference, Pages 671-675

[11] Hsing-Chung (Jack) Chen, Marsha Anjanette Violetta, Cheng-Ying Yang, Contract RBAC in cloud computing, The Journal of Supercomputing archive Volume 66 Issue 2, Nov 2013 Page 1111-1131 Kluwer Academic Publisher Hingham,MA, USA

[12] John Linkous, Don't Let Hybrid Clouds Rain on Your Security, RSA Conference | Where the world talks security, 4th Sep 2014 http://www.rsaconference.com/blogs/dont-let-hybrid-clouds-rain-on-your-security
[13] T. H. Kim, I. H. Kim, C.W.Min, and Y. I. Yeom, Security technical trend of cloud computing, Computer Science Managing, vol. 30, no. 1, pp. 30–38, 2012
[14] Y. H. Bang, S. J. Jeong, and S. M. Hwang, Security requirement development tools of mobile cloud system, Information and Communications, vol. 28, no. 10, pp.19–29, 2011
[15] Priyanka Nema, An Innovative Approach for Dynamic Authentication in Public Cloud: Using RSA, Improved OTP and MD5, International Journal of Innovative Research in Computer and Communication Engineering, Vol. 2, Issue 11, November 2014
[16] Parekh, T ; Gawshinde S ; Sharma M K “ Token based authentication using mobile phone” published in procedding of Communication Systems and Network Technologies (CSNT) 3-5 June 2011, Jammu, India