International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 4 Issue 03, March-2015

A Secure Cloud Storage Virtualization Model And user Authentication using Threshold Kerberosv5 Scheme

Kavery PM
M.tech CSE,
Nitte Meenakshi Institute Of Technology
Bangalore 560064

Afroz Pasha
Asst. Prof, CSE(UG)
Nitte Meenakshi Institute Of Technology
Bangalore 560064

***Abstract* - The enormous growth in Information technology industry has led to the growth of server virtualization but to build this virtualization storage server lot of cost is involved. The cost not only involves investing in technology but also maintaining these technologies. This led IT industry to virtualize storage to cloud. Cloud provides resource to user on demand. It is highly scalable, flexible and platform in dependable. Though it provides service on demand the data stored in cloud replica has constrained on security issue like attacks, data loss, other authentication and security issues. Here in this paper we propose a new authentication scheme which improves the existing security loop holes .Here we propose an authentication model by using kerberosV5 web service with threshold cryptography for Amazon S3 cloud. The Kerberos web service which performs authentication acts as a third party between user and Amazon cloud. Our proposed model reduces the overhead of Amazon cloud for doing authentication check thus improving the performance and it is highly secured and efficient.**

Keyword: Cloud computing, Kerberos web service, Amazon, cryptography.

I .INTRODUCTION

Cloud computing represents a computing paradigm where computing resources are not physically present at user's location. These resources, usually collectively called clouds, are owned and managed by cloud service providers and can be accessed by end users remotely via the Internet. The past few years have witnessed a rapid shift of computing from the desktop to the cloud. With the rapid advancement of both wireless network technologies and mobile smart phones, there is an increasing need for cloud services to be provided to mobile users Cloud computing is a large-scale distributed network system implemented based on a number of servers in data centers. The cloud services are generally classified based on a layer concept. In the upper layers of this paradigm, Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) are stacked.

- *Infrastructure as a Service (IaaS):*

IaaS is built on top of the data center layer. IaaS enables the provision of storage, hardware, servers and networking components. The client typically pays on a per-use basis. Thus, clients can save cost as the payment is only based on how much resource they really use. Infrastructure can be expanded or shrunk dynamically as needed. The examples of IaaS are Amazon EC2 (Elastic Cloud Computing) and S3 (Simple Storage Service).

- *Platform as a Service (PaaS):*

PaaS offers an advanced integrated environment for building, testing and deploying custom applications. The examples of PaaS are Google App Engine, Microsoft Azure, and Amazon Map Reduce/Simple Storage Service.

- *Software as a Service (SaaS):*

SaaS supports a software distribution with specific requirements. In this layer, the users can access an application and information remotely via the Internet and pay only for that they use. Salesforce is one of the pioneers in providing this service model. Microsoft's Live Mesh also allows sharing files and folders across multiple devices simultaneously.

A. Kerberos authentication service:

Kerberos is an authentication mechanism that provides a secure means of authentication for network users. It prevents transmission of clear text passwords over the network by encrypting authentication messages between clients and servers. In addition, Kerberos provides a system for authorization in the form of administering tokens, or credentials. In the Kerberos authentication system is based on a trusted third party authentication model. The Key Distribution Center (KDC) serves as a repository of symmetric keys which are used to cryptographically validate users and the devices or hosts which they access. The designers of the Kerberos protocol anticipated the potential need for authorization and implemented an optional payload field for carrying authorization information. The format and specification of this field was deliberately left undefined. The only other form of authorization implemented by Kerberos is a system for defining whether or not authentication from foreign realms will be accepted. Another define Kerberos is an authentication protocol for trusted hosts on untrusted networks.

B. Third party auditor:

The third party defines who has the correctness, expertise, capabilities to access and utilize the cloud service provider. They configure the service access policy for users.

C. Key distribution center (KDC).

The KDC comprises of authentication server (AS), Ticket Granting server(TGS) and a centralized database. The KDC is a trusted entity. The KDC is aware of the secret keys of all the entities (client, server). The secret keys are hashed and stored in the centralized database.

D. Authentication service:

Authentication service that know the password of all user and stores these in a centralized database. In addition, the AS shares a unique secret key with each server.

E. Tickets granting service:

TGS provide and issue tickets to user who have been authenticated to AS.

F. Cloud service provider:

Cloud service providers offer cloud solutions, like Google Apps, that are delivered electronically over the internet. Unlike a managed service provider, cloud service providers do not sell or install hardware everything they offer is stored online and accessible securely from anywhere. There are many advantages to working with a cloud service provider like Cloud Sherpa when switching from your old email and collaboration software.

Firstly we have defined and introduced the users, their attributes, and their tasks. Secondly, we have introduced one application program as the third party auditor. Third, we surveyed the Kerberos with threshold cryptography effect in cloud computing server. Finally, we examined the cloud server provider. Kerberos uses strong encryption and a complex ticket-granting algorithm to authenticate users on a network. Also, since many users are interested in Kerberos, it has the ability to distribute "session keys" to allow encrypted data streams over an IP network. Users who want to connect to the cloud, at first, they should make the profile and user ID in the third party. The information of all users such as User ID and hashed password will be saved in the database for more secure. After the registration in the third party, it must get the password and user ID. In the next race, it should be connected to the Kerberos real and do this process. Send the Request for ticket granting ticket to the As. As verifies user's access right in data base, create ticket-granting ticket and session key. Results are encrypted using key derived from user password. User will send the request cloud service granting ticket to TGS. The TGS will send the Ticket + session key to the user. (It executes one per type of service).Workstation sends ticket and authenticator to cloud server provider. Server verifies ticket and authenticator match, then grant access to service. By using Shamir's Secret Sharing algorithm with Kerberos these limitations can be removed. Shamir's Secret Sharing is an algorithm in cryptography where a secret is breaked into parts, distributing each participant its own unique part, where some of the parts or all of them are needed in order to reconstruct the secret. Considering on all participants to combine together the secret might be impractical, and therefore the threshold Cryptography scheme is used where any k of n parts are sufficient to reconstruct the original secret. Due to some limitations of Kerberos we are using a Threshold cryptography algorithm to avoid single point of failure. The combination of these two algorithms will give a more secure authentication system.

II.LITERATURE SURVEY

M. Lillibridge et al. Here they present a novel peer-to-peer backup technique that allows computers connected to the Internet to back up their data cooperatively: Each computer has a set of partner computers, which collectively hold its backup data. In return, it holds a part of each partner's backup data. By adding redundancy and distributing the backup data across many partners, a highly-reliable backup can be obtained in spite of the low reliability of the average Internet machine. A. Juels et al. Here they define and explore proofs of retrievability (PORs). A POR scheme enables an archive or back-up service (prover) to produce a concise proof that a user (verifier) can retrieve a target file F, that is, that the archive retains and reliably transmits file data sufficient for the user to recover F in its entirety. G. Ateniese et al. Here they introduce a model for provable data possession (PDP) that allows a client that has stored data at an untrusted server to verify that the server possesses the original data without retrieving it. The model generates probabilistic proofs of possession by sampling random sets of blocks from the server, which drastically reduces I/O costs. The client maintains a constant amount of metadata to verify the proof. The challenge/response protocol transmits a small, constant amount of data, which minimizes network communication.

H. Shacham et al. In a proof-of-retrievability system, a data storage center convinces a verifier that he is actually storing all of a client's data. The central challenge is to build systems that are both efficient and provably secure that is, it should be possible to extract the client's data from any prover that passes a verification check. Here we give the first proof-of-retrievability schemes with full proofs of security against arbitrary adversaries in the strongest model, that of Juels and Kaliski. Our first scheme, built from BLS signatures and secure in the random oracle model, has the shortest query and response of any proof-of-retrievability with public verifiability. Our second scheme, which builds elegantly on pseudorandom functions (PRFs) and is secure in the standard model, has the shortest response of any proof-of-retrievability scheme with private verifiability (but a longer query). Both schemes rely on homomorphic properties to aggregate a proof into one small authenticator value. G. Ateniese, R. D. Pietro et al. Storage outsourcing is a rising trend which prompts a number of interesting security issues, many of which have been extensively investigated in the past. However, Provable Data Possession (PDP) is a topic that has only

recently appeared in the research literature. The main issue is how to frequently, efficiently and securely verify that a storage server is faithfully storing its client's (potentially very large) outsourced data.

K. D. Bowers et al. Here they introduce HAIL (High-Availability and Integrity Layer), a distributed cryptographic system that permits a set of servers to prove to a client that a stored file is intact and retrievable. HAIL strengthens, formally unifies, and streamlines distinct approaches from the cryptographic and distributed-systems communities. Proofs in HAIL are efficiently computable by servers and highly compact— typically tens or hundreds of bytes, irrespective of file size. Cong Wang et al. Cloud Computing has been envisioned as the next generation architecture of IT Enterprise. In contrast to traditional solutions, where the IT services are under proper physical, logical and personnel controls, Cloud Computing moves the application software and databases to the large data centers, where the management of the data and services may not be fully trustworthy. This unique attribute, however, poses many new security challenges which have not been well understood. In this article, we focus on cloud data storage security, which has always been an important aspect of quality of service. To ensure the correctness of users' data in the cloud, we propose an effective and flexible distributed scheme with two salient features, opposing to its predecessors. By utilizing the homomorphic token with distributed verification of erasure-coded data, our scheme achieves the integration of storage correctness insurance and data error localization, i.e., the identification of misbehaving server(s). Unlike most prior works, the new scheme further supports secure and efficient dynamic operations on data blocks, including: data update, delete and append. Extensive security and performance analysis shows that the proposed scheme is highly efficient and resilient against Byzantine failure, malicious data modification attack, and even server colluding attacks.

III.PROPOSED SYSTEM

This paper emphasizes on authenticating a client before gaining access to the Amazon cloud service. Just the local credentials (Amazon web service credentials) are not adequate to authenticate users in Amazon cloud computing environment which is distributed and shared. Kerberos is a distributed environment authentication protocol based on symmetric key cryptography and provides mutual authentication, 3 levels of protection and single sign on. Kerberos was developed in the mid 1980's.It is upgraded to many versions since it came into existence. The latest upgraded version is Kerberos version 5. The main Control node at cloud acts as interface between cloud and client. Control node receives the requests from clients and must check each client for identification.

A. Kerberos Web Service

Here, Kerberos web service is used to authenticate the client who wants to access the applications at the Amazon cloud server. Other reasons for using Kerberos web service is that it provides mutual authentication, single sign-on, three tier protection, limits the memory usage and the computational burden of Amazon cloud service. Awareness of authenticity of user and server to each other is known as Mutual authentication. Fig 1 shows main components of Kerberos web service and its transaction of messages between the components.

Whenever a user wants to access a service from the Amazon Cloud Server it requires a Kerberos web service ticket. Only on the basis of that ticket, Amazon cloud server will grant access to all the subscribed services to the client. The users can create bucket, delete buckets, view the contents of bucket, upload all kinds of files, download files and also delete the files in a particular bucket. Ticket proves the client's authentication to Amazon cloud server. Since, the authentication is done by the Amazon Cloud server, the computational overhead used to authenticate the client and the memory usage burden of the Amazon cloud is minimized. To get ticket client sends an authentication request to KSWTC. The Authentication Server of the KSWTC creates a "session key" based on the client's Amazon web service credentials (AWS credentials). The session key is adequately a KSWTC "Ticket Granting Ticket" that will be used by the client to get The session key is adequately a "Ticket Granting Ticket" that will be used by the client to get master ticket (opener ticket) to access services from the Amazon server.

Ticket Granting Ticket performs a ticket exchange to retrieve service granting Ticket. Client next sends the Ticket Granting Ticket to a KSWTC Ticket Granting Server (TGS). In traditional Kerberos Authentication Model there is only one TGS, so if the opener key(master key) sent by TGS is known by someone, then one who is not authorized can use the services provided by the cloud server. The security of data and the Amazon cloud is compromised.

To avoid single point failure of Kerberos and the compromise of the KDC we implement a threshold cryptography algorithm to the TGS of KSWTC. Through this algorithm instead of single KSWTC TGS, multiple TGS (m) have been used where at least the threshold k number of parts are needed to decrypt the opener key (k<m). Client sends a request for opener key to k number of TGS of the KSWTC. If k number of TGS reply, then the client can get the required master key otherwise client will send the request to TGS [k+1] and wait for reply. This process will continue until at least k number of TGS will not reply. After getting the opener key client can request the required service from the Amazon cloud.

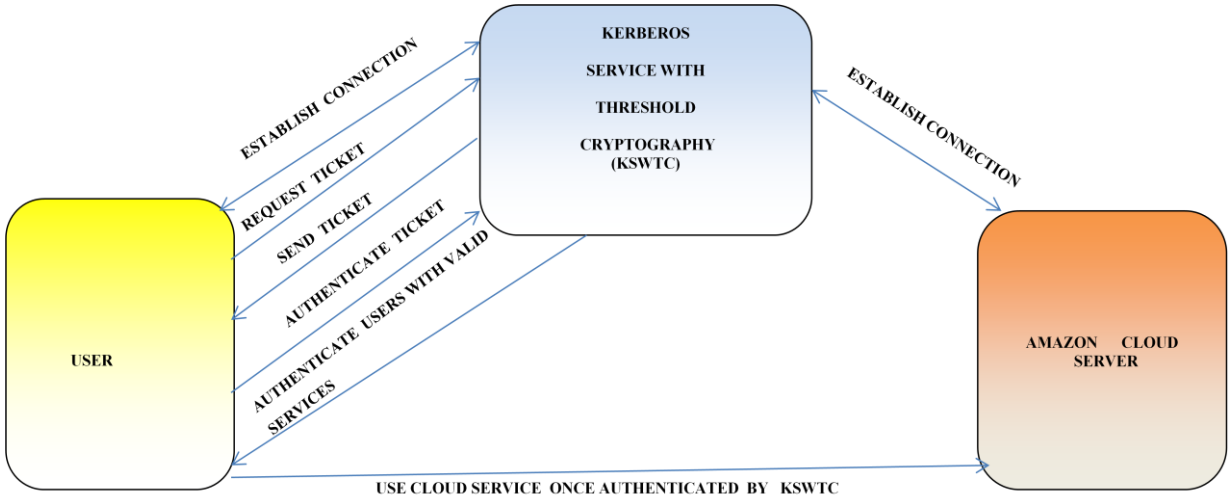

Fig. 1: Kerberos using Threshold Cryptography

The Amazon server either rejects the ticket or accepts it and performs the service. The opener key granted to the client can only be decrypted by the Amazon cloud server with the secret key shared between the cloud server and TGS. Authenticated client or anybody else will never be able to decrypt the opener ticket or the master ticket .Since the ticket which the client has received from the TGS is time-stamped, it allows the client to make additional request using the same ticket within a certain time period (session)

Without the need to prove the authentication again and thereby provides single sign-on facility. As the ticket is valid for a limited period of time, this makes fewer chances that anyone else will be able to use it later and thus ensures another layer of protection. A flow chart for understanding the working of new authentication model is described in Fig. 2 below.

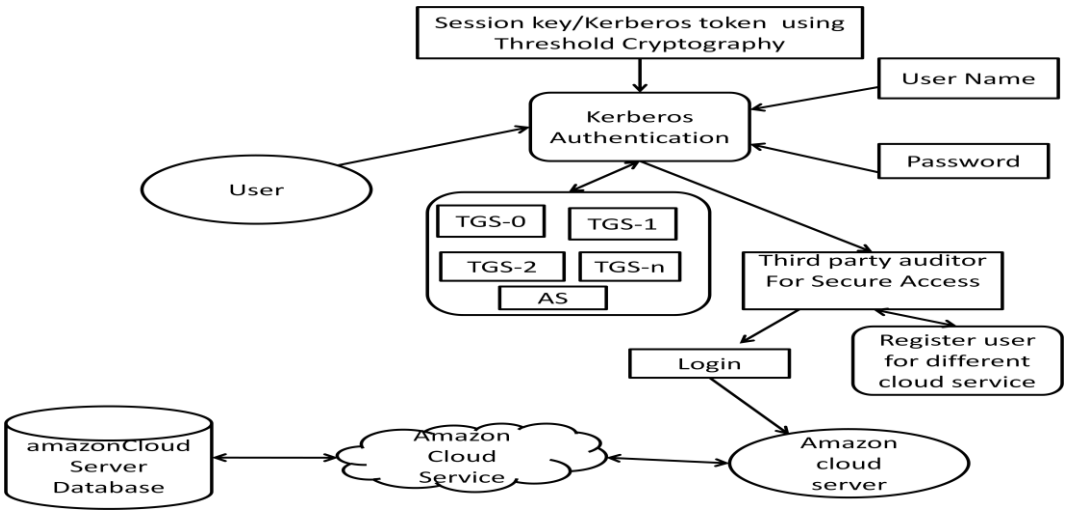

Fig.2: Working model of KSWTC

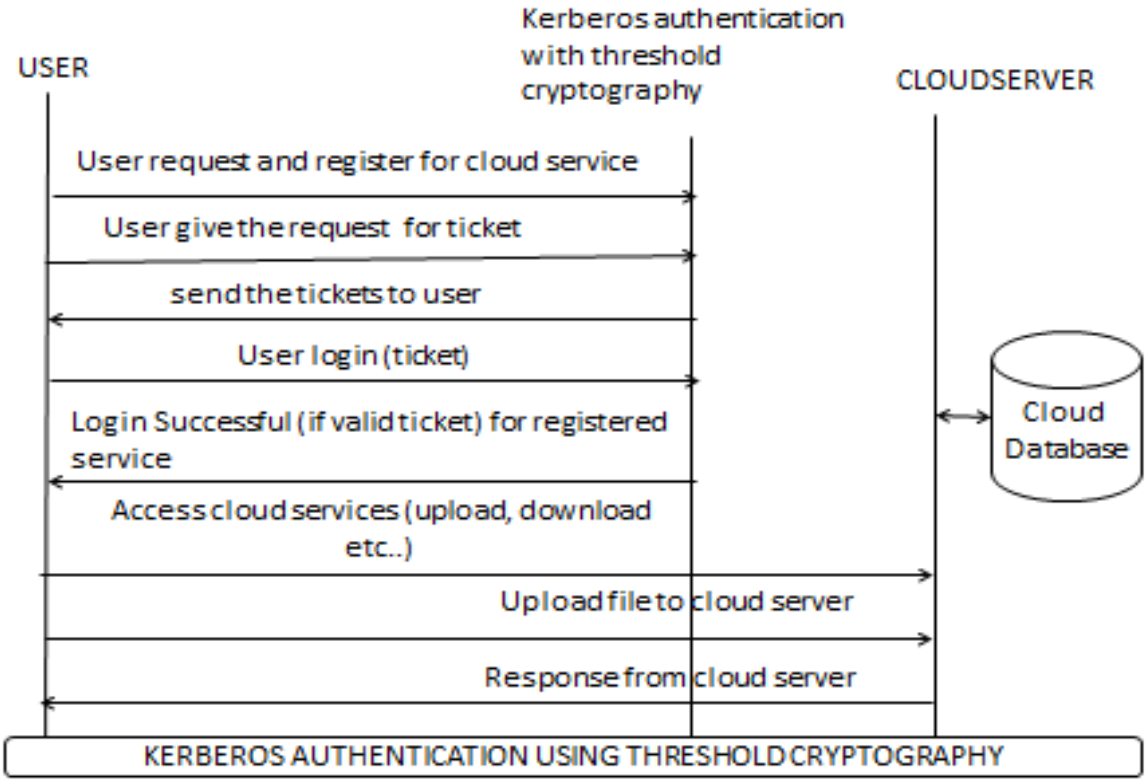

Fig 3: A sequence diagram for describing the new authentication model.

IV.RESULT

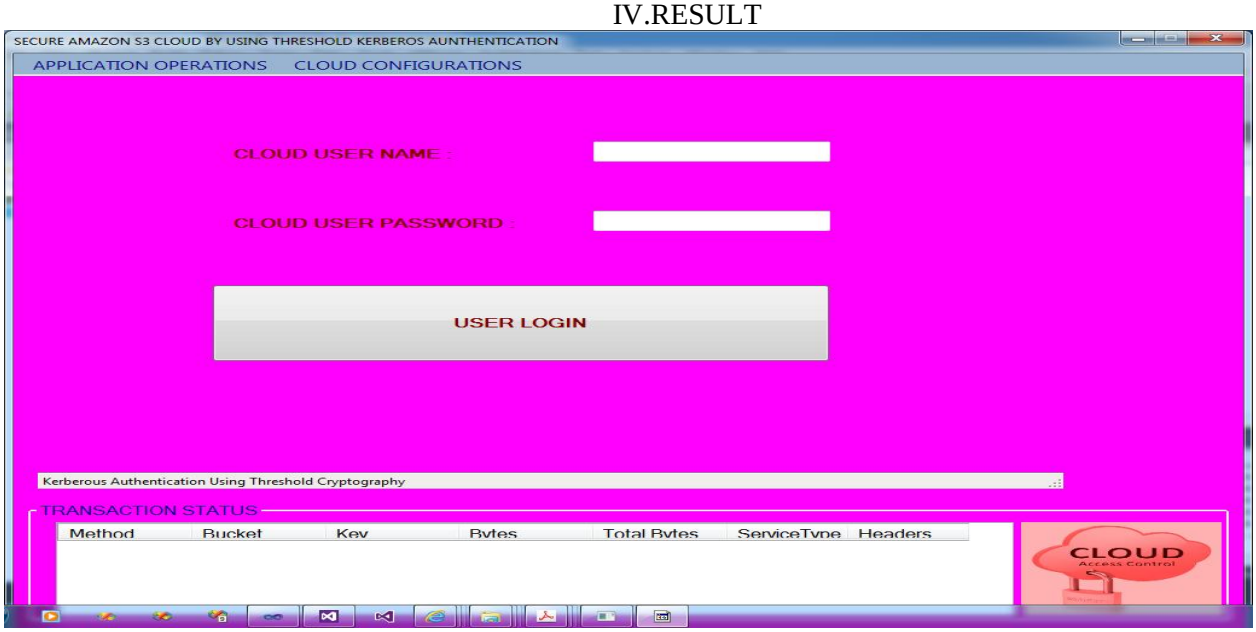

Fig.4: Login Page Using Kerberos with threshold cryptography

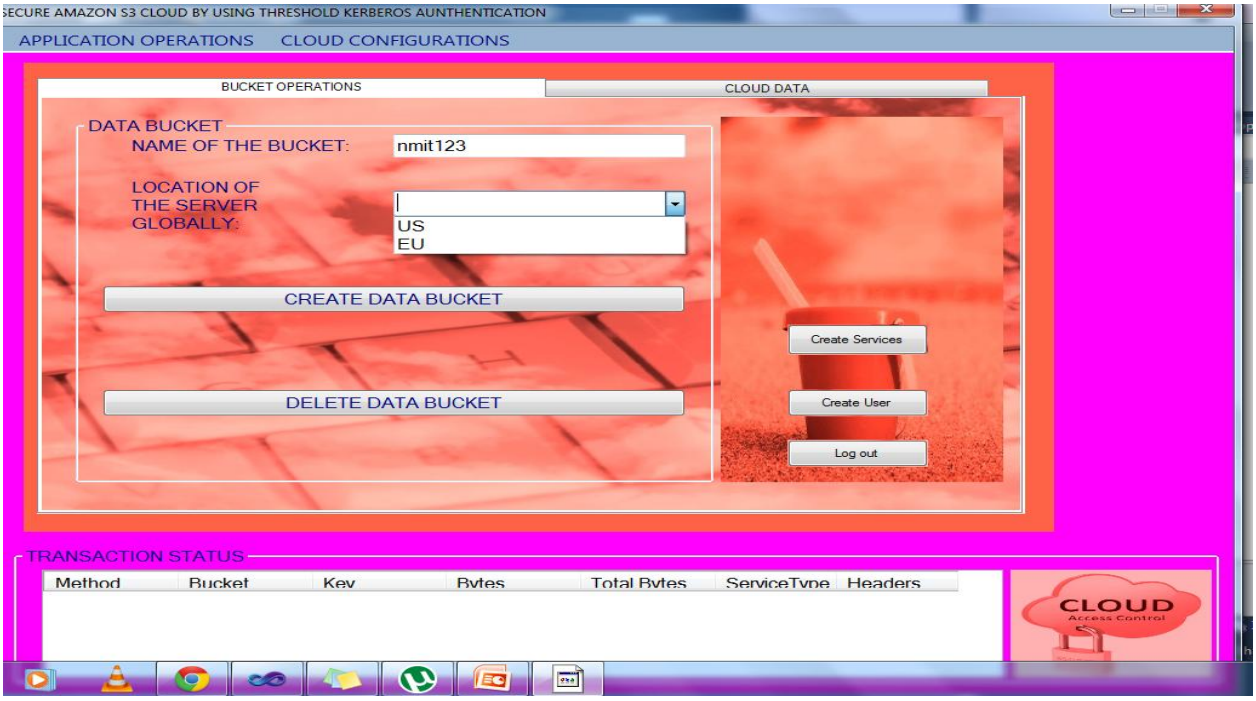

Fig.5: Admin Page

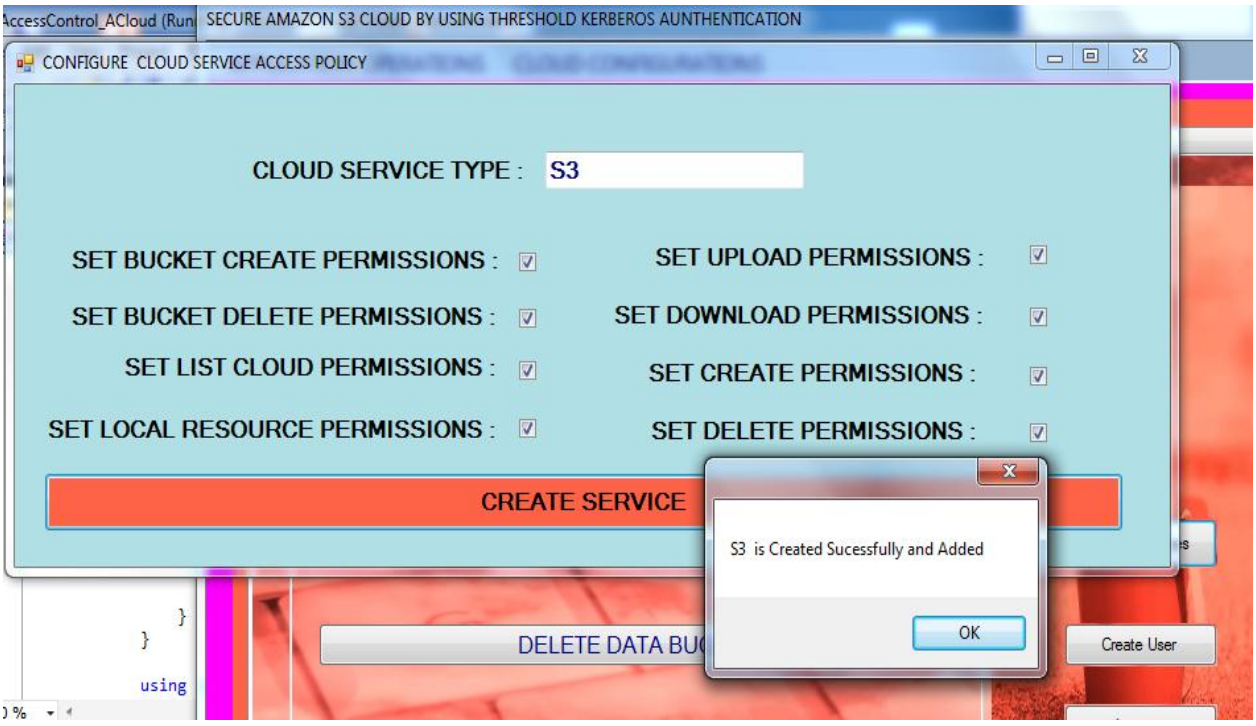

Fig. 6: Configuring service access policy

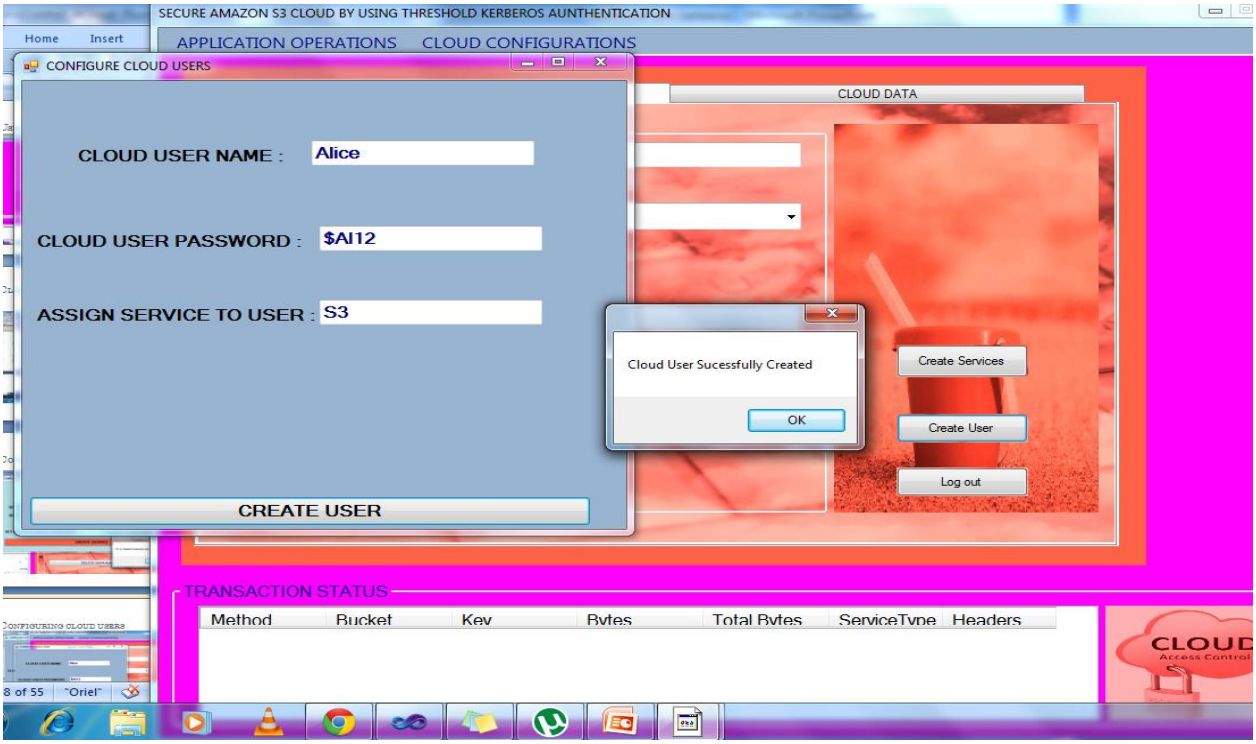

Fig. 7: Configuring cloud users

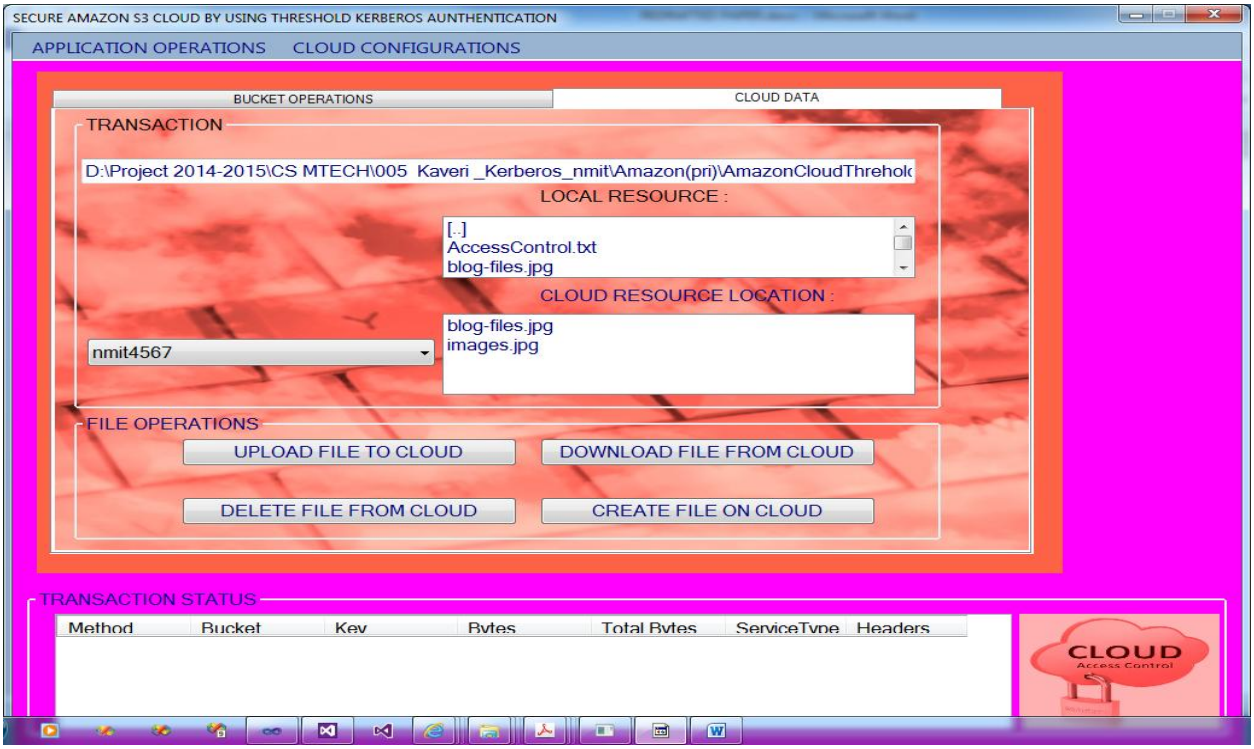

Fig 8. Service Available for Amazon s3 cloud

V.CONCLUSION

Here we discussed the existing security scheme and the need for stronger authentication scheme for cloud environment. Here in this paper we proposed an efficient authentication scheme by using kerberosV5 with threshold cryptography which provide enhanced security and overcomes the limitation of single point failure of Kerberos and also increases the performance of Amazon cloud by eliminating the computational burden and memory usage. Our scheme not only authenticates the user but also authenticates user with type of service he has been permitted with. By configuring the service access policy for a user and authenticating the user with the configured service access policy our proposed scheme is highly efficient. In future we would also Provide more security and also test this model with other cloud provider such as azure, Google etc… and check how it performs.

REFERENCES

1. Mehdi Hojabri, K.venkat rao "Innovation in Cloud Computing: Implementation of Kerberos version5 in cloud computing in order to enhance the security issues" IEEE, International Conference on Information Communication and Embedded System (ICICES), 2013, pp 34-45.
2. Jeong-Kyung Moon, Jin-Mook Kim and Hwang-Rae Kim, "A Secure Authentication Protocol for Cloud Services", Journal of Advanced Information Technology And Convergence, 2012, pp 33-36.
3. Jeffrey Lok Tin Woo, and Mahesh V., "Tripunitara Composing Kerberos and Multimedia Internet Keying (MIKEY) for Authenticated Transport of Group Keys" , IEEE, Transactions on Parallel and Distributed Systems, 2013, pp 1-11.
4. X. Zhang, H. Du, J. Chen, Y. Lin, L. Zeng "Ensure Data Security in Cloud Storage" International Conference on Network Computing and Information Security,2011, pp. 284-287.
5. Rosa Sánchez, Florina Almenares, Patricia Arias, Daniel Díaz-Sánchez, Andrés Marín "Enhancing Privacy and Dynamic Federation in IdM for Consumer Cloud Computing", IEEE, Transaction on Consumer Electronics, 2012, pp 95-103.
6. M. R. Tribhuwan, V. A. Bhuiyar, S. Pirzade "Ensuring Data Storage Security in Cloud Computing through Handshake based on Token Management" IEEE, International Conference on Advances in Recent Technologies in Communication and Computing,2010, pp. 386-389.
7. Ming-Huang Guo, Horng-Twu Liaw, Li-Lin Hsiao, Chih-Yuan Huang, Chih-Ta Yen , "Authentication Using Graphical Password in Cloud", IEEE, 15th International Symposium on Wireless Personal Multimedia Communications, 2012, pp 177-181.
8. Sanjeev Pippal, Vishu Sharma, Shakti Mishra, D.S.Kushwaha , "An Efficient Schema Shared Approach for Cloud based Multitenant Database with Authentication & Authorization Framework", 2011, IEEE, International Conference on P2P, Parallel, Grid, Cloud and Internet Computing, pp 213-218.
9. W. Itani, A. Kayssi. A. Chehab "Privacy as a Service: Privacy-Aware Data Storage and Processing in Cloud Computing Architectures" Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing,2009, pp. 711-715
10. Sushmita Ruj∗, Milos Stojmenovic†, Amiya Nayak, "Privacy Preserving Access Control with Authentication for Securing Data in Clouds", 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, 2012, pp 556-563.
11. Lifei Wei, Haojin Zhu, Zhenfu Cao, Weiwei Jia and Athanasios V. Vasilakos, "SecCloud: Bridging Secure Storage and Computation in Cloud", IEEE 30th International Conference on Distributed Computing Systems. 2010, pp 52-61.
12. Hassan Takabi, James B. D. Joshi, Gail-Joon Ahn, "Secure Cloud: Towards a Comprehensive Security Framework for Cloud Computing Environments", 34th Annual IEEE Computer Software and Applications Conference Workshops,2010, pp 393-398.
13. Jinyue Xia and Yongge Wang, "Secure Key Distribution for the Smart Grid", IEEE, Transactions on Smart Grid, 2012, pp 1437-1443.
14. C.C. Tan, Q. Liu, and J. Wu, "Secure Locking For Untrusted Clouds", 4th IEEE International Conference on Cloud Computing, 2011, pp. 131-138.
15. Farhan Bashir Shaikh and Sajjad Haider, "Security Threats in Cloud Computing", Sixth IEEE International Conference on Internet Tehnology and Secure Transaction, pp 120-126.
16. Ashish G. Revar, Madhuri D. Bhavsar, "Securing User Authentication using Single Sign-On in Cloud Computing", IEEE, International Conference On Current Trends In Technology, 2011, pp 1-4.
17. Zubair Ahmad and Jamalul- Lail AbManan, "Trusted Computing based Open Environment User Authentication Model", 3rd IEEE, International Conference on Advanced Computer Theory and Engineering ,IEEE,2010,pp 487-491.