An Approach to Secure Access to Cloud Storage Service

Jyoti Shetty; Anala M R & Shobha G
R V College of Engineering, Banglore, India
sjyothi.12@gmail.com; shobhag@rvce.edu.in; analamr@rvce.edu.in

Abstract:

Cloud computing is an agglomeration of technologies such as operating systems, network, data storage and virtualization with inherent security issues such as data theft, eavesdropping, infrastructure misuse and so on. Because of these security concerns sectors such as banking, defense, healthcare and finance are hesitant to use cloud services and hence are deprived of its advantages. This work is an attempt to provide users a secure access to the cloud services. It addresses authentication, confidentiality and data integrity. The prototype is implemented and tested.

Keywords:
Cloud Security; Homorphic encryption.

I. Introduction

Cloud computing is defined as a computing model which facilitates convenience, on-demand access to a distributed pool of configurable computing resources such as infrastructure, applications and platform. It enhances computing services by enabling users to access software applications and computing services which are not stored locally, i.e. in user's system. These services are provided over the internet and the service model follows as that of utility services i.e the user pays for what he/she uses. Enterprises can also request IT services without having to purchase large amount of hardware, and invest for deployment and management of such resources. Cloud computing is beneficial to various industries and individuals which they access via internet. Services such as Consulting, Management, Financial and Data Storage Services can be provided among numerous other beneficial services [1].

The recent insights into this technology have explored various critical aspects of security. Various categories of such security concerns are trust, architecture, identity management, software isolation, data protection, confidentiality and availability[2]. All these security vulnerabilities lead to various threats on the cloud such as authentication, misuse of cloud infrastructure, eavesdropping, network intrusion, denial of service attack, session hijacking [2]. Further Cloud Forensic is an emerging challenge related to cloud security[3].

Although cloud computing has received lot of attention from the users because of its utility model but its implementation is facing lot of problems at security front. Security issues in cloud computing jeopardize confidentiality, integrity and availability. The cloud service provider(CSP) has complete control of the data stored in cloud and the consumer cannot trust the CSP for the security. There is threat to the confidentiality, integrity and availability of the data stored at cloud[4][5]. As the data is at the CSP premises it is visible to CSPs personnel who has access to it, thus threat to the confidentiality of the data. Also during uploading or downloading of data from the cloud any unauthorized parties can eavesdrop or modify the data-in-transit performing man-in-middle attack thus threat to confidentiality and integrity of data. Because of these security concerns consumers are hesitant to use the cloud services. Thus it is important to have secure access to cloud services by having a system for secure access to cloud which

includes security measures for protecting confidentiality, integrity and authorized access.

II. Implementation

This manuscript presents an approach to secure access to cloud services. A software is developed which provides secure access to the cloud services. The measures of security can largely be solved by making sure the access is provided only to the genuine user and no other malicious user is able to access the resources. Confidentiality is provided by encrypting the client data in the client side before transferring it to the cloud. It uses a secret key using Triple-DES technique[6]. Thus ensuring that the data-in-transit is encrypted and hence secure against man-in-middle attack. Further the data is again encrypted by the CSP before storing thus providing double encryption. Integrity is provided by calculating message authentication code using Secure Hash Algorithm(SHA)[7]. All these measure make sure the client data is secure and is not susceptible to eavesdropping during transmission and is not visible to CSP. The system further ensures that CSP does not tamper with the usage data using Homomorphic encryption[8][9][10]. It uses homomorphic encryption to perform secure computation and store the encrypted data.

The system developed includes:

- User registration and authentication interface as shown in figure1.
- E-mail notification on successful registration and password generation as shown in figure 2.
- Authorization of registered users to access the cloud service - Exclusive client software is designed for the users. This rules out the possibility of distributed denial of service attack on the cloud instance.
- Encryption for the data in both user and the cloud side.
- Homomorphic encryption of user billing. This acts a trust mechanism for the user in CSP.
- Message authentication to check the integrity of the user data in the cloud.
- After successful registration a storage is created using Amazon S3's buckets as shown in figure 3.

Figure 1: User registration form

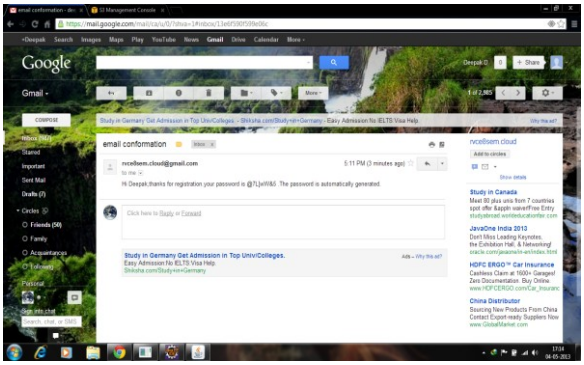

Figure 2: E-mail confirmation and automated password generation

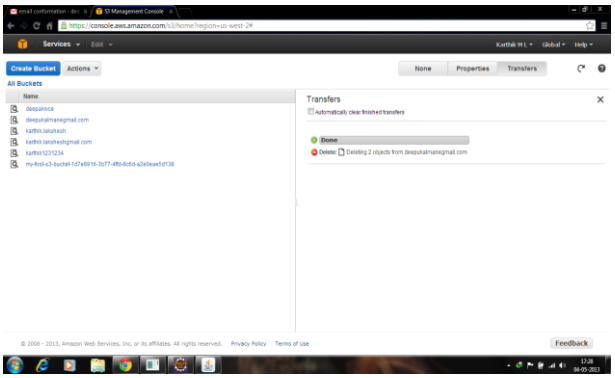

Figure 3: Creation of S3 Bucket after registration

Amazon Web Service(AWS)[11] is a bunch of tools and services available for setting a cloud. A user is provided with all

the tools that might be needed for setting up the cloud. For using the services, the user has to register for the service on aws.amazon.com. After the registration the user is provided with a portal where he can manage the services which he wishes to use. We have used Amazon Elastic Compute Cloud and Amazon Simple Storage Service.

- Amazon Elastic Compute Cloud (EC2) - This service is an IaaS form of service in the cloud. A virtual instance of an operating system is provided. User can deploy and run applications. We have chosen Ubuntu 12.04 as an EC2 instance. A public DNS is provided for this instance from which it's possible to access the applications deployed on the instance.
- Amazon Simple Storage Service (S3) – This service is Data Storage as a Service provided by Amazon Web Services. A storage entity called bucket is created which can be used to store files. We will be creating a bucket to each user.

Amazon EC2 instance is started in the AWS Portal. T1.Micro is the name of the configuration that we have made use of. It has 700MB of RAM and 8GB of Elastic Block Storage (EBS) volume. A key-pair is generated to obtain the access to the machine. EBS ensures that the data in the volume is persistent. When the EC2 instance is started, a public DNS is allocated for the instance. This is used to connect to the instance.

A Secure Shell (SSH) client is used to connect to the instance. It requires the key-pair file and the public DNS. We have used the native SSH client present in Ubuntu running in our system. After connection is established we will be able to use the commands to control the instance.

The instance is used to deploy the server program which will be running all time as shown in figure 4. All the necessary configurations would be done for the program to run. user files would be encrypted using Triple-DES before upload. A message signature is also calculated which is sent along with the upload file. By this method integrity of the file can be checked. Files from the users are stored in Amazon Simple Storage Service(S3); before they are stored file is again encrypted using DES by AWS S3 as shown in figure 6. Upon download the signature of the file is calculated and is matched to the one which was generated when the file was uploaded to check the integrity of the data. If the data is integrate it is decrypted and stored at the user system.

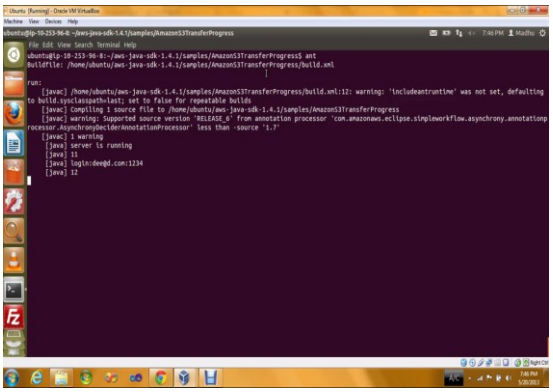

Figure 4: File Upload by user

Consumer is given a client program from which he can upload/download their files as shown in figure 5. This is interface for the user. User credentials are managed to the ones that are given during the registration. If an unknown user attempts to make unsuccessful attempted to get into the system, it is considered an intrusion and that IP will be blocked to safeguard the system. For uploading process based on the user action a file is selected for upload. It is encrypted in the client program before transmission to the EC2 Instance. MAC process generates the message digest which is also sent along with file to the server which checks for integrity of data before storing. The integrate data is forwarded to the cloud storage service, where encryption is done again by AWS S3. This doubly encrypted file is stored in the storage service. For downloading the doubly encrypted file is fetched from the file storage service, which is then decrypted by AWS S3. It is sent back to the user system where it is again decrypted to obtain the

original file. MAC is generated and verified to check the integrity of the data. The decrypted file is stored ate selected location in the user system as shown in figure 7.

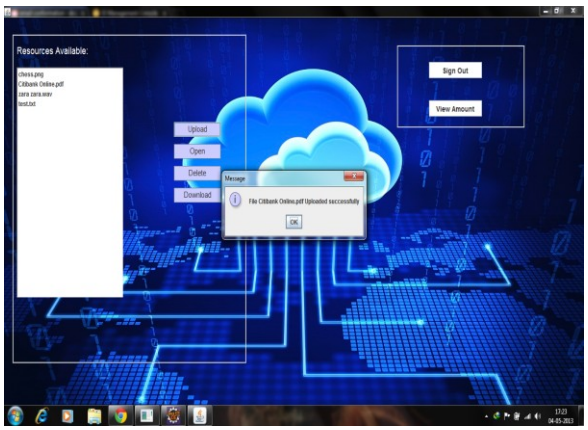

Figure 5: File Upload by user

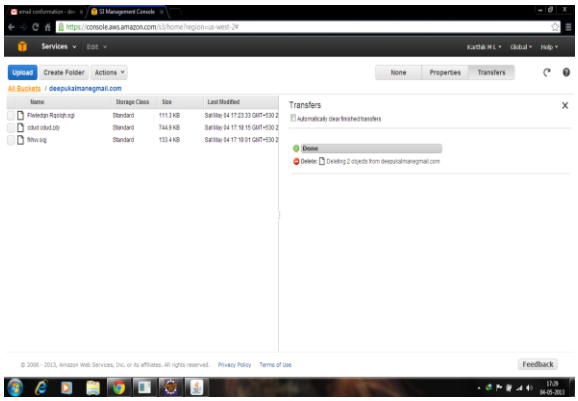

Figure 6: Double encrypted in S3

Further a module that manages the client usage data is constructed. This module calculates the client usage billing data and stores it in encrypted form. The addition is also done in encrypted form using Homomorphic encryption. Homomorphic encryption is provided using Paillier encryption method. The previous usage data is stored in encrypted form in the MySQL database. The usage data of current session is encrypted in the client side and is sent to the server where it is added to the previously stored value and the database value is updated. This addition is done in cipher form and thus is not visible to the Cloud service provider. This is shown in figure 8. This acts as trust mechanism for user as he can verify whether the CSP is billing right or not.

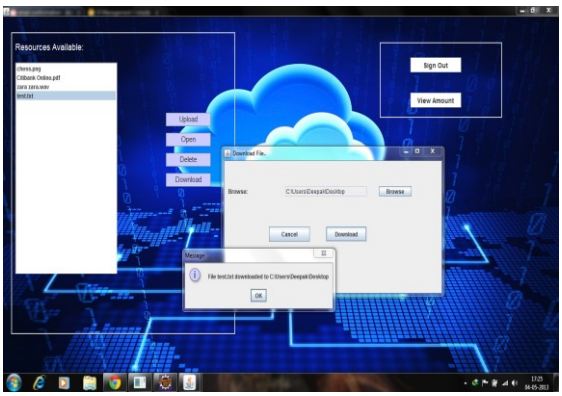

Figure 7: File downloaded successfully to user system

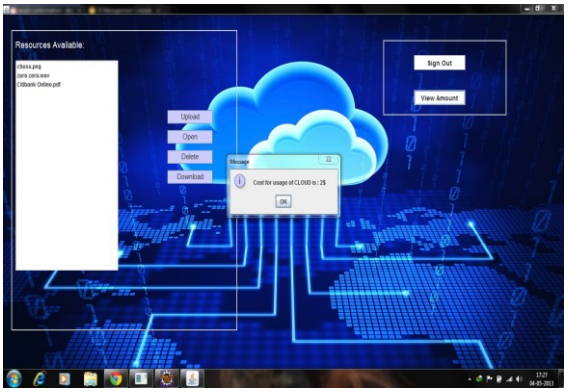

Figure 8: Usage cost computation using secure Homomorphic encryption

III. Conclusion

Thus a prototype software that can provide secure access to cloud services id implemented and demonstrated. Although the file size increases by 30.6122% at the end and a processing time of 0.4392s/MB is required for the encryption, which also increases the transmission time, it is acceptable when the security provided by the software service is considered.

IV. References

[1] Wayne A. Jensen, NIST, "Cloud Hooks: Security and Privacy Issues in Cloud Computing", Proceedings of the 44th Hawaii International Conference of System Sciences 2011, Hawaii, pp 1-6.

[2] Ashish Agarwal, Aparna Agarwal , "The Security Risks Associated with Cloud Computing", International Journal of Computer Applications in Engineering Sciences, Volume 1,

Special Issues on CNS, July 2011, ISSN- 2231-4946, pp 257-259.

[3] Jyoti Shetty, Anala M R, Shobha G, "A study on cloud forensics: challenges, tools and CSP features", CiiT journal of Biometrics and Bioinformatics, Vol 6, No 6, August 2014, pp 149-153.

[4] Jyoti shetty, Anala M R, Shobha G, " A Framework for secure live migration of virtual machines", in proceedings of Advances in Computing, Communications and Informatics (ICACCI), Mysore, August 2013, pp 243-248.

[5] R Kalaichelvi, L Arockiam, " Secure and Robust Cloud Storage with cryptography and Access Control", published by Elixir Comp. Sci. & Engg., March 2013, pp 13481-13484

[6] http://csrc.nist.gov/publications/nistpubs/800-67-Rev1/SP-800-67-Rev1.pdf [Accessed on 25th December 2014].

[7] http://csrc.nist.gov/groups/STM/cavp/documents/shs/sha256-384-512.pdf [Accessed on 25th December 2014].

[8] Maha Tebaa, El Hajji, Abdellatif El Ghazi, National Days of Network Security and Systems (JNS2), Marrakech, 978-1-4673-1050-5, pp 86-89.

[9] Hao-Miao Yang, Qi XIA, Xiao-fen Wang, Dian-hua Tang, "A New Somewhat Homomorphic Encryption Scheme over Integers", 2012 International Conference on Computer Distributed Control and Intelligent Environmental Monitoring, Hunan, 978-1-4673-0458-0, pp 61-64.

[10] Michael O'Keeffe, "The Paillier Cryptosystem – A look into the Cryptosystem and its Potential Applications", The College of New Jersey, Mathematics Department, April 18, 2008, pp 1-16.

[11] aws.amazon.com [Accessed on 25th December 2014].