Middle-East Journal of Scientific Research
24 (Special Issue on Innovations in Information, Embedded and Communication Systems): 293-298, 2016
ISSN 1990-9233; © IDOSI Publications, 2016
DOI: 10.5829/idosi.mejsr.2016.24.IIECS.23176

Securing Data in Public Cloud

[1]V.B. Buvaneswari, [2]S. Kuppuswami and [3]R.B. Karthiga

[1]Assistant Professor, P.G and Research,
Department of Computer Science, Government Arts College, Coimbatore, India
[2]Professor and Principal, Kongu Engineering College, Perundurai, India
[3]PG Scholar, Department of EEE, Kongu Engineering College, Perundurai, India

Abstract: Cloud computing is the latest and fast growing technology that offers an efficient model to adopt various resources like software, hardware, network, bandwidth and memory. Cloud computing provides cheap and scalable on-demand computing service. The vital issue in cloud computing is protecting the outsourced data in cloud against corruption. The proposed work highlights and categorizes risk factors linked with security in cloud computing. In this research work both the data storage tool Hadoop and authorization database Cassandra toolbox are used. The Advanced Encryption Standard and the Elliptic Curve Cryptography are employed for the design of secured cloud storage. This research work ensures data security and privacy in public cloud environment. The software used for securing data in cloud storage is MATLAB.

Key words: Secured cloud storage · Advanced Encryption Standard · Elliptic curve cryptography · Symmetric Key Cryptography · Public Key Cryptography

INTRODUCTION

Cloud computing is the most demanding technology used all over the world. It offers all forms of services for the users. One of the most outstanding services offered by cloud computing is cloud storage. Cloud storage is a term that refers to the online space that the user can store large amount of personal data. In a more rigorous way, cloud storage is a service model in which data are preserved, handled and backed up remotely and made available to users over a net. Compared with hard disc storage, the cloud storage is some kind of network storage, which provides the memory space for the user. Security is the most important aspect in cloud storage [1]. User's main concern is the vulnerability of the data stored on remote storage system [2].Possibility that a hacker will see an electronic rear door to capture data is always prevalent. Hackers could also try to steal data stored on computers.

Cloud Storage Access: Cloud storage companies invest heavily in security measures in order to curtail data theft or corruption. The data faces the security threats not only during transfer between users and cloud servers but also during storage on servers [3]. The proposed scheme builds a trusted cloud storage system, which allow users to store and access the data securely in the cloud by encrypting the data in the client side and decrypting the data after downloading it from the cloud. Since the secret key is owned by the user of the information, no one can decrypt the data, even though hackers can acquire the information through different approaches [4]. The proposed scheme can assure users about the security of data stored in the cloud.

Storage in Cloud by means of Hadoop: The Apache Hadoop defines the primary storage system used by Hadoop applications. Multiple replicas of data blocks are created and distributed on compute nodes in a cluster to enable the super fast computations. A scaled-out architecture that utilizes group of servers is configured with each server possessing low cost internal disk drives. The Hadoop data are divided into blocks and distributed throughout a cluster. MapReduce involves mapper and reducer. Mapper produces intermediate key/ value pairs from input key/value pairs and the reducer reduces intermediate key/value pairs to a smaller set of values. The MapReduce is suitable for heavy data processing.

Corresponding Author: V.B. Buvaneswari, Assistant Professor, P.G and Research,
Department of Computer Science, Government Arts College, Coimbatore, India.

The Benefits of Hadoop:

- Built-In Redundancy and Failover
- Big Data Capable
- Portability
- Cost-Effective

Cloud Security and its Issues: Security in the cloud is a challenge where confidentiality, integrity and authentication are the essential regions. Data location plays a major role in Cloud Computing security. Location transparency which is one of the important advantages for cloud computing could also be a security threat. Users' personal data security is thus a critical concern in a cloud computing environment [5]. In Cloud Computing, security issues [6, 7] have two folds:

Provider Level: In this step, it is important to make sure that the server is well secured from all the external problems it would come across.

User Level: Although the service provider provides a better security layer, it is important for the users to make sure that there is no tampering or loss of data.
The Cloud Security Issues are broadly divided into

- Data related Issues
- Privacy and legal issues
- Malicious applications.

Related Works: Arokia L and Manikandan S (2014) introduce an ARO crypt technique for security of data stored in cloud storage. This proposed ARO crypt technique is based on the symmetric encryption algorithm. The data encryption is performed before data are stored to the cloud storage. This technique introduces confidentiality as well as protection of data stored in cloud. It ensures better security performance for the stored data and takes larger encryption and decryption time [8]

Randeep Kaur and Supriya Kinger (2014) enhance the security performance for the stored data in the cloud environment. Security and performance is analysed by using various symmetric algorithms. BLOWFISH is less secured but fast. RC5 is secured but slow. AES is secured and very fast [5].

Devi D and Arun P S (2014) proposed an attribute based encryption scheme for providing confidentiality and access control to cloud data storage. The advancement of attribute based technique is Hierarchical Attribute Set Based Encryption that provides scalable, flexible and fine grained access control. This technique allows the owners to securely ensure the integrity of their data in the cloud but fails to prove the confidentiality of data in the cloud environment [4].

Pooja HP and Nagarathna N (2015) explore security issues and various problems faced by cloud users and service providers and also various security threats. Approaches utilized for resolving the privacy issues in cloud resources are analyzed. Data can be accessed anywhere without storing a local copy of data with the help of cloud storage. The major focus is data security. Enabling public auditability for cloud storage is of critical importance so that the user can resort to the Third Party Auditor (TPA) to check the integrity of outsourced data. This ensures the data availability, integrity and unforgeability [9].

Henry C.H Chen and Patrick P.C.Lee (2014) proposed the functional minimum storage regenerating code data integrity protection (FMSR-DIP). It works on basis of thin-cloud storage (NCC cloud) and permits fine tuning of different parameters for a performance-security trade off [10].

This literature survey gives insight on previously performed work on data integrity, security and storage services in cloud.

Background of the Proposed Work: Cloud computing connect many systems either through private or public networks, to provide infrastructure for application, data and file storage that is scalable and dynamic. In the proposed system, there is a great deal of data security in public cloud. There are some cryptographic techniques such as AES and ECC used for securing data in public cloud [11].

AES: Advanced Encryption Standard(AES) is based on a design principle known as a substitution and permutation network. AES is the symmetric key cryptography. The same key performs encryption and decryption of data.

ECC: Elliptic curve cryptography (ECC) is public-key cryptographic approach formed on the basis of algebraic structure of elliptic curves over finite fields. Each user has both public and private key.

- Public Key is provided for encryption or signature verification.
- Private Key is provided for decryption or signature generation.

Proposed System: The main objective of the proposed system is to implement integrity and high level of security in public cloud storage using algorithms.

Symmetric Key: Advanced Encryption Standard
Public Key: Elliptic Curve Cryptography

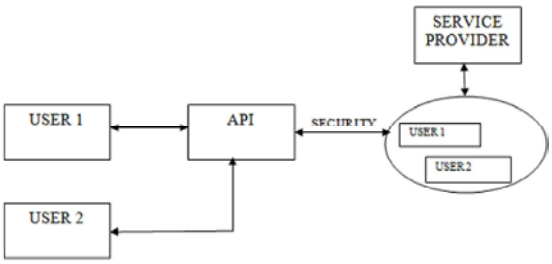

Fig. 1: Block diagram of Proposed System

The Fig. 1 represents the block diagram of the proposed system. The service provider provides data storage in which the user can access and store the data. In present scenario the data is easily hacked by third party or unknown user because of low security level. In this research work, cryptographic techniques are incorporated in order to secure the data in an efficient way.

Cryptographic Techniques for Data Security in Cloud Computing: Cryptographic technique performs data integrity verification in Cloud Storage without the use of Trusted TPA (TTPA). TTPA is a component that is trusted by cloud users as well as service providers. Even though TTPA is reliable, there exist few issues such as leakage of data, scalability, accountability, performance overhead and dynamic data support [12]. In cryptographic algorithms, there are two types of keys namely symmetric key and asymmetric key for data encryption and decryption. Algorithms AES and ECC are used for encrypting and decrypting data are used to achieve data integrity and security in the proposed method [13].

Securing Data Using Aes in Single Node Configuration: Hadoop Distributive File System (HDFS) implements translucent, end-to-end encryption. Reading and writing data to special HDFS directories are transparently encrypted and decrypted without the requirement for modifying user application. The data can be encrypted and decrypted solely by the client and can be termed as end to end. HDFS does not store user data or data encryption keys that are unencrypted.

The requirements for encryption that need to be satisfied are:

- At-rest encryption (i.e., data on disk)
- In-transit encryption (e.g. data transmitted over the network)

Multi-Node Network using AES: An encryption and decryption system and procedure for message forwarding in a multi-node network provides fast message forwarding with less CPU time and power requirements. It performs decryption of all incoming messages and encryption of all outgoing messages that travel through the forwarding nodes unconditionally [14].

Performance: By using AES, the same key is used by both the client and server creates the key distribution problem. Henceforth in the proposed methodology a public key algorithm ECC is suggested to obtain security credentials.

Improved Form of Security Using Elliptic Curve Cryptography: The security of Diffie-Hellman and ElGamal relies on the discrete logarithm in a finite cyclic group. The group traditionally used in public-key cryptography is Z*p, where the best known algorithm for calculating the discrete logarithm is the index calculus method.

Both RSA and ElGamal can be adapted for elliptic curves, where the respective variants are called the elliptic curve analogues. As an instance, security from an RSA key size of 2048 bit can already be reached with a 224 bit elliptic curve key [15].

Design of Elliptic Curve: The cryptosystem consists of a key generation algorithm based on the elliptic curve analogue of Diffie-Hellman and an encryption and decryption algorithm based on the elliptic curve analogue of ElGamal.
An elliptic curve is a plane curve. It is of the form

$$Y^2 = (X^3 + aX + b) \bmod P \quad (1)$$

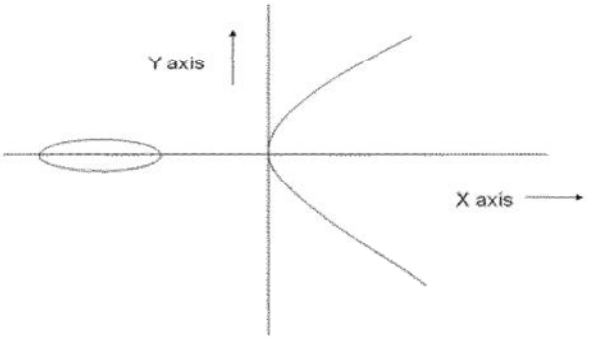

Fig. 2: General form of ECC

The Fig 2. represents the general for of ECC in which X and Y are elements of finite field.
The parameters known in advance are:

- E is the elliptic curve
- P is a curve point which generates a cyclic subgroup of E
- The cyclic subgroup E of order n

For cyclic subgroup of E, the generator is P

<P> = {8, P, 2P, 3P… (n -1)P}

Table 1: Number of Elliptic Curve Operations Done in the Cryptosystem

Operations	Add	Subtract
Key Generation	0	1
Encryption	1	2
Decryption	1	1

The Table 1 describes the no of elliptic curve operations done in the cryptosystem.

- Key-Pair Generation

The input public parameters are used from previous one. The procedure executes as follows:

- Choose d ₤ u, a, r {1… (n - 1)}
- Compute Q = dp
- Output pair (Q, d) Public key: Q, Private key: d

Encryption: Besides the public parameters, the public-key is given as Q. The message M, can be encrypted, which is encoded as a point on the elliptic curve

- Choose k ₤ u, a, r. {1… (n-1)}
- Calculate C1 = kp
- Compute C2 = M + kQ
- Output cipher text: (C1; C2)

One observation is that since k is chosen at random, C2 = M + kQ actually also appears to be random.

Decryption: The cipher text-pair is considered as (C1; C2) and the private key d. The destination is to rebuild the message-point M

C2- dC1 = M + kQ – dkp = M

Performance: By using the ECC, during various sessions various shared secret keys are generated and it is impossible for the intruder to track the data. So, the security performance is found to be high and key distribution problem is also rectified.

Implementation

Steps for the Secure Transmission in Cloud System: Initializing the cloud storage can be made for storing large amount of data (i.e., more than 1 Tera bytes of data) by using the *Hadoop* tool.

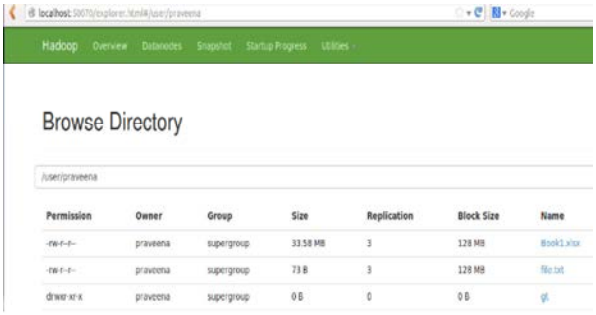

Fig. 3: Storage of Data in Hadoop Environment

The Fig. 3 describes the Hadoop tool provides the space for storing data based on user privileges.

When the user1 wants to access the stored data in cloud environment, he wants to send the user credentials (i.e., user name, id, key).

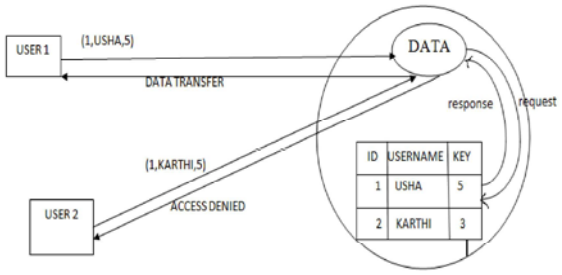

Fig. 4: Authorization using AES

The Fig.4 describes that AES provides the Authorization to the users with the use of Username and key.

The authorized database stores Id, Username and key can be created by using the *Cassandra* tool.

```
cqlsh:demo1> create table table1 (id int,username text,key int,primary key(id));

cqlsh:demo1>
cqlsh:demo1> insert into table1 (id,username,key) values (101,'usha',5);
cqlsh:demo1> insert into table1 (id,username,key) values (102,'karthi',3);
cqlsh:demo1> select * from table1;

 id  | key | username
-----+-----+----------
 102 |   3 |   karthi
 101 |   5 |     usha

(2 rows)
cqlsh:demo1>
```

Fig. 5: Authorized Database

The Fig. 5 illustrates the authorized database created using the tool.

The user1 wants to access the stored information in Hadoop and the centralized authorizer checks the database with the user credentials. If the user credentials matches the authorized database it gives the acknowledgement to the centralized authorizer as YES.

Then the centralized authorizer gives the response as accepted to the user1 for accessing the data.

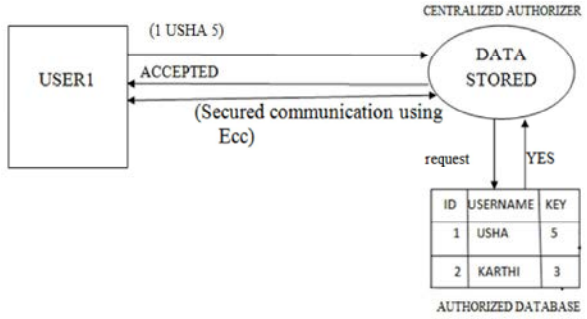

Fig. 6: Secured Communication using ECC

The Fig. 6 shows that once authorization is successful, data transmission between the user and centralized authorizer can take place using ECC.

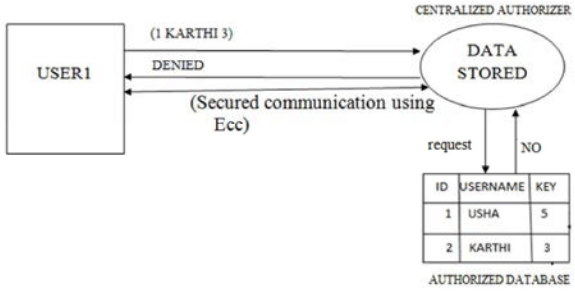

Fig. 7: Rejection of Communication for Hacker

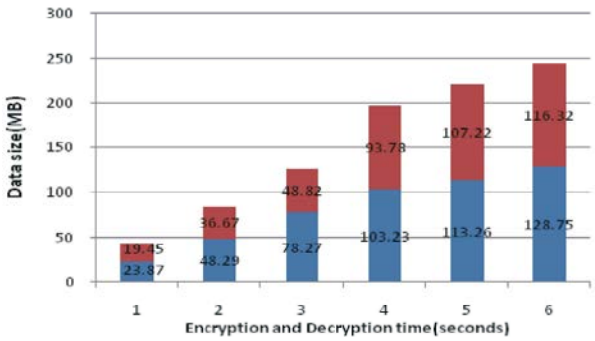

Fig. 8: The Encryption and Decryption Time for AES

The Fig. 7 shows that if authorization is unsuccessful the communication using ECC will be denied.

The graph projected in the Fig.8 highlights the time required for encryption and decryption in AES algorithm.

The Table 2 reports that the key generated for each and every session is a different one which shows that the data transfer using ECC provides high security.

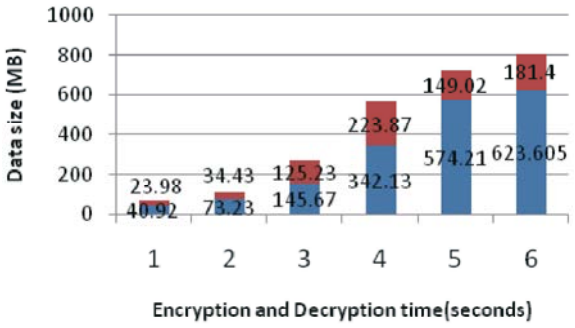

Fig. 9: The Encryption and Decryption Time for ECC

The graph shows in the Fig.9 describes the encryption and decryption time taken by various data size using ECC.

Table 2: Shared Keys Generated for Various Session

Size of Data	Session1	Session2	Session3
1 KB	[13 97 248 55 12 173 29 65 47 128 184 77 20 62 18 87]	[51 137 218 227 97 175 121 176 76 156 142 112 87 246 12 198]	[82 6 86 10 48 106 46 8 90 67 127 210 88 235 87 206]
10 KB	[51 109 94 188 84 54 83 78 97 209 110 99 221 251 163 39]	[82 74 80 120 33 120 153 128 33 168 139 140 212 100 216]	[11 201 241 196 91 216 164 142 218 27 38 235 21 97 200 105]
50 KB	[61 82 103 118 11 35 44 112 23 64 88 42 163 220 102 145]	[36 555.87 41 68 114 108 75 54 26 131 142 163 182 14 59]	[13 144 123 137 103 58 165 12 6 219 75 47 193 247 216 19]
100 KB	[133 58 233 15 3 81 50 75 215 62 166 21 230 117 23]	[126 106 42 254 85 30 6 122 117 250 245 166 217 121]	[127 197 98 112 231 167 15 168 26 89 53 183 46 172 190 41]
150 KB	[22 7 39 165 76 99 34 34 231 34 22 45 63 196 112 23]	[51 45 28 27 87 175 121 176 56 186 152 112 87 246 12 198]	[127 210 45 235 86 83 61 6 92 48 106 40 8 78 86 206]
200 KB	[81 142 210 149 37 115 140 235 218 196 156 73 230 14 169 211]	[74 138 8 240 157 55 183 55 149 100 144 56 64 139 95 51]	[51 137 218 227 97 175 121 176 78 156 142 112 87 246 12 198]

CONCLUSION

In the proposed work, Hadoop tool is used to store data in the cloud. The authorized database can be made using Cassandra. By using the AES the authorization of genuine user was done between the user and authorized database. Once authorization is successful, the secured communication can be made between user and centralized authorizer using Elliptic Curve

Cryptography: The time for encryption and the decryption is 0.005776 seconds.

The security is found to be high and the intruder will be unable to access the data in cloud storage by using Elliptic Curve Cryptography.

In future, storage of data in Public cloud can be implemented using Hyper elliptic Curve Cryptography and its performance can be analyzed.

ACKNOWLEDGMENT

I thank Kongu Engineering College Research Director Prof.S.Kuppuswami for his valuable guidance and support. I extend my thanks to Dr.S.Usha, Assistant Professor Electrical and Electronics Engineering Department, Kongu Engineering College for her valuable support.

REFERENCES

1. Nelson Gonzalez, Charles Miers, 2012, A quantitative analysis of current security concerns and solutions for cloud computing (SPRINGER), pp: 1-18.
2. Ayesha Malik, Muhammad Mohsin Nazir, 2012, Security Framework for Cloud Computing Environment: A Review, Journal of Emerging Trends in computing and Information Sciences, 3: 390-394.
3. Cong Wang, Qian Wang, Kui Ren and Wenjing Lou, 2013. Privacy Preserving Public Auditing for Secure Cloud Storage, IEEE TRANSCATIONS ON COMPUTERS, 62(2): 362-375.
4. Devi, D. and P.S. Arun, 2014, A Design for secure data sharing in cloud, International Journal of Engineering Research and General Science, 2(5): 72-77.
5. Rajkumar Chalse, Ashwin Selokar and Arun Katara, 2013, A New Technique of Data Integrity for Analysis of the Cloud Computing Security, 5th International Conference on Computational Intelligence and Communication Networks, pp: 469-472.
6. Subashini, S. and V. Kavitha, 2011, A survey on security issues in service delivery models of cloud computing (ELSEVIER), pp: 1-11.
7. Nada Ahmed, Ajith Abraham, 2013, Modelling Risk Factors in a Cloud Computing environment, Journal of Information Assurance and Security, 8: 279-289.
8. Arokia, L. and S. Manikandan, 2014, A security service algorithm to ensure confidentiality of data in cloud storage, International Journal of Engineering Research & Technology, 3(12): 1053-1058.
9. Pooja, H.P. and N. Nagarathna, 2015. Privacy Preserving Issues and their Solutions in Cloud Computing: A Survey, IJCSIT, 6(2): 1588-1592.
10. Henry, C.H. Chen and Patrick P.C Lee, 2014. Enabling Data Integrity Protection in Regenerating-Coding-Based Cloud Storage: Theory and Implementation IEEE TRANSACTION, 25(2): 407-416.
11. Anshu Parashar and Rachna Arora, 2013, Secure User Data In Cloud Computing using Encryption Algorithm, International Journal of Scientific & Engineering Research, 3: 1922-1926.
12. Abdul, Salah H., 2014. Secure Third Party Auditor for Ensuring Data Integrity in Cloud Storage, IEEE Conference, 3(6): 510-517.
13. Manpreet Kaur and Rajbir Singh, 2013. Implementing Encryption Algorithms to Enhance Data Security of Cloud in Cloud Computing, International Journal of Computer Applications, 70: 16-21.
14. Subhasri, P. and A. Padmapriya, 2013, Multilevel Encryption for Ensuring Security in Public Cloud, International Journal of Advanced Research in Computer Science and Software Engineering, 3: 527-532.
15. Xiao Chun YIN, Zeng Guang LIU and Hoon Jae LEE, 2014. An Efficient and Secured Data Storage Scheme in Cloud Computing Using ECC-based PKI, ICACT, pp: 523-527.