International Journal of Engineering and Technical Research (IJETR)
ISSN: 2321-0869 (O) 2454-4698 (P), Volume-5, Issue-1, May 2016

Design and Implementation of Secure Cloud Storage System using Hybrid Cryptography Algorithms with Role based Access Control Model

Anjali DV, Dr. S.N Chandrashekara

***Abstract*— Cloud computing is one of the significant shift in technological evolution, is gaining momentum as multitude of organizations. It provides services over a network to an organization with the ability to scale up or down their service requirements. Cloud computing services are established and provided by a third party. There are many security requirements models need to present the security policies intended to protect information against unauthorized access and modification stored in a cloud. The presenting work describes the approach for modelling the security requirements by applying the cryptography concepts. This work is designed by AES, RSA, SHA-1 algorithm for encryption and decryption of data and role based access control model is used to provide access according to the role played by user. This paper also describes the mathematical model for calculating the trust of the user. This model gives the uploading rights to the user when he/she recommended by the Administrator and Owner when users exceeds the specified experience and trust threshold value.**

***Index Terms*— Role Based Access Control, AES, RSA, SHA-1, Cloud computing, Trust Management.**

I. INTRODUCTION

The Cloud Computing provides main services, Information as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS), Data as a Service (DaaS).It reduces the cost of hardware required to store data that could have been used at user end. Instead of purchasing the infrastructure that is required to store data and run the processes we can lease the assets according to our requirements. The cloud computing provides the number of advantages over the traditional computing and it include: quickness, lower cost, scalability, device independency and location independency. There are a number of security concerns associated with cloud computing. These issues fall into two broad categories: security issues faced by cloud providers and security issues faced by their customers. The responsibility is shared, however. The provider must ensure that their infrastructure is secure and that their clients' data and applications are protected while the user must take measures to fortify their application and use strong passwords and authentication measures are taken. Some of the security problems and their solutions of them are described below. Due to sharing computing resources with another company physical security is lost .User does not have knowledge and control of where the resources run and stored. It can be insured by using secure

Data Transfer. Second, maintaining the consistency or integrity of the data. It can be insured by providing Secure Software Interfaces. Third, Privacy rights may be violated by cloud service providers and hackers. Forth, when cryptographic technique was used then who will control the encryption/decryption keys? It can be ensured by implementing security in cloud computing is must which will break the difficulty of accepting the cloud by the organizations. There are varieties of security algorithms which can be implemented to the cloud. There are two types of algorithms symmetric key and asymmetric key. In cloud computing, symmetric key and asymmetric key algorithms is used to encrypt and decrypt the data. In this work RSA algorithm is used to generate encryption and decryption keys for AES symmetric algorithm and SHA-1 hashing algorithm for key generation. Another major issue is how to manage user access to cloud storage system. For that different access control mechanism can be enforced for cloud users. Access Control is nothing but giving the authority to users to access the specific resources, applications and system. There are three access control models, such as MAC (Mandatory access control model), DAC (Discretionary access control model) and RBAC Role based access control models. These access control models specify the set of rules or criteria to access the system and its resources. In this present work, I used RBAC model for providing access control to the users.

II. RELATED WORK

(1) Developing Secure Cloud Storage System by Applying AES and RSA Cryptography Algorithms with Role Based Access Control Model addressed security issues in cryptographic role-based access control systems for securing data storage in a cloud environment. It presented a RBAC with AES and RSA based secure cloud storage system which allows an organization to upload data securely in a public cloud, while it stores organizational information on a private cloud. The experience trust model was integrated into the SCSS. (2) Achieving Secure Role-Based Access Control on Encrypted Data in Cloud Storage, describes a new RBE scheme that achieves efficient user revocation. Then it presents a RBAC based cloud storage architecture which allows an organisation to store data securely in a public cloud, while maintaining the sensitive information related to the organization's structure in a private cloud. Then it develops a secure cloud storage system architecture and have shown that the system has several superior characteristics such as constant size cipher text and decryption key. From this experiment, it had been observe that both encryption and decryption computations are efficient on the client side, and decryption time at the cloud can be reduced by having multiple processors, which is common in a cloud environment. It believed that the proposed system has the potential to be useful in commercial situations as it captures practical access policies based on roles in a flexible manner

Anjali DV,M. Tech, Department of CSE, SJCIT, Chickballapur
Dr. S.N Chandrashekara, Professor & HOD, Department of CSE, SJCIT, Chickballapur

and provides secure data storage in the cloud enforcing these access policies. (3) A Trust-Based Access Control Model for Pervasive Computing Applications explains traditional access control models are mostly not be suitable for pervasive computing applications. Towards this end, it proposes a trust based access control model as an extension of RBAC. It use the context-sensitive model of trust proposed earlier as the underlying trust model. It investigates the dependence of various entities and relations in RBAC on trust. This dependency necessitates changes in the invariants and the operations of RBAC. The configuration of the new model is formalized using graph-theoretic notation. In future, it has been planned to incorporate other environmental contexts, such as space and time, to the designed model. It also plan to investigate conflicts and redundancies among the constraint specification. Such analysis is needed before the model can be used for real world applications. (4)Secure Role Based Data Access Control in Cloud Computing, aims at fine-grained data access control in cloud computing. One challenge in this context is to achieve fine graininess, data confidentiality, and scalability. Simultaneously, which is not provided by current work? This paper proposed a scheme to achieve the goal by exploiting KP- ABE and uniquely combining it with techniques of proxy re-encryption and lazy re-encryption. Moreover, this proposed scheme can enable the data owner to delegate most of computation overhead to powerful cloud servers. Confidentiality of user access privilege and user secret key accountability can be achieved. Formal security proofs show that our proposed scheme is secure under standard cryptographic models.(5) Achieve Fine Grained Data Access Control in Cloud Computing Using KP-ABE along-With Lazy and Proxy Re-Encryption.

III. ALGORITHMS

AES performs all its computations on bytes rather than bits. Hence, AES treats the 128 bits of a plaintext block as 16 bytes. These 16 bytes are arranged in four columns and four rows for processing as a matrix. AES uses 10 rounds for 128-bit keys which are calculated from the original AES key. The schematic of AES structure is given in the following illustration −

Encryption Process

Here, each round comprise of four sub-processes. The first round process is depicted below −

Byte Substitution *Sub Bytes*

The 16 input bytes are substituted by looking up a fixed table *S − box* given in design. The result is in a matrix of four rows and four columns. **Shift rows,** each of the four rows of the matrix are shifted to the left. Any entries that 'fall off' are re-inserted on the right side of row. Shift is carried out as follows, First row is not shifted. Second row is shifted one *byte* position to the left. Third row is shifted two positions to the left. Fourth row is shifted three positions to the left. The result is a new matrix consisting of the same 16 bytes but shifted with respect to each other. **Mix Columns**, each column of four bytes is now transformed using a special mathematical function. This function takes as input the four bytes of one column and outputs four completely new bytes, which replace the original column. The result is another new matrix consisting of 16 new bytes. It should be noted that this step is not performed in the last round. **Add round key**, the 16 bytes of the matrix are now considered as 128 bits and are XORed to the 128 bits of the round key. If this is the last round then the output is the ciphertext. Otherwise, the resulting 128 bits are interpreted as 16 bytes and we begin another similar round.

Decryption Process

The process of decryption of an AES ciphertext is similar to the encryption process in the reverse order. Each round consists of the four processes conducted in the reverse order − Add round key, Mix columns, Shift rows, Byte substitution.

RSA Cryptosystem

A user wishing to exchange encrypted messages using a public-key cryptosystem would place their public encryption procedure, E, in a public file. The user's corresponding decryption procedure, D, is kept confidential. Rivest, Shamir, and Adleman provide four properties that the encryption and decryption procedures have:

1. Deciphering the enciphered form of a message M yields M. That is, D(E(M)) = M
2. E and D are easy to compute.
3. Publicly revealing E does not reveal an easy way to compute D. As such, only the user can decrypt messages which were encrypted with E. Likewise, only the user can compute D efficiently.
4. Deciphering a message M and then enciphering it results in M. That is, E(D(M)) = M

SHA-1 operates as follows:

1. Pad message so its length is 448 mod 512
2. Append a 64-bit length value to message
3. Initialize 5-word (160-bit) buffer (A;B;C;D;E) to (67452301;ef cdab89;98badce;10325476;c3d2e1 f 0)
4. Process message in 16-word (512-bit) chunks:
   _ Expand 16 words into 80 words by mixing and shifting
   _ Use 4 rounds of 20 bit operations on message block and buffer
   _ Add output to input to form new buffer value
5. Output hash value is the final buffer value

IV. IMPLEMENTATION

This procedure contains following operations. In this work, Advanced Encryption Standard (AES) algorithm used for encrypting and decrypting the data and RSA algorithm is used to encrypt the secret key generated by the AES algorithm. Finally for each encrypted data key is generated using SHA-1 (Secure Hashing Algorithm) where these key data are stored on cloud which provides second level of security. Were key generated using encrypting and hashing key are different. When the roles in the system defined then for each role one public key and private key is created. This public key is used by the Data Owner to encrypt and upload the data in a private key is used by the user to gain access for downloading data from the cloud. When administrator creates the role manager it will generate the secret key for that role and this secret key is used by Role Manager to assign role to users. When the user wants to decrypt the data he will first request for the key data used by the SHA-1 hashing functions which will direct to the

cipher text generated by the AES and RSA algorithm. As the decrypting values are stored in private cloud this request will be forwarded to the private cloud which will return the private key for decrypting the cipher texts. After validation the user can run the decryption algorithm to recover the data. User will get uploading rights when he/she finishes a specified experience threshold value and got the recommendation from the Administrator and Data Owner. To receive recommendation from the administrator and data owner users past behaviour and their transaction history will be considered. The received recommendations and his/her experience are uploaded in the central repository. When the trust value needs to be calculated the trust engine will use this record for its reference. The entities which are outside the trust management system will not be able to access this repository. Another entity is the Role behaviour audit which keeps track of the feedbacks stored for particular role. These feedbacks will be stored again in the central repository. Based on these parameters the trust decision engine takes decision whether or not user will get uploading rights or not. Experience based trust uses the past experience of the user to build the trust on the user. There are a range of other attributes and credentials such as different types of privileges, the state of the platform being used as well as reputations, recommendations and histories that come into play in decision making. Experience-based trust model is one such trust management system which enables the trust.

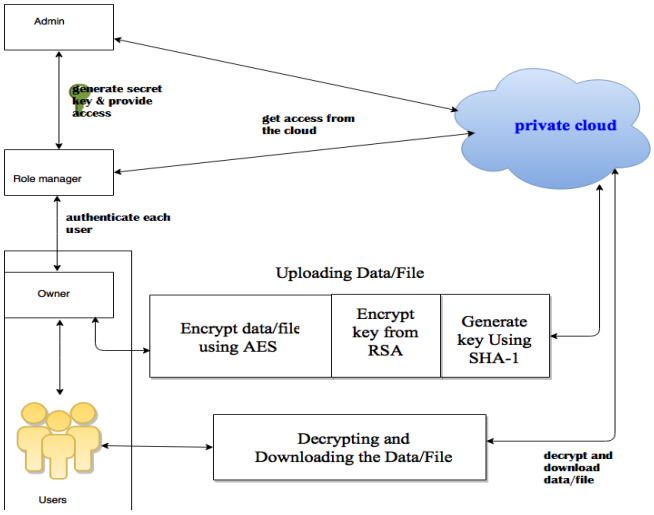


Figure 1: System Architecture

Decisions to be made based on the historical behaviour of an entity. Such a system allows an entity to rate the transactions with other entities, and the trustworthiness of an entity is determined using the collection of Decisions to be ratings of the transactions that other entities have had with this entity. Most experience based trust systems derive the trustworthiness of an entity from both its own experience and the Feedbacks on the transactions provided by other entities which have had interactions with the entity concerned in the past. When a user finishes a specified experience threshold value and got the recommend from the Administrator and Data Owner he/s got the rights to upload data in a cloud. In symmetric-key algorithm, the same secret key is used for both encryption and decryption, in contrast to asymmetric-key cryptography algorithm symmetric-key cryptography algorithm like AES (Advanced Encryption Standard) is high speed and it requires low RAM requirements, but because of the same secret key used for both encryption and decryption, it faces big problem of key transport from sender side to receiver side. But in asymmetric-key algorithm, it needs two different keys for encryption and decryption, one of which is private key and one of which is public key. The public key can be used to encrypt plaintext; whereas the private key can be used to decrypt cipher text .As compared to symmetric-key algorithm, Asymmetric-key algorithm does not having problem while key exchanging and transporting key, but it is mathematically costly. To solve the problem of key transport and get better performance these algorithms can be combined together. In this data receiver generates the key pairs using asymmetric- key algorithm, and distributes the public key to sender. Sender uses one of the symmetric-key algorithms to encrypt data, and then sender uses asymmetric-key algorithm to encrypt the secret key generated by the symmetric-key algorithms with the help of receiver's public key. Then receiver uses its private key to decrypt the secret key, and then decrypt data with the secret key. In this paper, asymmetric-key algorithm is used only for encrypting the symmetric key, and it requires negligible computational cost. It similarly works like SSL. For encrypting the files or file data, AES (Advanced Encryption Standard) algorithm is used, and RSA (Rivest, Shamir and Adleman) is used to encrypt AES key. These encrypted files can be uploaded according to role perspective. In proposed system, Role based access control is used for authenticating the users to access files uploaded or given rights for the specific roles and to maintain the data privacy and integrity AES and RSA algorithms are used. General description of AES and RSA algorithms is given below. AES algorithm starts with an Add round key stage then followed by 9 rounds of four stages and a tenth round of three stages. The four stages are Substitute bytes, Shift rows, Mix Columns and Add Round Key. The tenth round not performs the Mix Columns stage. Next step is to generate key hash value using SHA-1 for each user data/file which provides data integrity acts as the second level of security and provides security inside the cloud whenever the data/file is not duplicated the file which we want to store inside the cloud will checked with generated key by using SHA-1 algorithm and the file will get encrypted and stored on the cloud. These stages also apply for decryption. The first nine rounds of the decryption algorithm consist of Inverse Shift rows, Inverse Substitute bytes, Inverse Add Round Key and Inverse Mix Columns. Again, the tenth round not performs the Inverse Mix Columns stage. Let us consider S is secret key and C is cipher key, then at encryption C=S mod n and at decryption side S = C mod n. n is very large number which is created during key generation process. In proposed scheme, the administrator of the system defines different job functionalities required in an organization, then according to the needs of organization he add users or employees. After that, owner of the data encrypts the data in such a way that only the users with appropriate roles as specified by a RBAC policy can decrypt and view this data. The Role Manager assigns roles to users who are appropriate for that role and he can also remove the users from assigned role. The cloud provider (who owns the cloud infrastructure) is not able to see the contents of the data. A Role Manager is able to assign a role for particular user after the owner has encrypted the data or file for that role. A user assigned to particular role can be revoked at any time in which case, the revoked user will not

have access rights to data or file uploaded for this role. Revocation of user from role will not affect other users and roles in the system. This approach, achieves an efficient encryption and decryption on the client side.

V. RESULTS

Different techniques used for applying Role based access control policies and encryption and decryption techniques to a Cloud storage system such as HKM, HIBE and ABE and RBE. First approach is to apply the access control policies is to transform the access control problem into a key management problem. Different approaches can be used to apply HKM schemes to enforce RBAC policies for data storage. But, these solutions have numerous limitations. For example, if the data owners and users are large then, it increase the overhead required to setting up the key infrastructure. Furthermore, when one of the user's access permission is deleted, then all the keys and public values known to this user need to be changed, which makes these schemes unfeasible. In this paper, the popular secret key algorithm that is AES, RSA with SHA-1 was implemented, and their performance was calculated by encrypting different input files of varying contents and sizes. The algorithm was implemented in a scripting language called python, using their standard specifications, and tested on three different hardware platforms, to compare their performance. The performance is represented in the below graph.

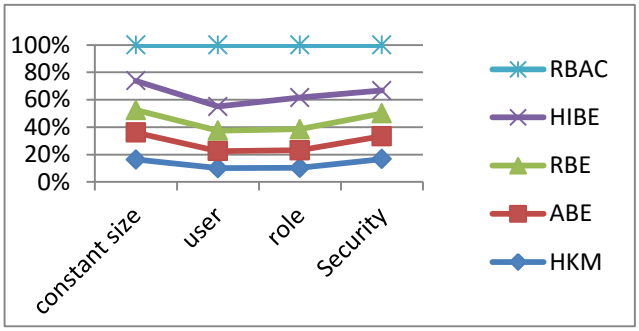


Figure 2: Performance Analysis

VI. CONCLUSION

In this paper, I have addressed security issues in cryptographic role-based access control systems for securing data storage in a cloud environment. I presented a RBAC with AES, RSA and SHA-1 based secure cloud storage system which allows an organization to upload data securely in a public cloud, while we have stored organizational information on a private cloud. The experience trust model was integrated into the SCSS. This helps administrator and owner to give uploading rights to users. The model also keeps the constant size of cipher text and decryption key. The computational cost required for encryption and decryption are efficient on the client side. The proposed system is useful in various commercial situations as it implements the role based access policies based on the job functionality in a flexible manner and provides secure data storage in the cloud enforcing cryptographic techniques.

REFERENCES

[1] Bokefodejayant d." Developing secure cloud storage system by applying aes and rsa cryptography algorithms with role based access control model" international journal of computer applications (0975 – 8887) volume 118– no.12, may 2015

[2] Lanzhou "achieving secure role-based access control on encrypted data in cloud storage" ieee transactions on information forensics and security, vol. 8, no. 12, december 2013

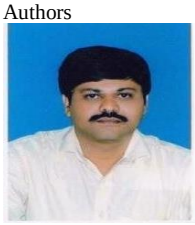
[3] Manachaitoahchoodee" a trust-based access control model for pervasive computing applications" u.s.afosr under contract fa9550-07-1-0042

[4]V.sathyapreiya "secure role based data access control in cloud computing" international journal of computer trends and technology- may to june issue 2011

[5] Hulawalekalyani" achieve fine grained data access control in cloud computing using kp-abe along-with lazy and proxy re-encryption" international journal of emerging technology and advanced engineering volume 4, issue 2, february 2014)

[6] Prachi shah" data security for cloud storage system using role based access control" international journal of science and research (ijsr) issn (online): 2319-7064 index copernicus value (2013): 6.14 | impact factor (2013): 4.438 [7] Sudipchakraborty" trustbac integrating trust relationships into the rbac model for access control in open systems" sacmat'06, june 7–9, 2006, lake tahoe, california, usa.

Authors



Dr Sn. Chandrashekara, B.E, M.Tech, Ph.D. Professor And HOD, Department Of Computer Science And Engineering, Sjcit, Chickballapur. Membership At Fie, Isse, Lmiste, Lmcsi.

Anjali Dv, B.E (Ise), M.Tech, Department Of Computer Science And Engineering, Sjcit, Chickballap