Developing Secure Cloud Storage System by Applying AES and RSA Cryptography Algorithms with Role based Access Control Model

Bokefode Jayant D.
Research Scholer,
Sinhgad College of Engineering,Korti,Pandharpur
Solapur University,India.

Ubale Swapnaja A
Professor,
Sinhgad College of Engineering,Korti,Pandharpur
Solapur University, India

Pingale Subhash V.
Professor,
Sinhgad College of Engineering,Korti,Pandharpur
Solapur University, India

Karande Kailash J., Ph.D
Principal,
Sinhgad College of Engineering,Korti,Pandharpur
Solapur University,India

Apate Sulabha S., Ph.D
Associate Professor,
Walchand Institue of Technology,Solapur
Solapur University, India

ABSTRACT
Cloud computing is one of the emerging and promising field in Information Technology. It provides services to an organization over a network with the ability to scale up or down their service requirements. Cloud computing services are established and provided by a third party, who having the infrastructure. Cloud computing having number of benefits but the most organizations are worried for accepting it due to security issues and challenges having with cloud. Security requirements required at the enterprise level forces to design models that solves the organizational and distributed aspects of information usage. Such models need to present the security policies intended to protect information against unauthorized access and modification stored in a cloud. The proposed work describes the approach for modeling the security requirements from the perspective of job functions and tasks performed in an organization by applying the cryptography concepts to store data on cloud with the smallest amount of time and cost for encryption and decryption processes. In this work, we used RSA and AES 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 shows 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.

Keywords
Role Based Access Control, AES, RSA, Cloud computing, Trust Management.

1. INTRODUCTION
The Cloud Computing provides three main services, Information as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (Saas)[1]. 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. Security in cloud computing is one of the most critical aspects due to importance and sensitivity of data stored on the cloud. But cloud computing has several major issues and concerns, such as data security, user access control, data integrity, trust and performances issues. In order to solve these problems, many schemes are proposed under different systems and security models [2]-[6].

Whenever we talk about security of cloud computing, there is various security issues come up in a cloud. 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. It can be ensured using cryptographic technique. Forth, when cryptographic technique was used then who will control the encryption/decryption keys? It can be ensured by giving rights to the users/customers. So 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 [22], [15]-[18]. DES, Triple-DES, AES, and Blowfish etc are some symmetric algorithms can be used to implement cloud security. RSA and Diffie-Hellman Key Exchange are the asymmetric algorithms these can be used to generate encryption and decryption key for symmetric algorithms. In cloud computing, symmetric key and asymmetric key algorithms is used to encrypt and decrypt the data. In presented work, RSA algorithm is used to generate encryption and decryption keys for AES symmetric algorithm.

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 [7]. In MAC, The administrator has all the privileges to assign the user's roles according to his wishes. And end users does not have authority to change the access policies specified by the administrator therefore it is very

restrictive and less used access control model. It can be used in a very sensitive environment. For example military, research centers [8]. In DAC, the end users have authority to change the access policy for any objects. But if an attacker gets control over the account it is too dangerous. So giving the complete authority to users is not good for any organization. In RBAC, first different roles or jobs can be specified and then these roles can be assigned to cloud user so these user can get access according to their jobs requirement. It is effectively and mostly used access control model within an organization because access to particular data and resources can be given according to the roles [7], [9]-[14]. In presented work, we used RBAC model for providing access control to the users.

2. LITERATURE SURVEY

Mainly there are two types of cryptographic algorithms used for encryption and decryption, such as Symmetric key algorithms, Asymmetric key algorithms and Combination key algorithms. Encryption of data and its keys will make the secure cloud network and maintain the data privacy. Encryption is the process in which one can encode a message or data into unreadable format so intruder not able to read it. User has plain text when it is encoded into unreadable format using one of the encryption technique called as cipher text. After received by the correct receiver, he/she can decrypt it into the original plain text. Encryption is mostly used in network communications to achieve the data confidentiality.

2.1 Symmetric Encryption

This is one of the simplest encryption technique in which have only one secret key for encryption and decryption. There are number of symmetric key encryption algorithms in use which includes block ciphers like DES Blowfish, AES, Camellia, Serpent etc. and stream ciphers like FISH, Py,RC4, QUAD, SNOW etc[15]-[18].

2.2 Asymmetric Encryption

Asymmetric cryptography techniques known as public key cryptography, in which two separate keys, are used for encryption and decryption. Where one key is publicly available called as public key and it is used for encryption, and the other key is private key and it is used for the decryption.

Following techniques are the mostly used cryptographic techniques for cloud computing.

AES: In cryptography, the Advanced Encryption Standard (AES) is mostly used symmetric-key encryption standard. AES is a block cipher having block length of 128 bits block size, with key sizes of 128, 192 and 256 bits, respectively.AES allows for three different key lengths: 128, 192, or 256 bits. Encryption and Decryption for 128-bit keys needs 10 rounds of processing, for 192-bit keys needs 12 rounds of processing, and for 256-bit keys needs 14 rounds. All other rounds are identical for encryption and decryption; except for the last round in each case. It provides greater efficiency for software as well hardware also [20].

MD5: MD5 (Message-Digest algorithm 5) is a very famous and well known hash function and it generates a 128-bit resulting hash value. MD5 is commonly used in various applications to provide security, and it is also used to ensure the integrity of files. The MD5 value generated for specific file is considered as reliable fingerprint that can be used to check the integrity of the file contents. This algorithm had been implemented in different computer languages including C, Perl, and Java. In MD5 algorithm sender uses the public key provided by the receiver to encrypt the message and receiver uses its private key to decrypt the message [16].

DES: The DES (Data Encryption Standard) algorithm is one of the most used encryption algorithm in the world. DES was developed by IBM and it is symmetric block cipher.DES uses a 56-bit key for encrypting and decrypting a 64-bit block of data. The algorithm is more suitable to implement on hardware and not for software, because it gives low performance and it is time consuming [19].

RSA: RSA also called as public-key cryptography algorithm, it involves a public key and a private key. The public key can be used for encrypting messages and it is known to everyone. Messages can be decrypted using the private key. Data can be encrypted prior to storage, and establishes secure transmission channels [21].

3. PROPOSED SCHEME

3.1 Components of Architecture

Proposed scheme has the following four main entities. System Administrator is the authority who generates the username and password for the all users and secret key for the Role Manager, and to define the role hierarchy. Role Manager manages the user membership of a role. Owner is the person who has the authority to upload/store data securely in the cloud. Users will want to access and decrypt the stored data in the cloud.

3.1.1 Public Cloud:

Public cloud is a third party cloud provider which resides outside the infrastructure of the organizations and organizations outsource their users' encrypted data to the public cloud. Since the public cloud is untrusted, data stored in the public cloud could be accessed by unauthorized parties, such as employees of the cloud provider and users from other organizations who are also using services from the same cloud. Therefore only public information and encrypted data will be stored in the public cloud. An untrusted public cloud may deny a user's request for accessing stored data in the cloud or provide users with incorrect data. Such behaviors will result in the users not being able to access the data stored in cloud but will not cause violation of RBAC policies. These behaviors can be detected, as a user can observe the failure immediately after s/he communicates with the public cloud. In this case, organization may choose to change the cloud provider to a more reliable one, especially if the current provider is found to be malicious.

3.1.2 Private Cloud:

Private cloud is built on an internal data centre that is hosted and operated by a single organization. The organization only stores critical and confidential information in this private cloud. The amount of this in public cloud, so this cloud does not need to have the information is relatively small comparing to the data stored capacity to handle large volumes of data. The private cloud only provides interfaces to the administrator and role managers of the role-based system and to the public cloud. Users do not have direct access to the private cloud. This helps to reduce the attack surface of the private cloud. The purpose of using a private cloud is to ensure that correct and up-to-date information about the organization's structure and user membership are used in the decision making. To achieve efficient user revocation, the private cloud is assumed

to be honest-but-curious in order to use the proposed scheme in this architecture. That is, the cloud will faithfully execute the scheme and will not collaborate with revoked users.

3.1.3 Hybrid Cloud:

This cloud is a mixture of the two or more clouds. In this the public cloud and private cloud both are used. In this it integrates the advantages of each one for overcoming the others obstacle. The private cloud will not be available for the user. The user will only interact with the public cloud and the administrator of the system will be allowed to access the private cloud. This model is managed both by the third party entity and organization. It can be placed in the onsite or off site location.

3.1.4 Administrator:

The administrator is the main authority of the organization and secure cloud storage system. The administrator has all the system parameters and credentials required to manage the secure cloud storage system. The administrator specifies the organization structure and creates the role hierarchy according to it. Administrator adds user and role manager in the system and gives credentials to access the secure cloud storage system.

3.1.5 Role Manager:

A role manager is the party who manages the relationship between users and roles. When updating the user membership of a role, the role manager needs to enter the secret given by the administrator. None of users are affected by this operation, so role managers do not need to communicate with users, and they only need to interact with the private cloud. Before a user is included into a role, the role manager will need to authenticate the user in order to ensure that the user is qualified user.

3.1.6 Owner:

An owner can be a user within the organization who has the authority to encrypt and upload data in the cloud for other users to access; owners specify who can access the data according to the role-based policies. In the proposed model, owner manages the relationship between permissions and roles. Owner performs the encryption operation for that it does not require any secret key.

3.1.7 User:

Users are the employees of the organization who has particular job functionality according to their skills and wants certain data from the public cloud to perform this job functionality. Each and every user is authenticated by the administrator of the secure cloud storage system. Users are not having authority to update the organization structure. They are allowed only for downloading the data assigned for their roles.

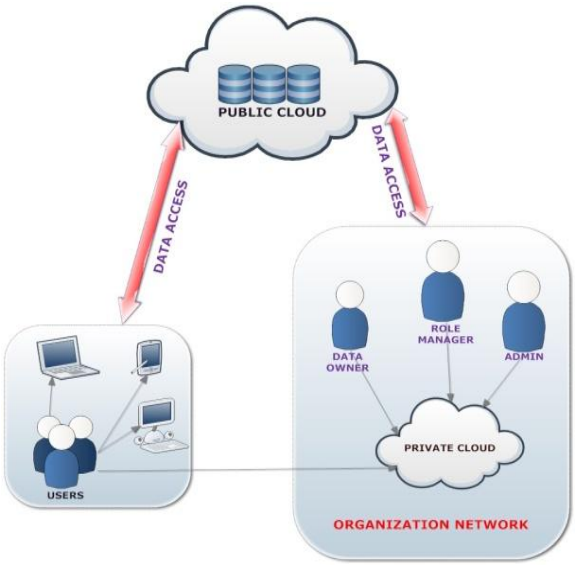


Fig 1. Secure Cloud Storage System

3.2 Experience Based Trust

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. Recently, a number of models have been developed using soft trust techniques to determine the trustworthiness of systems. Experience-based trust model is one such trust management system which enables the trust decisions to be made based on the historical behavior of an entity and it is shown in figure1. 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 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. In proposed system, user does not have authority to upload data in the public cloud through system. When a user finishes a specified experience threshold value and got the recommendation from the Administrator and Data Owner he/s got the rights to upload data in a public cloud. To receive recommendation from the administrator and data owner users past behavior and their transaction history will be considered. The received recommendations and his/her experience are uploaded in the central repository.

3.3 Working of SCSS

In this, first needs to create the user, assign roles to the users. This procedure contains following operations. In this work, Advanced Encryption Standard (AES) [20] algorithm used for encrypting and decrypting the data and RSA algorithm [21] is used to encrypt the secret key generated by the AES algorithm. 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 public cloud and 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

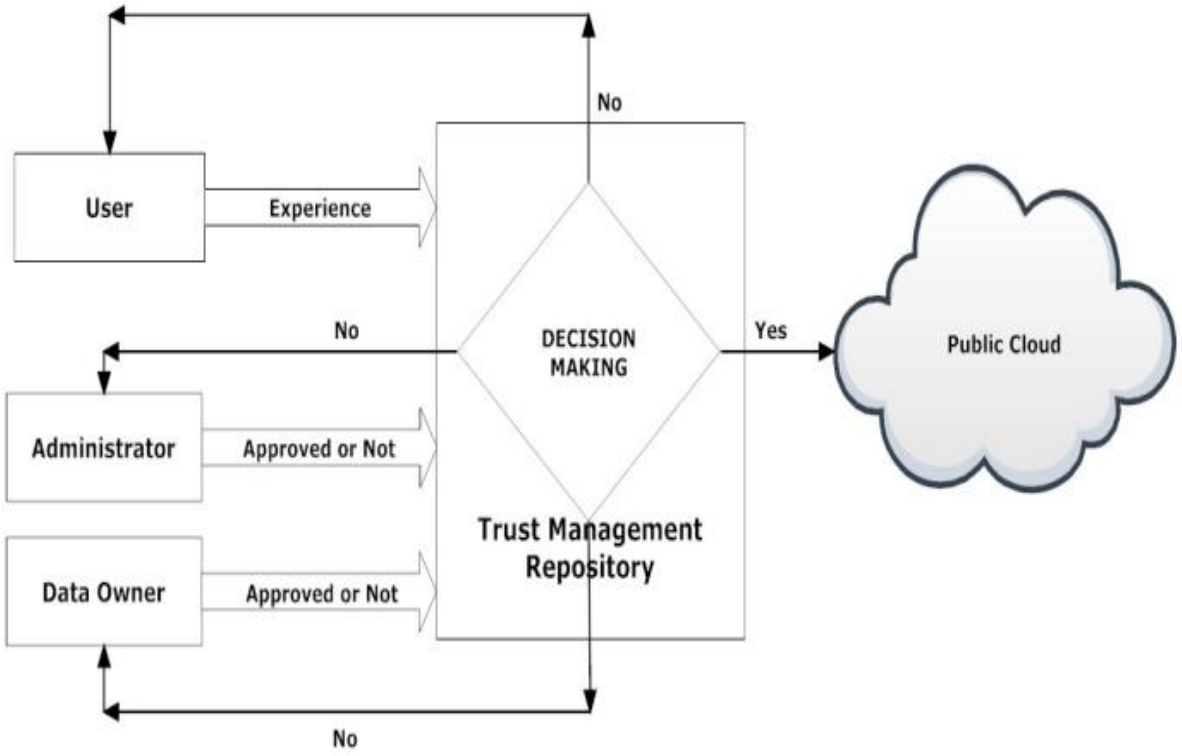


Fig 2. Trust Management system

to users. When the user wants to decrypt the data he will first request for the cipher text from the public cloud.

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 behavior 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 behavior 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. The architecture of the trust management system is shown in figure1.

3.4 System Parameters

Our system uses the AES for the purpose of encryption and decryption of the data. The main purpose of using this algorithm is to provide more security to the data which will be uploaded on to the cloud. The system is developed in asp.net. For the public cloud we have taken instance from Microsoft azure and private cloud is created using the Linux with i5 processor and 8 GB ram. The use of latest processor will reduce the response time for uploading and delivering of the data to the owner and user respectively. The size of the decryption key is another important factor in cloud storage system. The decryption key needs to be portable as users may use the storage service from different clients. The experimental results show that the size of the decryption key is 48 bytes, which is convenient for the users.

4. METHODOLOGY

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 [15]-[18].

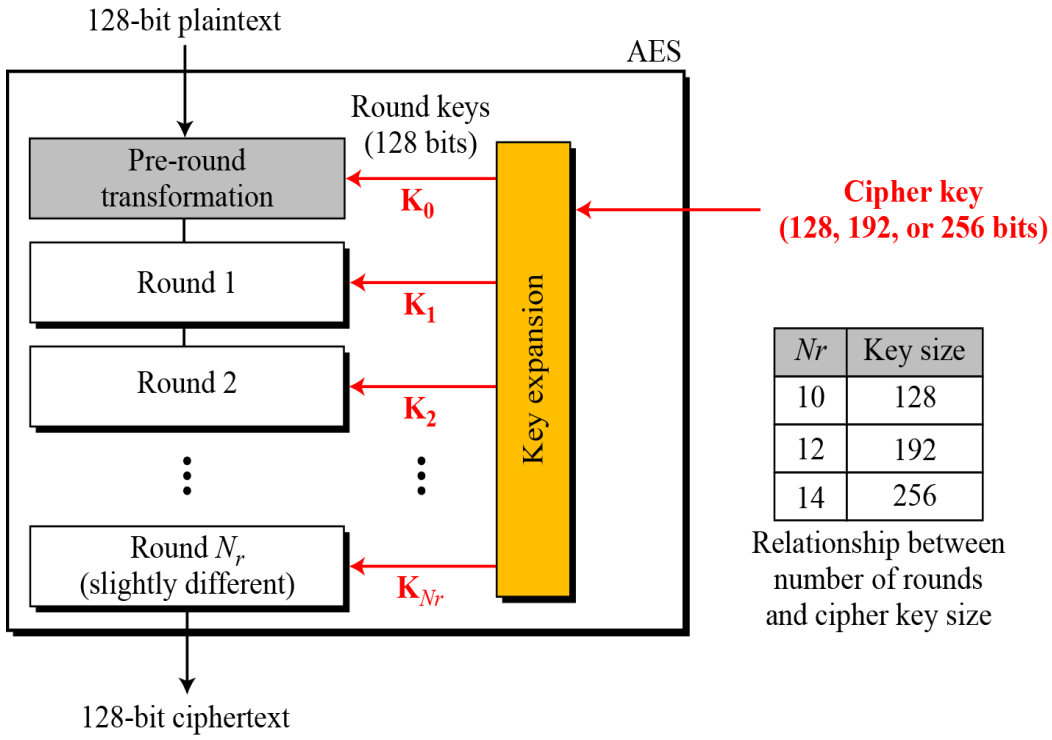
To solve the problem of key transport and get better performance these 2 algorithms can be combined together. In this data receiver generates the key pairs using asymmetric-key algorithm, and distributes the public key to sender. Sender uses one of the symmetric-key algorithms to encrypt data, and then 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 [20], [21]. 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. Working of AES is shown in figure 3.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. For more details see [20].

RSA make use of measured exponential for encoding and decoding symmetric key that is secret key generated by the AES algorithm. 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 [21].

In proposed scheme, the administrator of the system defines different job functionalities required in a 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.



Nr	Key size
10	128
12	192
14	256

Relationship between number of rounds and cipher key size

Fig 3. AES Encryption Cipher

5. RESULT AND DISCUSSION

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 are discussed in [24]–[26]. 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.

Another scheme of keys management is Hierarchical ID-based Encryption (HIBE) [27], [28].But in this technique the length of the user identity becomes longer when the depth of hierarchy increases. Another approach is role-based encryption scheme (RBE) in [29]. But, the user revocation in this scheme needs to update of all the role related parameters. Another approach was introduced in [30]. In this approach, the size of the cipher text increases linearly when role

hierarchy increases. If single user belongs to different roles, then multiple keys need to be managed by this user. The implemented technique solves all these limitations. In this technique, for each and every role separate secret keys given to manage the user membership. Another scheme is ABAC; in this scheme access is given to the user depending on the attributes. In this techniques first attributes are defined as the access rights or rules, and to gain the access users have these attributes [31]-[34]. But, in this approach, the size of key is not constant and if one of these users can be revoked from the role hierarchy then the other user's keys are updated for the same role [32]. The result of implemented scheme is compared with all existing techniques shown in Table I.

In this paper, the popular secret key algorithm that is AES with RSA was implemented, and their performance was calculated by encrypting different input files of varying contents and sizes. The algorithm was implemented in a uniform language (Asp.net), using their standard specifications, and tested on three different hardware platforms, to compare their performance. Result of comparison is shown in Table II.

Table 1. Comparison of different scheme with implemented scheme

Techniques	HKM	HIBE	ABE	RBE	PROPO
Constant size	Yes	Yes	No	Yes	Yes
Constant size	Yes	No	No	Yes	Yes
Constant size	Yes	No	No	Yes	Yes
User	No	No	No	Yes	Yes
Role	No	No	No	No	Yes
Security	Low	Low	Low	Low	High

Table 2. Testing result of implemented scheme on different platforms

(Input Size) MB approximately	Intel P-4 2.4 GHz	Intel Core2 Duo	Intel Core i3/i5/i7 & AMD.
20-30	1.8	0.3	0.028
50-60	4.5	0.9	0.071
100-110	9.09	1.8	0.14
200-210	18.20	3.2	0.29
400-410	37	6.1	0.57
500-510	45.49	7.3	0.71
600-620	54.54	7.9	0.85
700-730	63.86	8.3	1
800-899	72.77	9	1.15
900-1000	81	9.8	1.29

6. CONCLUSION

In this paper, we have addressed security issues in cryptographic role-based access control systems for securing data storage in a cloud environment. We 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 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. Finally we have also shown the experimental results of encryption and decryption the message. The implemented model also keeps the constant size of cipher text and decryption key. Also we observe that, computational cost required for encryption and decryption are efficient on the client side. We trust that 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.

7. ACKNOWLEDGMENTS

I would like to express my special thanks and gratitude to my friend Mr.Babashaeb Kale who helped me a lot to configure private cloud using Microsoft technology.

8. REFERENCES

[1] Zhidong Shen, Li Li , Fei Yan, Xiaoping Wu. Cloud Computing System Based on Trusted Computing Platform. In Proc.International Conference on Intelligent Computation Technology and Automation, Volume 1, May 2010, pp. 942-945.

[2] Pearson S., Benameur A. Privacy, Security and Trust Issues Arises from Cloud Computing, In Proc. IEEE Second International Conference on Cloud Computing Technology and Science (CloudCom). 2010, pp. 693-702.

[3] Rohit Bhadauria and Sugata Sanyal, A Survey on Security Issues in Cloud Computing and Associated Mitigation Techniques. International Journal of Computer Applications, Volume 47- Number 18, June 2012, 47-66.

[4] Mohammed E.M. , Ambelkadar H.S, Enhanced Data Security Model on Cloud Computing, In Proc. 8th International Conference on IEEE publication 2012,pp.12-17.

[5] Sang Ho. Na, Jun-Young Park, Eui- Nam Huh, Personal Cloud Computing Security Framework, In Proc. Service Computing Conference (APSSC) IEEE publication, Dec 2010,pp. 671-675.

[6] Wang, J.K.; Xinpei Jia, Data Security and Authentication in hybrid cloud computing model, Global High Tech Congress on Electronics (GHTCE) on IEEE publication, 2012, 117-120.

[7] Bokefode J.D, Ubale S. A, Apte Sulabha S,Modani D. G, Analysis of DAC MAC RBAC Access Control based Models for Security, International Journal of Computer Applications, Volume 104 – No.5, October 2014.

[8] B. W. Lampson. Protection, ACM SIGOPS Operating System Review, 8(1), January 1974, pp18–24.

[9] H. L. F. Ravi S. Sandhu, Edward J. Coyne and C. E. Youman. Role-based access control models. IEEE Computer, February 1996, pp.38–47.

[10] R. Sandhu. The next generation of access control models: Do we need them and what should they be? In SACMAT'01, May 2001, page 53

[11] D. Ferraiolo and R. Kuhn. Role-based access controls. In Proc. of the 15th NIST-NCSC Naional Computer Security Conference, Baltimore, MD, October 1992, pp 554–563.

[12] R. Sandhu and Q. Munawer. The ARBAC99 model for administration of roles. In Proc. Of the 15th Annual Computer Security Applications Conference, Phoenix, Arizona, December 1999.

[13] R. Sandhu, E.J. Coyne, H.L. Feinstein, C.E. Youman. The ARBAC97 Model for Role-Based Administration of Roles. In Proc. of 2nd ACM Work-shop on Role Based Access Control, 1997.

[14] R. Sandhu, Q. Munawer. The RRA97 Model for Role Based Administration of Role Hierarchies. In Proc. of 3rd ACM Workshop on Role Based Access Control, 1998.

[15] W. Stallings, Cryptography and Network Security Principles and Practices Fourth Edition, Pearson Education, Prentice Hall, 2009.

[16] Tingyuan Nie, and Teng Zhang ,A Study of DES and Blowfish Encryption Algorithm, IEEE publications, 2009.

[17] Singh, S preet, and Maini, Raman Comparison of Data Encryption Algorithms,International Journal of Computer science and Communication,vol.2,No.1,January-June 2011,pp.125-127.A.

[18] Atul khate, Cryptography and Network Security, 2nd Ed, Tata Mcgraw hill, 2009, pp.87-2004.

[19] Davis, R., The Data Encryption Standard in Perspective,In Proc. of Communication Society magazine, IEEE, Volume 16 No 6, Nov. 1978, pp. 5-6.

[20] Daemen, J., and Rijmen, V. ,Rijndael: The Advanced Encryption Standard. Dr. Dobb's Journal, March 2001.

[21] R.L.Rivest, A.Shamir, and L.Adleman, A Method for Obtaining Digital Signatures and Public-Key Cryptosystems, Communication of the ACM, Volume 21 No. 2, Feb. 1978.

[22] Prof. S.A.Ubale and Dr. S.S. Apte, Study and Implementation of Code Access Security with .Net Framework for Windows Operating System, International Journal of Computer Engineering & Technology (IJCET), Volume 3, Issue 3, 2012, pp. 426 – 434.

[23] Prof. S. A. Ubale, Dr. S. S. Apte, Comparison of ACL Based Security Models for securing resources for Windows operating system,IJSHRE Volume 2 Issue 6, Page No 63.

[24] S. D. C. Di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati, Over-encryption: Management of access control evolution on outsourced data, In Proc. VLDB, Sep. 2007, pp. 123–134.

[25] C. Blundo, S. Cimato, S. D. C. Di Vimercati, A. D. Santis, S. Foresti, S. Paraboschi, et al.,Efficient key management for enforcing access control in outsourced scenarios, In SEC (IFIP), vol. 297. New York, NY, USA: Springer-Verlag, May 2009, pp. 364–375.

[26] P. Samarati and S. D. C. di Vimercati, Data protection in outsourcing scenarios: Issues and directions," In Proc. ASIACCS, Apr. 2010, pp. 1–14.

[27] C. Gentry and A. Silverberg,, Hierarchical ID-based cryptography, in ASIACRYPT (Lecture Notes in Computer Science), vol. 2501. New York, NY, USA: Springer-Verlag, 2002, pp. 548–566.

[28] D. Boneh, X. Boyen, and E.-J. Goh, Hierarchical identity based encryption with constant size ciphertext, in EUROCRYPT (Lecture Notes in Computer Science), vol. 3494. New York, NY, USA: Springer-Verlag, May 2005, pp. 440–456.

[29] L. Zhou, V. Varadharajan, and M. Hitchens, Enforcing role-based access control for secure data storage in the cloud, In Comput. J., vol. 54, no. 13, Oct. 2011, pp. 1675–1687.

[30] Y. Zhu, H. Hu, G.-J. Ahn, H. Wang, and S.-B. Wang,Provably secure role-based encryption with revocation mechanism, J. Comput. Sci. Technol., vol. 26, no. 4,2011, pp. 697–710.

[31] V. Goyal, O. Pandey, A. Sahai, and B. Waters, Attribute-based encryption for fine-grained access control of encrypted data,In Proc. ACM Conf. Comput. Commun. Sec., Oct./Nov. 2006, pp. 89–98.

[32] S. Yu, C. Wang, K. Ren, and W. Lou, Achieving secure, scalable, and fine-grained data access control in cloud computing, In Proc. IEEE INFOCOM, Mar. 2010, pp. 534–542.

[33] Y. Zhu, D. Ma, C. Hu, and D. Huang,,How to use attribute-based encryption to implement role-based access control in the cloud,In Proc. Int. Workshop Sec. Cloud Comput., 2013, pp. 33–40.

[34] Swapnaja A. Ubale, S. S. Apte, Bio-enable Security for Operating System by Customizing Gina, High Performance Architecture and Grid Computing Communications in Computer and Information Science Volume 169, 2011, pp 179-185.