iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 2 **Issue:** XI **Month of publication:** November 2014

DOI:

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

Securing the Data in Cloud Storage Using Cryptosystem Along With Steganography

S.Priyanka[#1], R.Lavanya*[2]
CSE DEPARTMENT, Veltech Multitech Dr.Rangarajan Dr.Sakunthala Engineering College Chennai, Tamil Nadu

***Abstract*—Cloud Computing is a highly developing technology where user data (both critical and non-critical) is shared on a centralized server. The sharing of the user's critical data on a third party cloud server does not guarantee the promised level of security and there is a threat to compromise user data. The security issues are classified in to two categories where one is related to the cloud provider and the other to the consumer. In order to safeguard the data and improve the security measures in the consumer side, cryptographic encryption technique is used where aggregate key and steganography concepts are implemented. This project aspires to provide secured data sharing in cloud storage with the help of steganography concepts in addition with the sharing of aggregate keys between data owners and data users**
***Keywords*— Cloud storage, data sharing, aggregate key encryption, steganography**

I. INTRODUCTION

Cloud computing is an architecture for providing computing services via the internet on demand and pay per use access to a pool of shared resources namely networks, storage, servers.Service providers offers cloud platforms for their customer to use and create their web services, much like internet service providers offer customer high speed broadband to access the internet. There are numerous security issues for cloud computing as it encompossess many technologies. Security issues in cloud computing consists of application, platforms and infrastructure segment. Each segment performs different operations and offer different products for business and individuals around the world. The business application includes saas utility computing, web services, pass managed service providers, service commerce and internet integration. The cloud computing encounters various security issues, as it comprises of many technologies namely, networks, database, operating systems, virtualization, resources scheduling, transaction management, load balancing, concurrency control, memory management. Therefore security issues for many of these systems and technologies are applicable to cloud computing. For example 1.The network that interconnects the system in a cloud has to be secure.2.Mapping the virtual machines to physical machines has to be carried out securely.3.Data security involves encrypting the data as well as ensuring that appropriate policies are enforced for data sharing.
Considering data privacy, a traditional way to ensure it is to rely on the server to enforce the access control after authentication (e.g., [5]), which means any unexpected privilege escalation will expose all data. In a shared-tenancy cloud computing environment, things become even worse. Data from different clients can be hosted on separate virtual machines (VMs) but reside on a single physical machine in a target VM could be stolen by instantiating another VM coresident with the target one [7].Regarding the availability of files, there are the series of cryptographic schemes which as go as far as allowing the third party auditor to check the availability of files on behalf on the data owner without leakage of anything about the data [8] or without compromising the data owner anonymity [9].A cryptographic solution ,for example [11], with proven security relied on number-theoretic assumption is more desirable, whenever the user is not perfectly happy with trusting the security of the VM or the honesty of the technical staff. These are motivated to encrypt their data with their own keys before uploading them to server. Users can download the encrypted data from the storage, decrypt them, then send them to others for sharing, but it loses the value of cloud storage. Users should be able to delegate the access rights of the sharing data to others so that they can access these data from the server directly. Drop box is an example .Assume that data owner puts all her private photos on drop box and she does not like to share her photos to everyone.

Due to data leakage just relying on the privacy protection mechanisms provided by drop box, so she encrypts all the photos using her own keys before uploading .One day, Data owner friend i.e. data user asks her to share the photos taken over all these years which data user appeared in .Data owner can then use the share function of drop box, but the problem now is how to delegate the decryption rights for these photos to data user. A possible option data owner can choose is to securely send data user secret keys involved. In a proxy re-encryption scheme a semi-trusted proxy converts a cipher text for data owners into a cipher text for data requestor without seeing the underlying plaintext. The fundamental property of proxy re-encryption schemes is that the proxy is not fully trusted.[2].
Several efficient proxy re-encryption schemes that offer security improvements over earlier approaches, The primary advantage

of our schemes is that they are unidirectional (i.e., Alice can delegate to Bob without Bob having to delegate to her) and do not require delegators to reveal all of their secret key to anyone – or even interact with the delegate – in order to allow a proxy to re-encrypt their cipher texts. In this schemes, only a limited amount of trust **is** placed in the proxy[3]. Identity management is one of the most common services deployed within companies and organizations because of its key role in access control, authorization and accountability processes. However, it introduces an overhead in cost and time, and in most cases it requires specific applications and personnel for managing, integrating and maintaining this service the cloud offers an innovative opportunity of externalizing this workload; this is what has been called Identity Management as a service (IDaaS or IDMaaS).[6]In a proxy re-encryption (PRE) scheme, a proxy is given special. Information that allows into translate a cipher text under one key into a cipher text of the same message under a different key. The proxy cannot, however, learn anything about the messages encrypted under either key. This paper propose a definition of security against chosen cipher text attacks for PRE schemes address the problem of obtaining PRE schemes that are secure in arbitrary protocol settings, or in other words are secure against chosen cipher text attacks[10].

A. Our Contributions

Cryptographic algorithms play a dominant role in securing the files without getting attacked by the intruders. But in this current scenario there is numerous number of chances to break an encryption algorithm by performing cryptanalytic attacks. So the demand for a strong encryption algorithm becomes vital. Become It is obvious that if a message is encrypted by using more than one encryption algorithm then it cannot be easily breaked by the eavesdroppers. Cloud users probably will not hold the strong belief that the cloud server is doing a good job in terms of confidentiality. A cryptographic solution says whenever the user is not perfectly happy with trusting the security of the Honesty of the technical staff; these users are motivated to encrypt their data with their own keys before uploading them to the server.

B. Benefits of Cryptographically Secured Storage Services

1) Confidentiality Assurance: In a cryptographic storage service, the data is encrypted on-premise by the data processors. This way, customers can be assured that the confidentiality of their data is preserved irrespective of the actions of the cloud storage provider. This greatly reduces any legal exposure for both the customer and the provider.

2) Geographic restrictions: In a cryptographic storage service data is only stored in encrypted form so any law that pertains to the stored data has little to no effect on the customer. This reduces legal exposure for the customer and allows the cloud storage provider to make optimal storage infrastructure, thereby reducing costs.

3) Subpoenas: In a cryptographic storage service, since data is stored in encrypted form and since the customer retains possession of all the keys, any request for the (unencrypted) data must be made directly to the customer.

4) Reducing Risk of Security Breaches: Even if a cloud storage provider implements strong security practices there is always the possibility of a security breach. If this occurs the customer may be legally responsible. In a cryptographic storage service data is encrypted and data integrity can be verified at any time. Therefore, a security breach poses little to no risk for the customer.

5) Data retention and destruction: In many cases a customer may be responsible for the retention and destruction of the data it has collected. If this data is stored in the cloud, however, it can be difficult for a customer to ascertain the integrity of the data or to verify whether it was properly discarded. A cryptographic storage service alleviates these concerns since data integrity can be verified and since the information necessary to decrypt data (i.e., the master key) is kept on-premise. Secure data erasure can be effectively achieved by just erasing the master key.

We solve this problem by introducing a special type of public-key encryption which we call key -aggregate cryptosystem. In KAC users encrypts a message not only under a public-key, but also under an identifier of cipher text called class. The key owner holds a master-secret called master-secret key, which can be used to extract secret keys for different classes. More importantly, the extracted key have can be an aggregate key is as compact as a secret key for a single class, but aggregates the power of many such keys, i.e., the decryption power for any subset of cipher text classes. our solution, Data owner can simply send dta user a single aggregate key via a secure e-mail. Bob can download the encrypted photos from Alice's. Drop box space and then use this aggregate key to decrypt these encrypted photos. The scenario is depicted in Fig. 1. The public system parameter has size linear in the number of cipher text classes, but only a small part of it is needed each time and it can be fetched on demand from large (but nonconfidential) cloud storage. We propose several concrete KAC schemes with different security levels and extensions in this paper. All constructions can be proven secure in the standard model. To the best of our knowledge, our aggregation mechanism

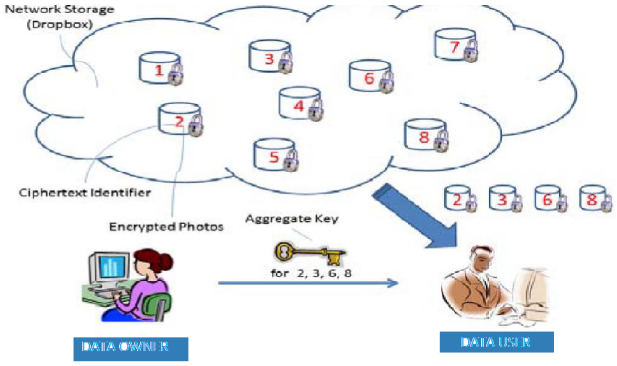


Fig. 1. Alice shares files with identifiers 2, 3, 6, and 8 with Bob by sending him a single aggregate key.

II. RELATED WORK

A. Aggregation of Secret Keys:

Introducing a special type of public-key encryption which we call key-aggregate cryptosystem (KAC).In KAC, users encrypt a message not only under a public key, but also under an identifier of ciphertext called class. The key owner holds a master-secret called master-secret key, which can be used to extract secret keys for different classes. More importantly, the extract key can be an aggregate key which is compact as a secret key for a single class, but aggregates the power of many such keys, i.e., the decryption power for any subset of ciphertext classes.

B. Framework

The data owner establishes the public system parameter via Setup and generates a public/master-secret3 key pair via KeyGen. Messages can be encrypted via Encrypt by anyone who also decides what cipher text class is associated with the plaintext message to be encrypted. The data owner can use the master-secret to generate an aggregate decryption key for a set of cipher text classes via Extract. The generated keys can be passed to delegates securely (via secure e-mails or secure devices) finally; any user with an aggregate key can decrypt any cipher text provided that the cipher text's class is contained in the aggregate key via Decrypt.

- Setup (1^λ,n): executed by the data owner to setup an account on an untrusted server. On input a security level parameter 1 λ and the number of cipher text classes n (i.e., class index should be an integer bounded by 1 and n), it outputs the public system parameter param, which is omitted from the input of the other algorithms for brevity.
- KeyGen: executed by the data owner to randomly generate a public/master-secret key pair (pk; msk).
- Encrypt (pk; i;m): executed by anyone who wants to encrypt data. On input a public-key pk, an index i denoting the cipher text class, and a message m, it outputs a cipher text C.
- Extract (msk,S):executed by the data owner for delegating the decrypting power for a certain set of cipher text classes to a delegate. On input the master-secret key msk and a set S of indices corresponding to different classes, it outputs the aggregate key for set S denoted by KS.
- Decrypt (Ks,S,i,C): executed by a delegate who received an aggregate key KS generated by Extract. On input KS, the set S, an index i denoting the cipher text class the cipher text C belongs to, and C, it outputs the decrypted result m if i ε S.

C. Steganography:

Steganography is the practice of concealing a message, image, or file within another message, image, or file. Steganography sometimes is used when encryption is not permitted. More commonly, steganography is used to supplement encryption. An encrypted file may still hide information using steganography, so even if the encrypted file is deciphered, the hidden message is not seen. Encrypted Outlet of original Data, Public Key and Index is made stegno into an Image.Data owner has to share the selected Image along with the ADK to download the Original Data.

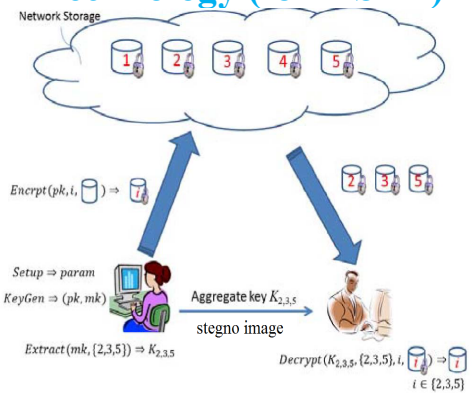


Fig 2.Encryption using stegno image

III. ARCHITECTURE DESIGN

Fig 3.Data sharing in cloud

In the proposed system, Data owner randomly generates public/master-secret key pair after account is created in the server. Data owner encrypts the data, public key and data index & then uploaded in the Cloud Server. In additional steganography concept is used. Encrypted Outlet of original Data, Public Key and Index is made stegno into an Image. Data owner has to share the selected Image along with the ADK to download the Original Data. The advantage of this concept it provides high security. Preserving data integrity and confidentiality, minimizing the number of ciphertext storage

We are going to create an User application by which the User is allowed to access the data from the Server of the Cloud Service Provider. Here first the User wants to create an account and then only they are allowed to access the Network. Once the User creates an account, they are to login into their account and request the Job from the Cloud Service Provider. Based on the User's request, the Cloud Service Provider will process the User requested Job and respond to them. All the User details will be stored in the Database of the Cloud Service Provider. In this Project, we will design the User Interface Frame to Communicate with the Cloud Server through Network Coding using the programming Languages like Java/ .Net. By sending the request to Cloud Server Provider, the User can access the requested data if they authenticated by the Cloud Service Provider.

Cloud Service Provider will contain the large amount of data in their Data Storage. Also the Cloud Service provider will maintain the all the User information to authenticate the User when are login into their account. The User information will be stored in the Database of the Cloud Service Provider. Also the Cloud Server will redirect the User requested job to the Resource Assigning Module to process the User requested Job. The Request of all the Users will process by the Resource Assigning

Module. To communicate with the Client and with the other modules of the Cloud Network, the Cloud Server will establish connection between them. For this Purpose we are going to create a User Interface Frame. Also the Cloud Service Provider will send the User Job request to the Resource Assign Module in Fist in First out (FIFO) manner.

Cloud servers are constructed with the files and the index information are maintained in the main cloud server. The data are added in each cloud servers, and network construction is made with the entire data index present in each cloud server. Query is given to the main cloud server, so that the main cloud server will verify the index information present in it & divert the query to the corresponding cloud servers.

Data owner uploading the file it is encrypted with AES algorithm and provided with public key Cloud Service Provider will contain the large amount of data in their Data Storage. Also the Cloud Service provider will maintain the all the User information to authenticate the User when are login into their account. The User information will be stored in the Database of the Cloud Service Provider. Also the Cloud Server will redirect the User requested job to the Resource Assigning Module to process the User requested Job. The Request of all the Users will process by the Resource Assigning Module. To communicate with the Client and with the other modules of the Cloud Network, the Cloud Server will establish connection between them. For this Purpose we are going to create a User Interface Frame. Also the Cloud Service Provider will send the User Job request to the Resource Assign Module in First in First out (FIFO) manner.

A. Algorithm details

1) AES (Asymmetric Encryption Standard): This algorithm is flexible in supporting any combination of data and key size of 128, 192, and 256 bits. However, AES merely allows a 128 bit data length that can be divided into four basic operation blocks. These blocks operate on array of bytes and organized as a 4×4 matrix that is called the state. For full encryption, the data is passed through Nr rounds (Nr = 10, 12, 14) . These rounds are governed by the following transformations

(i) Byte Substitution: This is a non linear byte Substitution, using a substation table (s-box), which is constructed by multiplicative inverse and affine transformation.
(ii) Shifting the rows: This is a simple byte transposition, the bytes in the last three rows of the state are cyclically shifted; the offset of the left shift varies from one to three bytes.
(iii) Mixing of columns: Is equivalent to a matrix multiplication of columns of the states. Each column vector is multiplied by a fixed matrix. It should be noted that the bytes are treated as polynomials rather than numbers.
(iv) Adding round key: Is a simple XOR between the working state and the round key. This transformation is its own inverse.

IV. CONCLUSION

In this paper a new approach is used as stegnography technique alone with the encryption of cryptosystem to provide more security for data owner whose who upload the data and data user who download the data.more preserving data integrity and confidentiality and Minimizing the number of ciphertext storage.

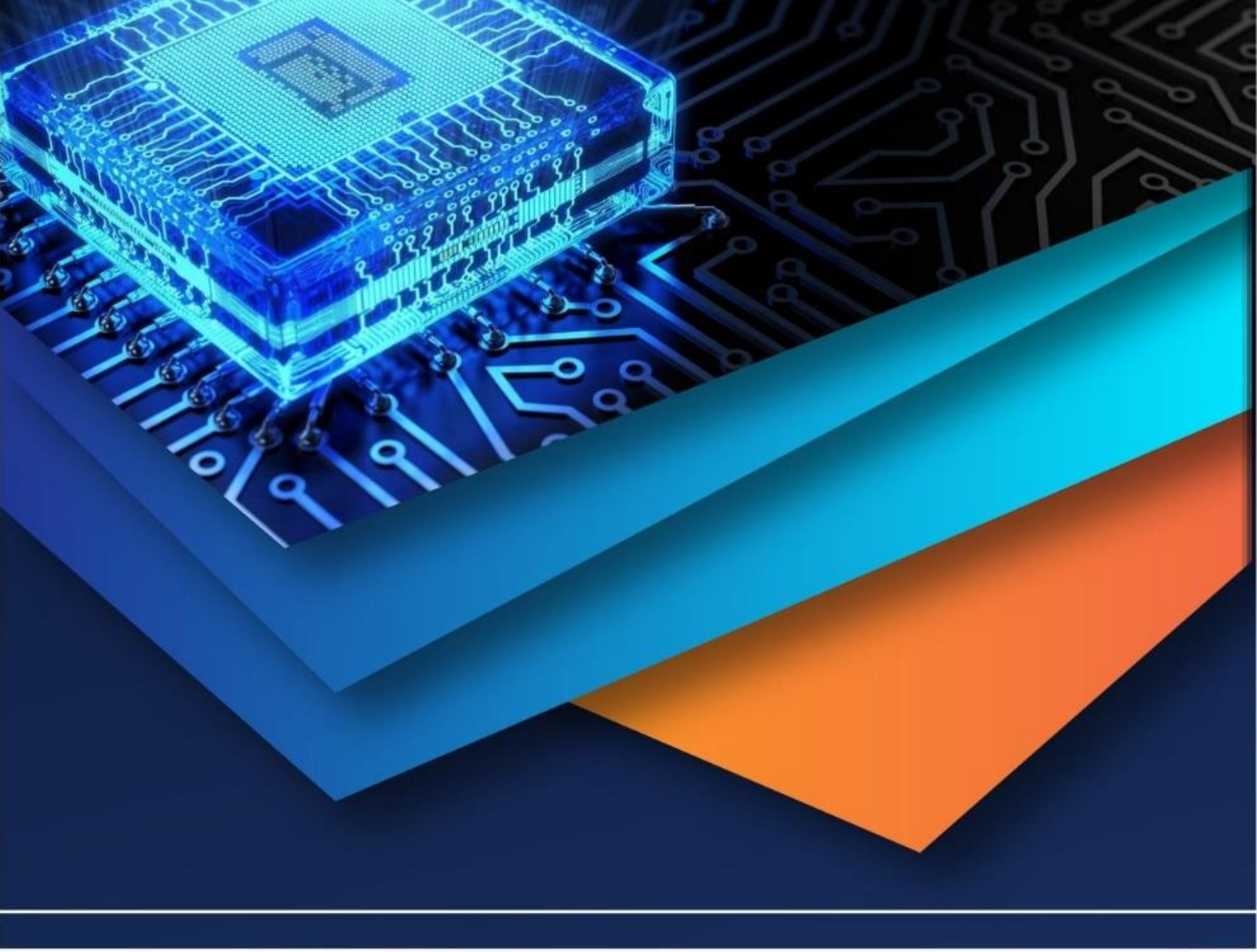
V. ACKNOWLEDGMENT

I would like to thank my guide Ms.R.Lavanya for assisting me in this paper work.

REFERENCES

[1] D. Boneh, C. Gentry, B. Lynn, and Shacham, "Aggregate and Verifiably Encrypted Signatures from Bilinear Maps," Proc. 22nd Int'l Conf. Theory and Applications of Cryptographic Techniques (EUROCRYPT '03), pp. 416-432, 2003.
[2] C.-K. Chu and W.-G. Tzeng, "Identity-Based Proxy Re-encryption without Random Oracles," Proc. Information Security Conf. (ISC '07), vol. 4779, pp. 189-202, 2007.
[3] G. Ateniese, K. Fu, M. Green, and S. Hohenberger, "Improved Proxy Re-Encryption Schemes with Applications to Secure Distributed Storage," ACM Trans. Information and System Security, vol. 9, no. 1, pp. 1-30, 2006.
[4] S.S.M. Chow, J. Weng, Y. Yang, and R.H. Deng, "Efficient Unidirectional Proxy Re-Encryption," Proc. Progress in Cryptology (AFRICACRYPT '10), vol.

6055, pp. 316-332, 2010.
[5] S.S.M. Chow, Y.J. He, L.C.K. Hui, and S.-M. Yiu, "SPICE – Simple Privacy-Preserving Identity-Management for Cloud Environment," Proc. 10th Int'l Conf. Applied Cryptography and Network Security (ACNS), vol. 7341, pp. 526-543, 2012.
[6]"Integrating OpenID with proxy re-encryption to enhance privacy in cloud-based identity services" David Nu˜nez, Isaac Agudo, and Javier Lopez Network, Information and Computer Security Laboratory Universidad de M´alaga M´alaga, Spain.
[7] L. Hardesty, Secure Computers Aren't so Secure.MITpress,http://www.physorg.com/news176107396.html, 2009
[8] C. Wang, S.S.M. Chow, Q. Wang, K. Ren, and W. Lou, "Privacy- Preserving Public Auditing for Secure Cloud Storage," IEEE Trans. Computers, vol. 62, no. 2, pp. 362-375, Feb. 2013.
[9] B. Wang, S.S.M. Chow, M. Li, and H. Li, "Storing Shared Data on the Cloud via Security-Mediator," Proc. IEEE 33rd Int'l Conf. Distributed Computing Systems (ICDCS), 2013.
[10] R. Canetti and S. Hohenberger, "Chosen-Ciphertext Secure Proxy Re-Encryption," Proc. 14th ACM Conf. Computer and Comm. Security (CCS '07), pp. 185-194, 2007.
[11] S.S.M. Chow, C.-K. Chu, X. Huang, J. Zhou, and R.H. Deng, "Dynamic Secure Cloud Storage with Provenance," Cryptography and Security, pp. 442-464, Springer, 2012.
[12] C.-K. Chu, J. Weng, S.S.M. Chow, J. Zhou, and R.H. Deng, "Conditional Proxy Broadcast Re-Encryption," Proc. 14th Australasian
Conf. Information Security and Privacy (ACISP '09), vol. 5594, pp. 327-342, 2009.
[13] D. Boneh, C. Gentry, and B. Waters, "Collusion Resistant Broadcast Encryption with Short Ciphertexts and Private Keys," Proc. Advances in Cryptology Conf. (CRYPTO '05), vol. 3621, pp. 258-275, 2005.
[14] S.S.M. Chow, Y. Dodis, Y. Rouselakis, and B. Waters, "Practical Leakage-Resilien Identity-Based Encryption from Simple Assumptions,"
Proc. ACM Conf. Computer and Comm. Security, pp. 152-161, 2010.
[15] M. Chase and S.S.M. Chow, "Improving Privacy and Security in Multi-Authority Attribute-Based Encryption," Proc. ACM Conf. Computer and Comm. Security, pp. 121-130. 2009,
[16] T.H. Yuen, S.S.M. Chow, Y. Zhang, and S.M. Yiu, "Identity-Based Encryption Resilient to Continual Auxiliary Leakage," Proc. Advances in Cryptology Conf. (EUROCRYPT '12), vol. 7237, pp. 117-134, 2012.
[17] D. Boneh, X. Boyen, and E.-J. Goh, "Hierarchical Identity Based Encryption with Constant Size Ciphertext," Proc. Advances in Cryptology Conf. (EUROCRYPT '05), vol. 3494, pp. 440-456, 2005.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY