An Improved Hybrid Re-Encryption Scheme for Mobile Cloud Computing Environment

Hasveen Kaur
Research scholar
Mtech CSE,DAVIET Jalandhar

P. S. Mann
Assistant Professor
Mtech CSE,DAVIET,Jalandhar

ABSTRACT
Cloud computing is a technology or distributed network where user can move their data and any application software on it. But there is some issues in cloud computing, the main one is security because every user store their useful data on the network so they want their data should be protected from any unauthorized access, any changes that is not done on user's behalf. There are different encryption and re-encryption techniques used for security purpose like FDE and FHE. In this paper, we have used the symmetric key agreement algorithm named Diffie Hellman, for improving cloud manager based scheme. OTP(One Time Password) is created which provides more security. The paper focuses to improve the cloud manager based scheme so that the security and integrity can be enhanced.

Keywords
Cloud Computing,Security,key Management.

1. INTRODUCTION
Cloud Computing is the environment which gives on-demand and convenient access of the system to the computing assets like storage, servers, applications, networks and alternate services which can be discharged least productivity way. Client recovers information and adjusts information which is put away by client or an organization in unified information called cloud. Cloud is a Design, where cloud administration supplier gives services to client on demand and it is otherwise called CSP remains for "Cloud Service Provider"[13]. As the protection against the malicious services or services like recognize fakes, all service provider organizations utilize the access control and client authentication components [9].

Network security, information security and many other security types like the computer security together make the term "Cloud Security". Because it consist all of the security mechanism given above. It gives the broad set of technologies, policies and controls that are used to secure the data and applications exist with the cloud computing environment [14][25]. Security is the most concerning point to any service. Only security ensures the privacy and integrity the cloud data [15]. In the last ten years, with more and more mobile electronic devices connected to Internet and continual progressing in embedded CPUs, there is an increasing demand and necessity to deploy cryptographic approaches to secure mobile platforms [16].

Cloud computing is the environment which gives on-demand and convenient access of the system to a computing assets like storage, servers, applications, networks and alternate services which can be discharged least efficiency way. The five key characteristics are made by cloud design. Cloud design likewise advances the accessibility [9]. Cloud services are mainly available in the three types of cloud which are, Public Cloud, Private Cloud and Hybrid Cloud [3][6]. Various characteristics of cloud computing are also described in the paper [7].

The cloud client applications are produced utilizing mobile application improvement platform and sent on mobile devices. The cloud client applications use the mobile network administrations, for example, wireless network (e.g. Wi-Fi, Wi-Max), cell network (e.g. 3G or 4G), or Satellite network for speaking with cloud controller. The cloud controller handles the mobile client demands for giving relating cloud administrations. It can be finished up from the investigation of come past reports that the security and privacy change in cloud administrations may build the cloud's subscribers. The essential parameters that should be considered while designing a security plan for mobile cloud processing environment are computational complexity of security plan and resource confinement of the mobile gadget. On the other hand, few security plans are concentrating on the decrease of the computational complexity of the cryptographic algorithms. Be that as it may, the decrease of the computational complexity of cryptographic algorithms may influence the privacy of the transferred information. For offloading of information access operations, the majority of the current plans depend on proxy re-encryption[18][19]. Despite the fact that the proxy re-encryption[20] plans give backing to offloading of computationally[22] concentrated re-encryption operations, the mobile client needs to play out the encryption and decryption that include huge augmentation and exponential operations of expansive numbers. Proxy re-encryption schemes are cryptosystems which allow third parties (proxies) to alter a cipher text which has been encrypted for one party, so that it may be decrypted by another. The goal of many proxy re-encryption schemes is to avoid revealing either of the keys or the underlying plaintext to the proxy, this method is not ideal. This paper proposed improved cloud-manager-based re-encryption plot that uses the attributes of the current manager-based re-encryption and cloud-based re-encryption[1] plans for distributing the computational assignments among mobile gadget, trusted-entity, and cloud [1]. To minimize the responsibilities of the manager in MReS, the Cloud-based Re-encryption Scheme (CReS) offloads the significant segment of data access operations on the cloud without revealing the security keys and data contents. Moreover, in cloud-based re-encryption conspire a single key is shared among all the group members of specific data partition on the cloud. Therefore, the current/leaving group member can unscramble the transferred data on the data partition of the cloud. In previous paper, manager and cloud based has been combined to overcome the drawbacks but in this work and we have worked on the further improvement in the technique using Elliptic Cryptography[21] on the basis of parameters like efficiency and escape time and complexity .The objective of the paper is to implement an

improved hybrid cloud manager scheme based on Elliptic Curve [21]Cryptography for re-encryption module in cloud computing environment.

2. CLOUD -MANAGER BASED SCHEME

By joining the qualities of the manager based re-encryption and cloud-based re-encryption conspires, the strategy proposed a cloud-manager-based re-encryption plan for offloading the complex computational operations on the trusted-entity and cloud. Moreover, from the exploratory results presented in next areas, this can be inferred that the energy consumption amid encryption and decryption is directly proportional to the size of the record. Increase in document size likewise increases the aggregate number of encryption and decryption operations with constant re-encryption operations. Therefore, there is a need of security plan that can offload the encryption and decryption operations on the cloud/outsider in a trusted mode. In the current CMReS, the encryption, decryption, and re-encryption assignments are appropriated between the trusted entity and cloud. There are four fundamental modules in this system, to be specific

(a) Cloud client application facilitated on the mobile users,

(b) Encryption/Decryption Service Provider (EDSP) module facilitated on private cloud inside the client association,

(c) Re-encryption Service Provider (RSP) module facilitated on public cloud, and

(d) Cloud storage services accessible on public cloud.

The cloud service provider offers computational and storage services to the mobile users. The mobile users upload/download the data to/from the data partition of the cloud through the cloud client application [1].

The EDSP is a completely trusted entity under the control of a client association whose prime responsibility is to give encryption and decryption services to the authorized mobile users. The RSP module is hosted on public cloud which is responsible for keeping up the re-encryption keys and giving the re-encryption services to each authorized mobile user. The RSP module just holds the re-encryption keys of the cloud users having a place with the same virtual association for giving re-encryption services. The exceptional feature of the plan is that the RSP is hosted on the cloud and gives re-encryption services without knowing the private keys of the mobile users. In the base paper the AES is the encryption algorithm which is based on public and private keys. The data which is given a input is divided into 64 blocks and each block is treated individually,

The 8 passes are executed and after each pass new key is generated which is used for the encryption of the data. Due to high number of passes and key generation after each phase complexity of the AES algorithm is high which leads to increase execution time of the algorithm

3. PROPOSED TECHNIQUE

There are many encryption algorithms to provide security to the cloud. "Fully Homomorphic" is more reliable. It gives more privacy and security as compare to scheme of "Full Disk Encryption". The main problem which is there in Fully Homomorphic Encryption is a key storage, key management, Access control and Data Aggregation list maintaining. To solve problem of Key management, Key Sharing various schemes have been proposed in last years. The various security attacks are possible in these schemes. The third party auditor is the scheme for key management and key sharing. The third party auditing scheme will be failed, if the third party's security is compromised or of the third party will be malicious. To solve this problem, In this thesis we will work on to design new model for key sharing and key management in fully Homomorphic Encryption scheme. In this work, we find that fully homomorphic encryption[17] technique is more efficient than full disk encryption. But the main problem exists in fully homomorphic encryption is of key management and key sharing which reduces the reliability of the scheme. For key management and key sharing, enhancement has been proposed in the encryption scheme and enhancement is based on Diffie-hellman algorithm[12] and HMAC and OTP is generated on the basis of secret key generated from diffie-hellman algorithm. This algorithm create session key between user and cloud. Each time new key is generated between two before communication. In the proposed work, One Time Password is generated on the basis of key k(number of login times).If the user is same OTP is generated, the rights are reserved. This will also improve the security in hybrid scheme.

Elliptic curve cryptography (**ECC**) is an approach to public-key cryptography based on the algebraic structure of elliptic curves over finite fields. ECC requires smaller keys compared to non-ECC cryptography (based on plain Galois fields) to provide equivalent security. Elliptic curves are applicable for re- encryption, digital signatures, pseudo-random generators and other tasks. This based on Diffie Hellman scheme.

One Time Password (OTP): Password is used for authentication by all the business and organization. Moreover Static passwords have many limitations. Password can be get hacked. Lackadaisical employee may note down passwords somewhere, system with saved passwords may be used by various users or a malicious user may reset all passwords just to create havoc. So it is very useful to use dynamic password i.e. one time password. Dynamic passwords are more secure as compared to static. There is no need to write down these passwords and remember these passwords. For each login session each time a new password is generated. One time passwords are more reliable and user friendly as well for authentication. OTP generation can be done by various OTP generation algorithms for generating strings of passwords. OTP ensures safety. This leads to authenticating them again and again over the period of time for each login session. To avoid the overhead we can use OTP for multi cloud environment.

3.1 Algorithm

selected node suppose user1

1. Login
2. Key generation
   - 2.1 Enter prime numbers
   - 2.2 Enter random numbers by client and cloud service provider
   - 2.3Secret key generation and securec hannel establishment

3. OTP (One Time Password) generation

3.1 cloud server will set count1=0, count2=0...count5=0 for respective user at its side.

3.2 Cloud Server will request for the OTP from

user 1

3.3 user1 enter (secret key+count) as OTP

3.3 server match it because server knows both secret key and count of each user.

3.3.1: count1++; // so for user 1 it will be count1=1; for remaining user their
count will be still 0;

3.3.2 if (secret_key+count(x) ==
secret_key+count(y))

{ Access granted;

Display message by server : print ("please enter the operation");}

else{ display message by server: print(" wrong password, your login number is count1);}
4. client will enter the operation using HMAC digest :
4.1 hmac(already generated secret key || v, file1,ver1 || sha1)
{
if(open==v)
{
server will check the file name and version;
if (file1,ver1== file1,ver1)

{ printf("file is valid"); }

else { print (file is invalid, please replace the file)

}}
if(open==I) { insert new file file2 }
5. encryption/decryption
6.data operation
7.logout;

note: // 1.at client side, user will enter prime number, random number for generating secret keys, once generating secret key user will enter OTP number, after inserting OTP number, user will enter operation(Insertion) with corresponding file name(file1 or file2).

3.2 Eperimental Results

The parameters which we have considered are the generally parameters like complexity, execution time and resource utilization. In the thesis, we have designed the efficient system in which values of complexity , execution time and resource utilization is less as compared to existing technique

3.2.1 Complexity

Many attempts have been made to increase the efficiency[23] [24]and the integrity is maintained. Also we want that resources can be used to upto maximum .In this work, the sharing of keys and secure channel is established between both i.e. user and the cloud service provider. It is analyzed that Space utilization of proposed technique is less than existing technique..Complexity =(number of phases used for encyprtion/size of the data)*time is the formula which is used to calculate complexity of the system or we can say The space used is in bytes which describe the complexity..

X axis represents number of nodes for cmparison between 5,8 and 10 nodes.

Y Axis represents complexity in bytes

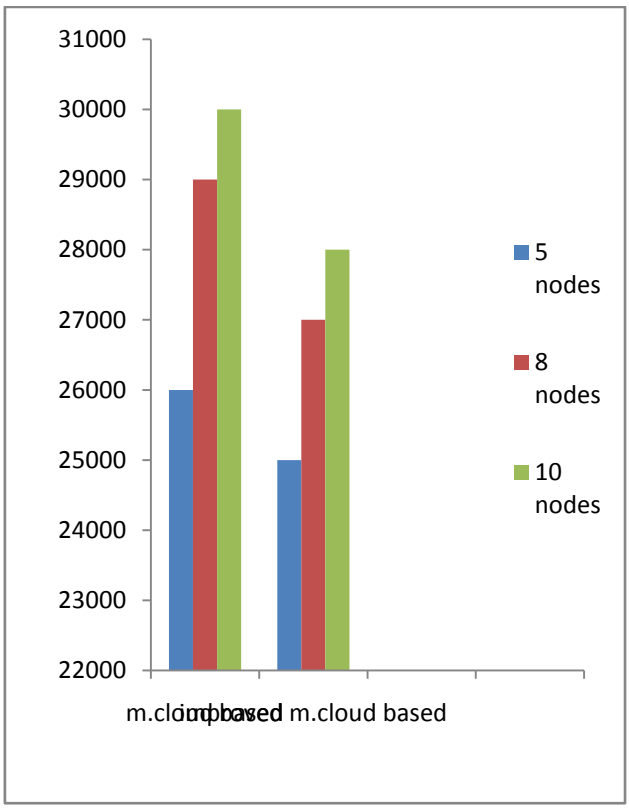

Fig. 1 The graph shows the comparison for Complexity in bytes between existing cloud manager based technique and proposed technique for 5 nodes,8 nodes and 10 nodes.

3.2.2 Time of Encryption

As shown in figure 2, the comparison between previous and proposed approach is shown in terms of delay.The time of encryption has been decreased in the proposed technique. The comparison between previous and proposed approach is shown in terms of delay. The delay in previous technique is increasing, when numbers of exchange messages are increased. In the proposed approach the delay is less due to the technique used. The delay is the time taken by the nodes to complete task which is assigned.

The execution time is reduced because the proposed technique is based on diffie-helman and OTP technique in which key generation procedure is not so complex as AES . That why in our technique execution time is reduced Time is calculated by subtracting time at which task completed and task at which task is start executed. The graph clearly shows that there is the improvement in the time of encryption .Though the nodes increase but the time of encryption does not increase at large scale..The time taken is in milliseconds. The AES is the encryption algorithm which is based on public and private keys. The data which is given a input is divided into 64 blocks and each block is treated individually, The 8 passes are executed and after each pass new key is generated which is used for the encryption of the data. Due to high number of passes and key generation after each phase complexity of the AES algorithm is high which leads to increase execution time of the algorithm . The graph shows the comparison for Time of Encryption in ms between existing cloud manager based technique and proposed technique for 5 , 8 and 10 nodes

The time of encryption may vary every time for each nodes because the OTP generated is dynamic..So this is not necessary for time of encryption to remain same .

X axis represents number of nodes for cmparison between 5,8 and 10 nodes.

Y Axis represents time of encryption in miliseconds

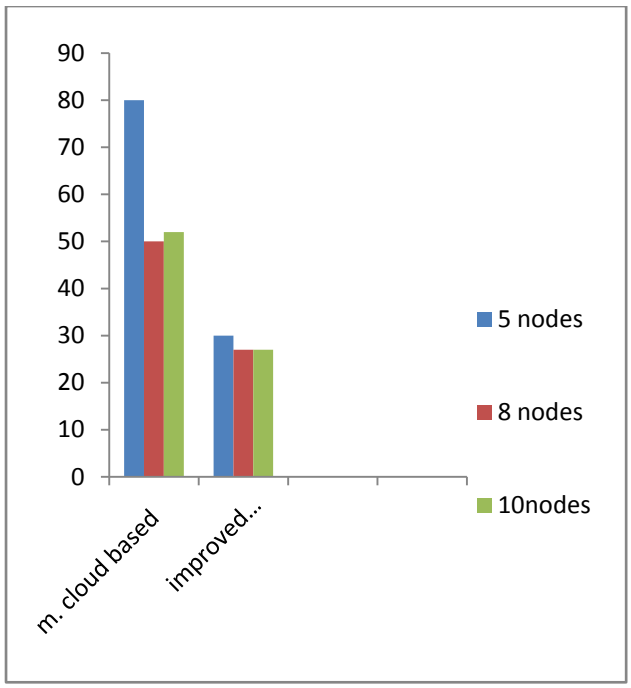

Fig. 2 The graph shows the comparison for Time of Encryption in ms between existing cloud manager based technique and proposed technique for 5 , 8 and 10 nodes

3.2.3 Efficiency

The comparison is made between efficiency of existing and proposed algorithm. It is analyzed that probability of proposed technique is less than existing algorithm which is in percentage .This graph shows the improvement in efficiency. The execution time is reduced therefore efficiency is increased because the proposed technique is based on diffie-helman and OTP technique in which key generation procedure is not so complex as AES . For key management and key sharing, enhancement has been proposed in the encryption scheme and enhancement is based on Diffie-hellman algorithm and HMAC and OTP is generated on the basis of secret key generated from diffie-hellman algorithm.

This algorithm creates session key between user and cloud. Each time new key is generated between two before communication. This reduces the time takes place in management and sharing of keys and secure channel is established between both i.e. user and the cloud service provider. The simulation shows that proposed enhancement is more efficient and reliable than the existing one.The graph shows the comparison for Efficiency in percentages between existing cloud manager based technique and proposed technique for 5 nodes,8 nodes and 10 nodes.

X axis represents number of nodes for cmparison between 5,8 and 10 nodes.

Y Axis represents efficiency in terms of number of resources

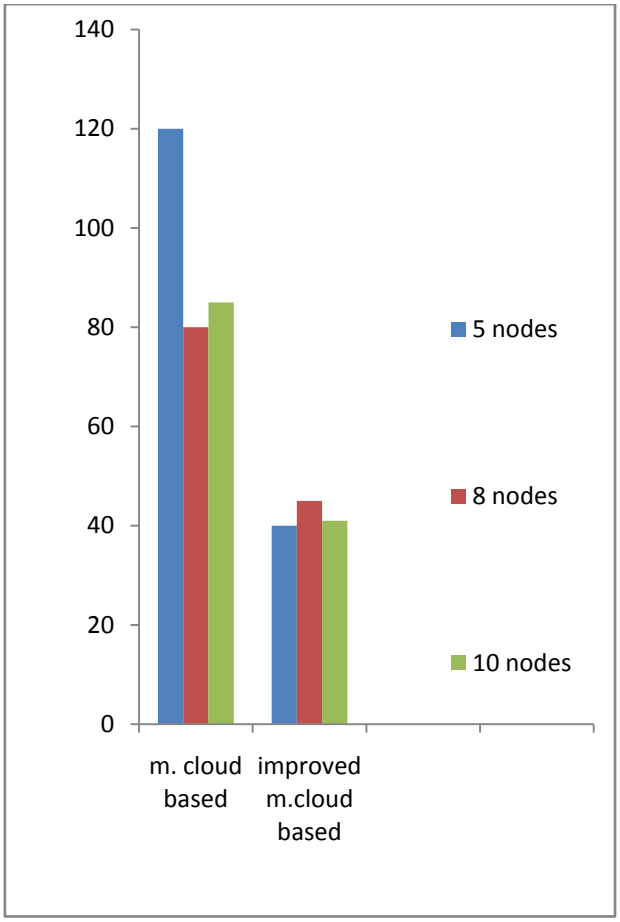

Fig. 3 The graph shows the comparison for Efficiency in percentages between existing cloud manager based technique and proposed technique for 5 nodes,8 nodes and 10 nodes

4. CONCLUSION

Cloud computing is the environment which provides on-demand & convenient access of the network to a computing resources like storage, servers, applications, networks and the other services which can be released minimum efficiency way. In this user can store their data and use different services and pay according to those services. The main factor is security that how we can store our data while storing into the cloud. In this thesis, we reviewed two most popular techniques for cloud data encryption. These techniques are full disk encryption and fully homomorphic encryption in manager and cloud based.. In this work, we find that fully homomorphic encryption technique is more efficient than full disk encryption. But the main problem exists in fully homomorphic encryption is of key management and key sharing which reduces the reliability of the scheme. For key management and key sharing, enhancement has been proposed in the encryption scheme and enhancement is based on Diffie-hellman algorithm and HMAC and OTP is generated on the basis of secret key generated from diffie-hellman algorithm. This algorithm creates session key between user and cloud. Each time new key is generated between two before communication.

This reduces the time takes place in management and sharing of keys and secure channel is established between both i.e. user and the cloud service provider. The simulation shows that proposed enhancement is more efficient and reliable than the existing one.

5. REFERENCES

[1] Abdul Nasir Khan, M. L. Mat Kiah · Mazhar Ali, Shahaboddin Shamshirband, Atta ur Rehman Khan," A Cloud-Manager-Based Re-Encryption Scheme for Mobile Users in Cloud Environment: a Hybrid Approach", 2015 Springer Science + Business Media Dordrecht

[2] Ankur Mishra, Ruchita Mathur, Shishir Jain, Jitendra Singh Rathore, 2013 "Cloud Computing Security" International Journal on Recent and Innovation Trends in Computing and Computation, pp 36-39

[3] Bhavna Makhija, VinitKumar Gupta, 2013 "Enhanced Data Security in Cloud Computing with Third Party Auditor", International Journal of Advanced Research in Computer Science and Software Engineering, pp 341-345

[4] Dawn Song, Elaine Shi, 2012 "Cloud Data Protection for the Masses" IEEE Computer Society, pp 39-4

[5] Deyan Chen, Hong Zhao, 2012" Data Security and Privacy Protection Issues in Cloud Computing" International Conference on Computer Science and Electronics Engineering, pp 647-651

[6] Dr Nashaat el-Khameesy, Hossam Abdel Rahman, 2012 "A Proposed Model for Enhancing Data Storage Security in Cloud Computing Systems" vol-3

[7] John Harauz, Lori M. Kaufman,Bruce Potter," Data Security in the World of Cloud Computing " IEEE Security and Privacy July 2009. pp. 61-64

[8] Sanjoli Singla, Jasmeet Singh, 2013 "Cloud Data Security using Authentication and Encryption Technique" International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 2, Issue 7, July 2013, pp 2232-2235

[9] Punithasurya K, Esther Daniel, Dr. N. A. Vasanthi, 2013 "A Novel Role Based Cross Domain Access Control Scheme for Cloud Storage" International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 2, Issue 3, March 2013, pp 942-946

[10] Vimmi Pandey, 2013 "Securing the Cloud Environment Using OTP" International Journal of Scientific Research in Computer Science and Engineering vol-1, Issue-4

[11] Young-Gi Min, Hyo-Jin Shin and Young-Hwan Bang, 2012 "Cloud Computing Security Issues and Access Control Solutions" Journal of Security Engineering, pp 135-140

[12] ElGamal, T.: *A public key cryptosystem and a signature scheme based on discrete logarithms.* In: Advances in Cryptology, pp. 10–18. Springer,1985

[13] Tysowski, P.K., Hasan, M.A.: *Re-encryption-based key management towards secure and scalable mobile applications in clouds.* IACR Cryptology ePrint Archive 668, 2011

[14]Hashemi, S.M., Ardakani, M.R.M.: *Taxonomy of the security aspects of cloud computing systems-a survey*. Int. J.Appl. Inf. Syst. 4, 21–28 ,2012

[15] Kumar, K., Lu, and Y.H.: Cloud computing for mobile users: *Can offloading computation save energy?* Computer 43,51–56 ,2010

[16]De Caro, A., Iovino, V.: "jPBC: *Java pairing based cryptography* ,presented at the IEEE Symposium on Computers and Communications (ISCC '11) Kerkyra ,2011

[17] *Secure Cloud Computing through Homomorphic Encryption,* Available at: http://arxiv.org/abs/1409.0829,2013

[18] Ateniese, G., Fu, K., Green,M., Hohenberger, S.: *Improved proxy re-encryption schemes with applications to secure distributed storage.* ACM Trans. Inf. Syst. Secur. (TISSEC)9, 1–30 ,2006

[19] Green, M., Ateniese, G.: *Identity-based proxy re-encryption*, presented at the Applied Cryptography and Network Security (ACNS '07), Zhuhai, China ,2007

[20] Ivan, A., Dodis, Y.: *Proxy cryptography* revisited, presented at the Proceedings of the Network and Distributed System Security Symposium (NDSS '03), San Diego, California,2003

[21]Certicom, Standards for Efficient Cryptography,SEC 2*: Recommended Elliptic Curve Domain Parameters*, Version 1.0. Available: at http://www.secg.org/download/aid-386/sec2 final.pdf ,2000.

[22] Kumar, K., Lu, and Y.H.: Cloud computing for mobile users: *Can offloading computation save energy?* Computer 43,51–56 ,2010

[23]Itani, W., Kayssi, A., Chehab, A.: *Energy-efficient incremental integrity for securing storage in mobile cloud computing*, presented at the International Conference on Energy Aware Computing (ICEAC '10) Cairo, Egypt ,2010

[24]Zhou, Z., Huang, D.: *Efficient and secure data storage operations for mobile cloud computing*, presented at the 8th International Conference on Network and Service Management (CNSM '12), AZ, USA ,2012

[25] Hashemi, S.M., Ardakani, M.R.M.: *Taxonomy of the security aspects of cloud computing systems-a survey*. Int. J.Appl. Inf. Syst. 4, 21–28 ,2012

[26] https://en.wikipedia.org/wiki/Proxy_re-encryption