Novel approach of hybrid encryption by ECDH and optimize blow fish for cloud data

Chimpy Salgotra, Mohit Marwaha, Sanjeev Mahajan
Student, Assistant Professor, Associate Professor
[1]Student, Pathankot, India

Abstract **-** Security aspects in this field which are remain at the core of interest. The aim of this research is to identify and solve the security issues related to the cloud computing. A detailed literature review is done to study the approaches and concepts used in the field of cryptography on cloud environment. In this work input is given by using Bench mark dataset and then key is created by using encryption algorithmRSA and MD5 then concatenate the keys. Optimize RSA algorithm is hybrid with MD5 to optimize the results. After this decryption process is also performed for decode the data. At last performance evaluation of the proposed system is done by using analysis of storage and time consumed during encryption and decryption process. The results show the attack is reduced by RSA approach with integrity MD5

I. INTRODUCTION

Nowadays, Cloud computing is a growing area in distributed computing that deliver dynamically adaptable services on demand over the internet through virtualization of hardware and software. The biggest advantage of the cloud is its flexibility to lease and release resources as per the user requirement. Furthermore, the cloud provider offer two type of plans namely short term plan on demand and long term reservation plan. It has got intelligent infrastructure i.e. Transparency, Scalability, Monitoring and Security.

The developing worldview of distributed computing gives another approach to address the limitations of constrained vitality, capacities and assets. Be that as it may, security and security insurance is a basic worry in the advancement and selection of distributed computing. To dodge framework delicacy and guard against vulnerabilities from digital assailant, different digital security apparatuses and methods were created. Contrasted and the conventional IT show, the distributed computing has numerous potential points of interest. Be that as it may, from the purchasers' point of view, distributed computing security concerns remain a noteworthy hindrance for the reception of distributed computing.

There are many administrations on web yet to use the client need to make their record and afterward they can begin utilizing the administrations. Record capturing is regular factor in cloud. In some cases because of programming vulnerabilities, trafficking and cradle flood it might happen. This all hazard may prompt loss of control over their record. A criminal deal with client record can listen stealthily on exchange, control information, and give false reactions to clients. This hazard trade off with secrecy.

At the point when client utilizes administrations of cloud computing, they may require some private data like Visa data. At the point when typical handling occurs by then of time it might conceivable that some unapproved client may burglary the classified data and they can abuse the data. Along these lines, there is danger of information break in cloud computing.

II. RELATED STUDY

Hong, Ming-Quan et al. proposed Elliptic Curve Cryptography (ECC) based homomorphic encryption plot for SMC issue that is drastically diminished calculation and correspondence cost. It demonstrates that the plan has focal points in vitality utilization, correspondence utilization and security assurance through the correlation analyze between ECC based homomorphic encryption and RSA&Paillier encryption calculation. Additional proof, the plan of homomorphic encryption conspires in light of ECC is connected to the estimation of GPS information of the seismic tremor and demonstrate it is demonstrated that the plan is achievable, superb encryption impact and high security[1]. Penn, Georg M., et al. computing the Hamming weight between parallel biometric highlight vectors in a homomorphic encryption space can be somewhat wasteful because of the required piece savvy encryption. A biometric coordinating procedure more productive than the Goldwasser-Mlcali approach is proposed in light of abusing Paillier's capacity of scrambling messages bigger than one piece at any given moment. The productivity is recorded in an iris ID setting [2].

Patel, Sankita J., et al. [3] they proposed Elliptic Curve Cryptography (ECC) based approach for SMC that is versatile as far as computational and correspondence cost and keeps away from TTP. In writing, there do exist different ECC based homomorphic plans and it is basic to explore and dissect these plans with a specific end goal to choose the appropriate for a given application. In this paper, they exactly dissect different ECC construct homomorphic encryption plans situated in light of execution measurements, for example, computational cost and correspondence cost. They suggest an effective calculation among a few chose ones, which offers security with lesser overheads and can be connected in any application requesting protection. Chatterjee, Ayantika et al. [4] They provided techniques to translate basic operators (like bitwise, arithmetic and relational operators), which are used for implementation of algorithms in any high level language like C. Subsequently, they addressed decision making and loop handling and related data structures which are vital to realize when the controlling variables are encrypted. Since, termination is a majorchallenge while handling encrypted data; they proposed a method of handling termination by message passing between server and client. . Fully Homomorphic Encryption (FHE) provides a method of performing arbitrary operations directly on encrypted data. This seemingly magical idea is a welcome to cloud computing. However, there are several challenges to overcome for making the technology viable in practical applications. In this paper, they made an initial effort to highlight the problem of translating algorithms that can run on unencrypted or normal data to those which operate on encrypted data. Here, they showed that although FHE provides the ability to perform arbitrary computations, its complete benefit can only be obtained if they also allow executing arbitrary algorithms on encrypted data.

Zhang, Zhongxing et al.[5]In view of the Fuzzy C-Means calculation, they proposed the PSO-FCM calculation consolidating the Fuzzy C-Means with PSO. At that point KDD container 99 dataset was connected to calculation, the tests show that the PSO-FCM calculation can keep away from the innate weaknesses of the FCM calculation, and has higher location execution with discovery rate rising and false caution rate falling. What's more, they contrast the execution of PSO-FCM and other grouping; it can be more tasteful outcomes. Manjula, S., et al. [6] The proposed work will help to achieve data privacy using division of data. The user data is encrypted using RSA and divided into multiple blocks and stored on different cloud servers. The Cloud Manager (CM) has the responsibility of storing the fragmented file in cloud server. In this proposed work the data cannot be processed by CM this increases the confidentiality of data. The division of data in the cloud environment for secure data storage achieves security as well as privacy to user data and enhances the performance by replication techniques.

Marinelli, Franca, et al [7] The correctness in decrypting a cipher text after some operations in the DGVH scheme depends heavily on the dimension of the secret key. In this paper they compute two bounds on the size of the secret key for the DGHV scheme to decrypt correctly a cipher text after a fixed number of additions and a fixed number of multiplications. Moreover they improved the original bound on the dimension of the secret key for a general circuit. Gentry, Craig et al [8]they described a few concrete instantiations of the new method, including a "simple" FHE scheme where we replace SSSP with Decision Diffle-Hellman, an optimization of the simple scheme that let us "compress" the FHE cipher text into a single Elgamalciphertext(J), and a scheme whose security can be (quantum) reduced to the approximate ideal-SIVP. They stress that the new approach still relies on bootstrapping, but it shows how to bootstrap without having to "squash" the decryption circuit. The main technique is to express the decryption function of SWHE schemes as a depth-3 Q2 (Σ Π Σ) arithmetic circuit of a particular form. When evaluating this circuit homomorphically (as needed for bootstrapping), they temporarily switch to a MHE scheme, such as Elgamal, to handle the Π part. Due to the special form of the circuit, the switch to the MHE scheme can be done without having to evaluate anything homomorphically. Then they translate the result back to the SWHE scheme by homomorphically evaluating the decryption function of the MHE scheme. Using their method, the SWHE scheme only needs to be capable of evaluating the MHE scheme's decryption function, not its own decryption function. Thereby avoid the circularity that necessitated squashing in the original blueprint.

Gentry, Craig et al. presented a simpler approach that bypasses the homomorphic modular-reduction bottleneck to some extent, by working with a modulus very close to a power of two. Their method is easier to describe and implement than the generic binary circuit approach, and they expected it to be faster in practice (although they did not implement it yet). In some cases it also allows to store the encryption of the secret key as a single cipher text, thus reducing the size of the public key [9].Lepoint, Tancrède et al. introduced a specific algorithm for 2-level encryption (first generation of FHE schemes) and an extended algorithm for ℓ_{max}-level encryption with arbitrary $\ell_{max} \geq 2$ to cope with more recent FHE schemes. They successfully applied their method to a range of real-world circuits that perform various operations over plaintext bits. Practical results show that some of these circuits benefit from significant improvements over the naive evaluation method where all multiplication outputs are bootstrapped[10].

III. PROPOSED METHODLOGY

Algorithm Used

Blowfish has a variable length key of 332-448 bits and it is a 64-bit square figure. The two techniques comprised in Blowfish calculation is: introducing the key and the stage in which information is scrambled. A client variable key is consumed in the first stage to sub-key varieties of 4168/8336-byte, which is given component clusters size of 4-byte or arrangement exhibits component size of 8-byte. The sub-key clusters (P section 18 and four S exhibits passage 256) era process is client key ward. There is an expansion in the security level with the intricacy upgrade of sub-keys and client key connection. In encryption handle of late, in spite of client key, sub-keys that are refreshed are utilized. The contribution of 32-bit is split into the contribution of 4 eight-piece quarters to S-boxes. The secluded 232 is added to yield and for delivering the last yield of 32-bit, XOR is utilized. Feisrel system is embraced by the blowfish for emphasizing the 16 (rounds) times a straightforward capacity of encryption. So also for Blowfish unscrambling, contribution to the starting taken as cipher text. In the turnaround request is utilized P1-P18 which is the fundamental distinction.

Proposed Methodology steps

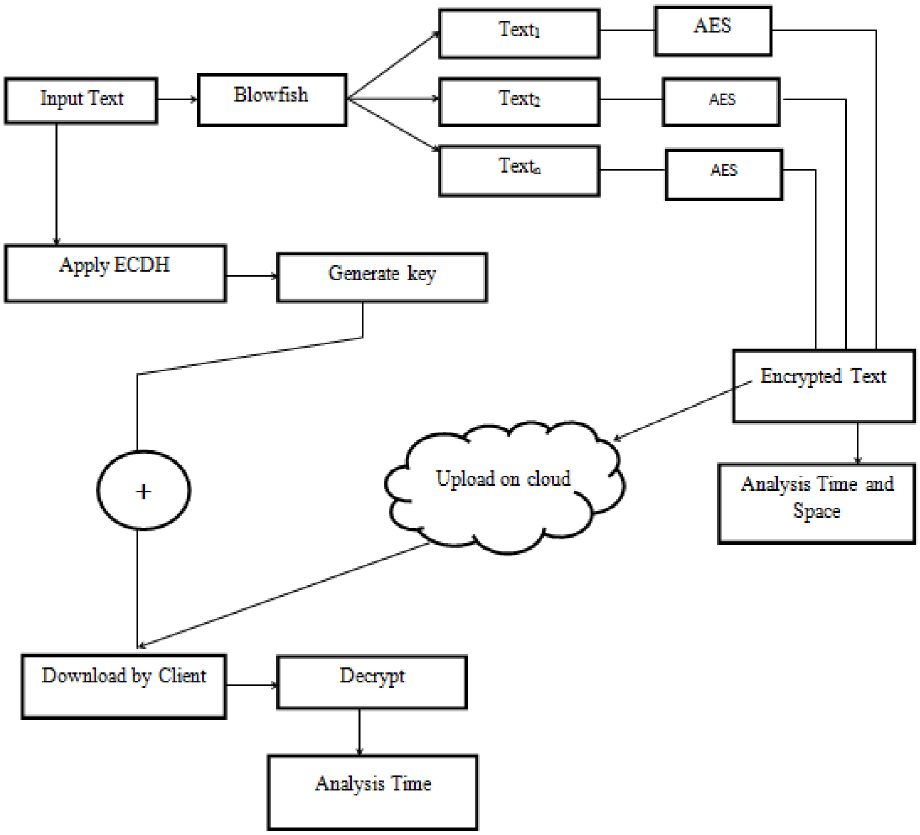
Step 1: Input the text file by bench mark data set.
Step 2: Generate the key by ECDH (Elliptic curve and Deffi Hell Men method) and concatenate them.
Step 3: After key generation apply the encryption algorithm in our case use Blowfish Hybrid with AES algorithm.

- First we encrypt by AES algorithm.
- Then make the slices and these slices optimized by Meta- Heuristic algorithm.
- Apply Blowfish on these slices parallely.

Step 4: After encryption upload the data on cloud
First encrypted data will send to cloud let. Cloud let scheduled by broker.
Broker scheduled data storage on virtual machine.
Step 5: After encrypt data storage start decryption step

- First download the data from cloud by client.
- Then client key which generate in 2 step decrypt the data.

Step 6: After decryption calculate the time and storage.



IV. RESULTS AND DISCUSSION

In this section comparative analysis of the result is performed on hybrid encryption algorithms. The table given below represents the encryption file size of comparison between ECDH Blowfish and RSA-MD5 encryption algorithm.

Table 4.1 Encryption Files with name and Decryption Size

File name	Size	Decryption size
Dataset.text	1315331 Bytes	2203647
new1.text	1057393Bytes	1057393
new2.text	1058426 Bytes	1058426
new3.text	686414 Bytes	686414
new4.text	2393301Bytes	2393301

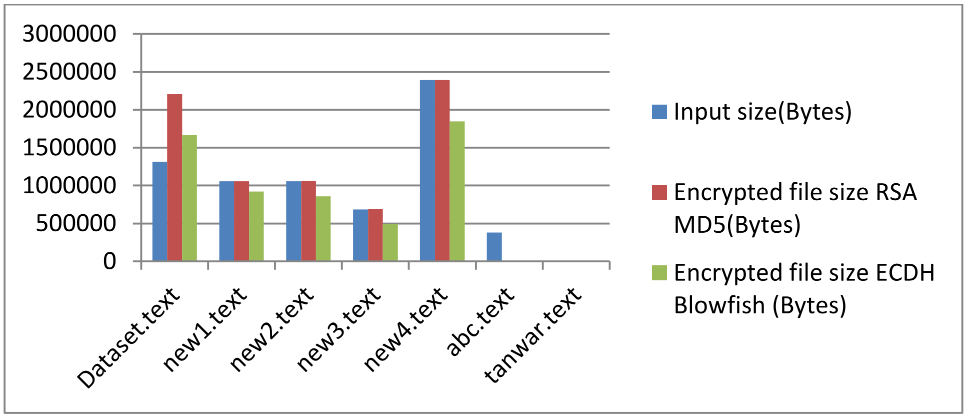


Figure 1.1: Comparison table of encryption file size between Blowfish-MD5 and ECDH-AES algorithm

The figure 1.1 shows the comparison of RSA_MD5 and Blow Fish_MD5 with file size. It shows the after encryption effect on size.

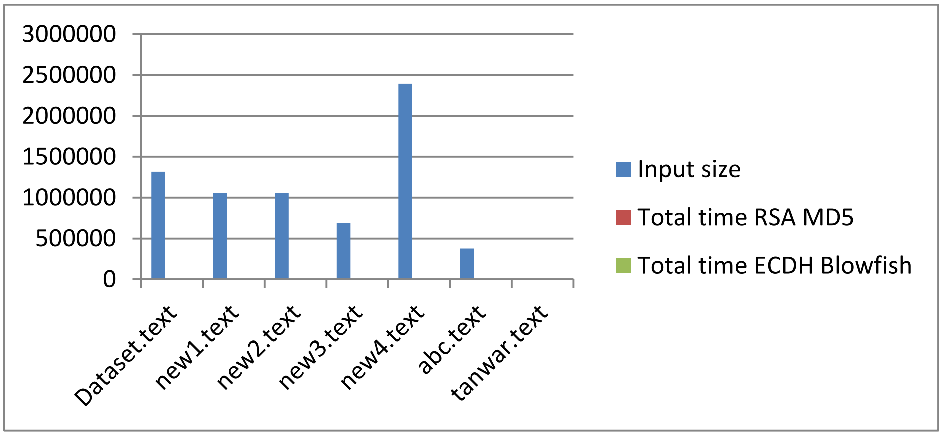


Figure 1.2 depicts the total time taken by the both algorithms in encryption and decryption time. The time consume by hybrid ECDH Blowfish and RSA MD5 algorithm.

A comparative analysis of the result is performed on hybrid encryption algorithms. In this analysis encryption time and decryption time in various algorithms is compared. Encryption and decryption time in Blow Fish MD5 is less than the existing method.

V. CONCLUSION

In this thesis, the work is based on the security on cloud by using hybrid algorithm. The comparison is shown between hybrid Optimize RSA-MD5 algorithm (proposed hybrid cryptographic algorithm) and RSA algorithm. The experimental results obtain shows that the proposed algorithm have lesser encryption and decryption time and needs less storage capacity in comparison to RSA algorithm. With future prominence is given to the proposed architecture implementation comparing with different algorithm to show their effectiveness.

VI. REFERENCES

[1]. Hong, Ming-Quan, Peng-Yu Wang, and Wen-Bo Zhao. "Homomorphic Encryption Scheme Based on Elliptic Curve Cryptography for Privacy Protection of Cloud Computing." *Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS), 2016 IEEE 2nd International Conference on*.IEEE, 2016.

[2]. Penn, Georg M., et al. "Customisation of Paillierhomomorphic encryption for efficient binary biometric feature vector matching." *Biometrics Special Interest Group (BIOSIG), 2014 International Conference of the*.IEEE, 2014.

[3]. Patel, Sankita J., AnkitChouhan, and Devesh C. Jinwala. "Comparative evaluation of elliptic curve cryptography based homomorphic encryption schemes for a novel secure multiparty computation." *Journal of Information Security* 5.01 (2014): 12.

[4]. Chatterjee, Ayantika, and IndranilSengupta."Translating Algorithms to handle Fully Homomorphic Encrypted Data on the Cloud." *IEEE Transactions on Cloud Computing* (2015).

[5]. Zhang, Zhongxing, and BaopingGu."Intrusion Detection Network Based on Fuzzy C-Means and Particle Swarm Optimization." *Proceedings of the 6th International Asia Conference on Industrial Engineering and Management Innovation*. Atlantis Press, Paris, 2016.

[6]. Manjula, S., M. Indra, and R. Swathiya."Division of data in cloud environment for secure data storage." *Computing Technologies and Intelligent Data Engineering (ICCTIDE), International Conference on*.IEEE, 2016.

[7]. Marinelli, Franca, et al. "Some security bounds for the key sizes of DGHV scheme." *Applicable Algebra in Engineering, Communication and Computing*25.5 (2014): 383-392.

[8]. Gentry, Craig, and ShaiHalevi."Fully homomorphic encryption without squashing using depth-3 arithmetic circuits." *Foundations of Computer Science (FOCS), 2011 IEEE 52nd Annual Symposium on*.IEEE, 2011.

[9]. Gentry, Craig, ShaiHalevi, and Nigel P. Smart."Better bootstrapping in fully homomorphic encryption." *International Workshop on Public Key Cryptography*.Springer, Berlin, Heidelberg, 2012.

[10]. Lepoint, Tancrède, and Pascal Paillier."On the minimal number of bootstrappings in homomorphic circuits." *International Conference on Financial Cryptography and Data Security*.Springer, Berlin, Heidelberg, 2013.