IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

ADVANCED CLOUD DATA SECURITY FEATURES UNDER KEY VULNERABILITY

Muhammad Abul Kalam[1] , Guguloth Ravinder[2]

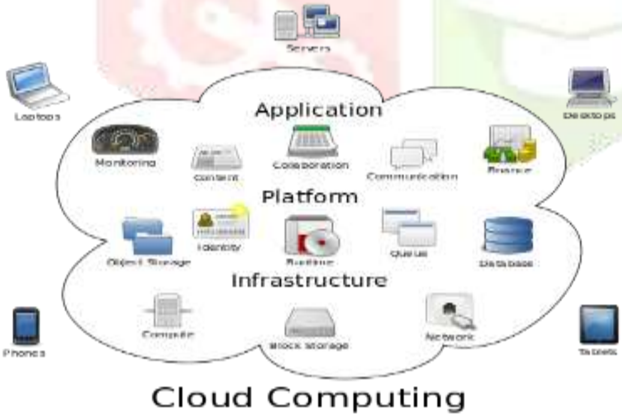
[1,2] Department of Computer Science and Engineering, Holy Mary Institute of Technology & Science, Hyderabad,

Abstract: Data confidentiality can be broken by powerful attackers by using cryptographic keys. This hacking can be done by coercion or backdoor in cryptographic software. if the encryption key is visible to the hacker then total data will be hacked so the only way to preserve the data confidentiality is to limit the attackers to access the cipher text. This may be done by using multiple administrative servers to share the cipher text so we can assume that the adversary cannot compromise all of them. If the data is encrypted by using the present method hackers can easily attack the data. So in this paper we are implementing data confidentiality against an adversary who knows the encryption key and accessing a large amount of cipher text. So we propose Bastion, which is very efficient and more guaranteed in data confidentiality. If the encryption key is visible to the adversary and he cannot access the whole data. Bastion is very secure and well suited for integration in the present system because it gives less than 5% overhead and also it implements secure encryption modes. We evaluated the performance of bastion by using prototype implementation.

***Index Terms* -** cryptography key, encryption, Bastion.

I. INTRODUCTION

One of the best computing resources that are delivered as a service by using the network is cloud computing. It is a cloud shaped symbol so it is called cloud computing. Cloud computing can give more confidentiality of data and also remote services to user data. Cloud computing is a third party service, it consists of software and hardware. This is available on the internet. These third party services provide more advanced access to the software applications and high end networks.



II. EXISTING SYSTEM

In the existing system if there is an exposed encryption key the adversary can access the cipher text. e.g., by sharing that key with all administrative dominoes. If data is encrypted and shared with all domains then an adversary tries with appropriate keying material and can comprise the server in one single domain and decrypt the total cipher text blocks.

Disadvantages of Existing System

An exciting scheme of encryption is AON which is two rounds of block cipher encryption method for each data. The first pre-processing step is to create AONT which is again followed by another second round of actual encryption and both must be sequential. so it leads to overhead for large data and unacceptable for some times.

III. PROPOSED SYSTEM

Here we are concentrating on data confidentiality against an adversary who already knows and has access for large fractions of cipher text blocks. An adversary can get the key by using some flaws or other back door software that provides key generation techniques. To overcome this problem we propose one technique that is Bastion. It is a very efficient and new technique. it is very secure that the plain text data cannot be recovered as long as adversaries have all access to cipher text blocks. This is achieved by using Bastion which uses standard encryption function. It is a very useful linear transform that is similar to notion of all or nothing transform.

Advantages of proposed system

We examine all other numbers of exciting encryption methods. Our result shows that bastion is only one best method which gives very less performance deterioration. and it also increases the performance of AON encryption methods. It prevents the leakage of plain text and is more secure compared to other techniques. We discuss practical insights with respect to the deployment of Bastion within existing storage systems, such as the Hydrator grid storage system.

IV. MODULES:

- Data Owner
- Data User
- Admin

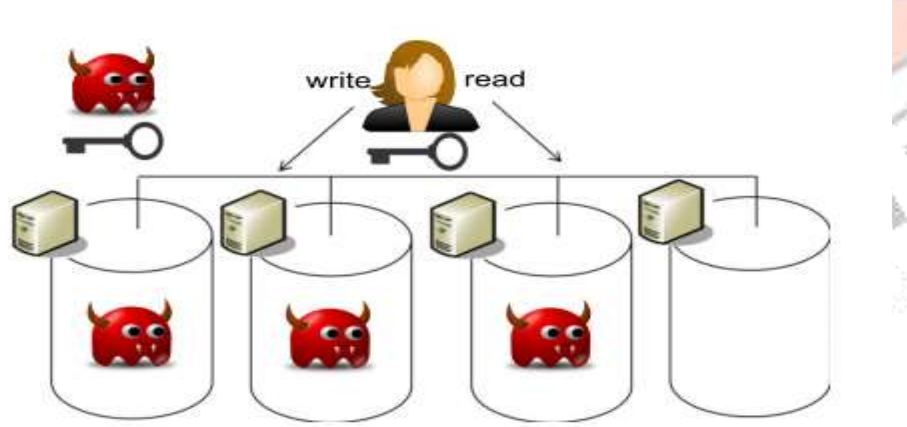
Data Owner: In Data Owner module, Initially Data Owner must have to register their detail and admin will approve the registration by sending signature key and private key through email. After successful login he/she have to verify their login by entering signature and private key. Then data Owner can upload files into cloud server with Polynomial key generation. He/she can view the files that are uploaded in cloud by entering the secret file key.

Data User: In Data User module, Initially Data Users must have to register their detail and admin will approve the registration by sending signature key and private key through email. After successful login he/she have to verify their login by entering signature and private key. Data Users can search all the files upload by data owners. He/she can send search request to admin then admin will send the search key. After entering the search key he/she can view the file

Admin: In Admin module, Admin can view all the Data owners and data user's details. Admin will approve the users and send the signature key and private key to the data owners and data users. Admin can also send search requests to users. Admin can able see the files in cloud uploaded by the data owners.

V. SYSTEM ARCHITECTURE

Below picture shows the architecture and here if any threat occurred then the user can easily identify and solve that issue.



VI. CONCLUSION

Here we explained and addressed the problem of securing confidentiality of data which is outsourced to the cloud and all adversaries can access the encryption key. We implemented one new technique that captures data confidentiality against any adversary. For that we proposed Bastion. It makes sure the confidentiality of encrypted data even the adversary has an encryption key, and all but two cipher text blocks. Bastion is more sustainable for cipher text settings because it uses multi cloud storage systems. Here in this setting the adversary needs to get an encryption key to compromise all servers. Bastion improves the performance by 50% and overhead of security is very negligible that is less than 5%. Finally, we implemented how Bastion can work with existing storage systems.

VII. REFERENCES

M. Abd-El-Malek, G. R. Ganger, G. R. Goodson, M. K. Reiter, and J. J. Wylie, "Fault-Scalable Byzantine Fault-Tolerant Services," in ACM Symposium on Operating Systems Principles (SOSP), 2005, pp. 59–74.

M. K. Aguilera, L. Xu and R. Janakiraman, "Using Erasure Codes Efficiently for Storage in a Distributed System," in (DSN), 2005, pp. 336–345.

A. Beimel, "Secret-sharing schemes: A survey," in International Workshop on Coding and Cryptology (IWCC), 2011, pp. 11–46.