ISSN No: - 2456 – 2165

Secure Data Storage over Cloud Using Content Features Processing

Shaikh Sufiya Akhlaque Ahmed Shaikh, Prof.Asmita Deshmukh, Prof.Ankit Sanghvi, Prof. Ashwini Sagar

Abstract:-With the advent growth of sensitive information on cloud, Cloud security is getting more important than even before. Cloud computing is a type of computing that relies on sharing computing resources rather than having local server, or personal devices to handle applications. The cloud computing aim to providing IT as a service to the cloud user on demand basis with greater flexibility, availability, and reliability, with utility computing model. Millions of users, use a cloud computing for various purpose, therefore security, integrity, authentication privacy and confidentiality on the cloud is a major concern.

To address this issue, cryptographic algorithm are use to protect the private data over the cloud. The two type of algorithm is highly used in cryptography are symmetric and asymmetric for encrypting and decrypting the text .As if data used through internet is highly confidential which are not public viewing. so ,this work focuses on providing security to a data on cloud using elliptic curve cryptographic algorithm. This algorithm raises the confidentiality level to an exceptionally high standard using minimum resources as compared to other existing algorithms like RSA, DES, AES etc .

The major concern on cloud computing is to maintain the integrity of data stored on the cloud. Cloud servers have multiple users and owners data which can be attacked or modified by outside attackers. Therefore, a new concept is added which check the data Integrity on both the module data admin and data user. This entity called data verifier. It consists of four entities data admin, cloud server, data Verifier and data user. Data verifier perform main role of data integrity check at both the side of data owner as well as data user. It verifies the integrity of data on demand of the data owner while data owner want to verify its own uploaded data on cloud, so that it verify whether the data uploaded on cloud is tampered or not .where as data user can check integrity after downloading the text file which is share by data owners .This verifier scheme make use of SHA-1 Hash key algorithm for integrity check.

Indexterms:-Virtual Private Cloud, Elliptic Curve Cryptographic Algorithm, Shamir Secret Key, Data Security, Data Integrity, Data Verifier, Data Admin, Data User.

I. INTRODUCTION

Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. Cloud computing resources are available when needed, and you pay for their use in much the same way as for household utilities. In the same way, shared cloud resources can be used by others when not used by you .Cloud computing has five essential characteristics. They are on-demand self-service, broad network access, resource pooling, rapid elasticity, and measured service .One of the deployment model of a cloud are private cloud. The private cloud infrastructure is operated solely for an organization. It may be managed by the organization or a third party and may exist on-premise or off-premise. Using this virtual private cloud ,significant capital investment is required. The virtual cloud impacts on, file storage and sharing, cloud database, CRM, emails ,file backup, web hosting, Ecommerce etc. To protect the data of enterprise level and personal level information need to be secure from unsolicited access in the cloud for example, Information, documents , email ,pictures ,financial transaction, health records. The data first need to be verify through data verifier (SHA-1)hashing algorithm where data integrity check while data upload on cloud by data admin. It is data admin wish to check uploaded data integrity or just upload text file on a cloud and perform the next operation on it. After verification of a data integrity on a cloud that data is not get tampered ,data admin start performing encryption using asymmetric algorithm(ECC). To prevent the data from unauthorized accessing ,data user can decrypted the data from another key. Asymmetric algorithm used which generated encryption and decryption key. Keywords analysis indexing, ranking performed and data user can view a share files by data admin. To download a share files decryption key required. Data user can again check there downloaded data integrity by

ISSN No: - 2456 – 2165

performing data verifier (SHA-1) hashing algorithm. Data user can also search files on cloud using search keywords mechanism applied to retrieve more relevant files on cloud.

As shown in Figure 1, cloud storage can be used from smaller computing devices to desktop computers and servers.

Fig 1: Cloud Storage.

A. Existing Technology

In the existing project, work on semi trusted server or a paid sever cloud which is not most likely by the enterprise and personal level storage .The semi trusted server having risk to store a large amount of data and information .The database are not in the real time world which cannot give you the accurate result of your multi key words search. Because of not using the real time data set all the document and data which uploaded from data owner side to access by the data user was not guaranteeing a correct keywords and documentation.

The security of a uploaded data from data owner side using encryption key to till accessing and downloading the data from data user side decryption key both using a single key ,which is less secure and performance get decrease.

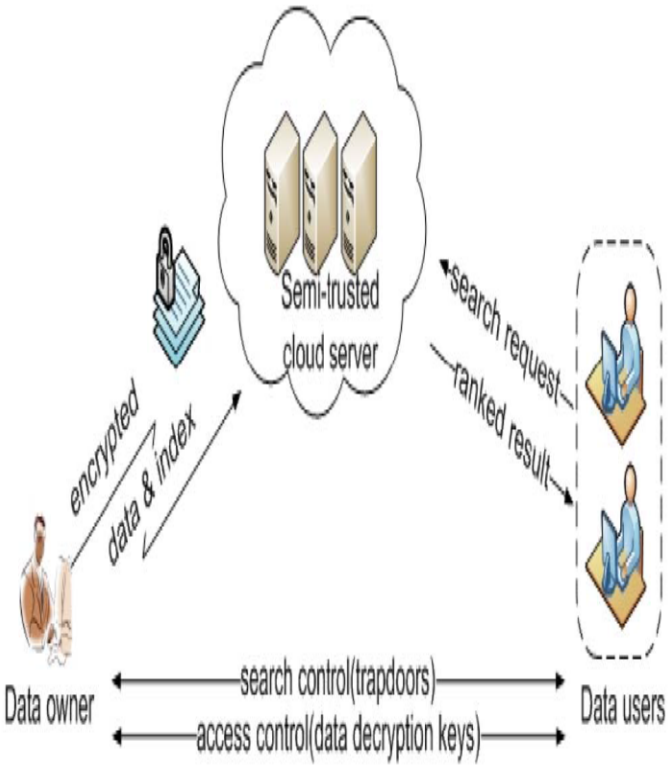

Fig 2: Architecture of the Search Over Encrypted Cloud Data.

B. ECC Algorithm

Elliptic Curve Cryptography (ECC) was discovered in 1985 by Victor Miller (IBM) and Neil Koblitz (University of Washington) as an alternative mechanism for implementing public-key cryptography.

Assume that those who are going through this article will have a basic understanding of cryptography (terms like encryption and decryption) .

The equation of an elliptic curve is given as,

$$y^2 = x^3 + ax + b$$

Few terms that will be used,
E -> Elliptic Curve
P -> Point on the curve
n -> Maximum limit (This should be a prime number)

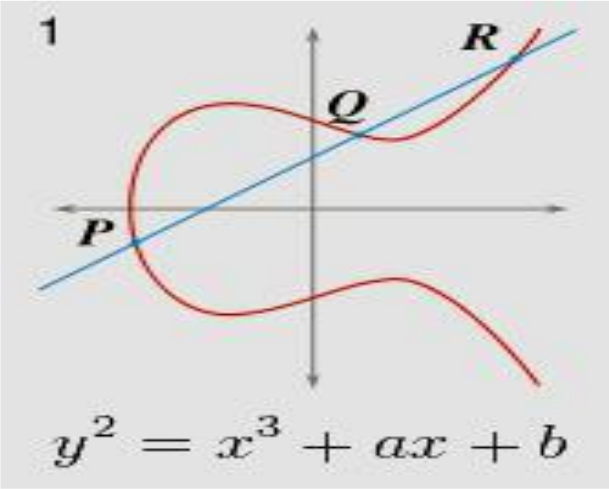

Fig 3: Simple Elliptic Curve

The Figure 3 shows simple elliptic curve.

Key Generation

Key generation is an important part where we have to generate both public key and private key. The sender will be encrypting the message with receiver's public key and the receiver will decrypt its private key.

Now, we have to select a number'd' within the range of **'n'**.

Using the following equation we can generate the public key
Q = d * P

d = The random number that we have selected within the range of (1 to n-1). P is the point on the curve.

'Q' is the public key and 'd' is the private key.

Encryption

Let 'm' be the message that we are sending. We have to represent this message on the curve. This have in-depth implementation details. All the advance research on ECC is done by a company called certicom.

Consider 'm' has the point 'M' on the curve 'E'. Randomly select 'k' from [1 – (n-1)].

Two cipher texts will be generated let it be C1 and C2.
C1 = k*P

C2 = M + k*Q

C1 and C2 will be send.

Decryption

We have to get back the message 'm' that was send to us,

M = C2 – d * C1

M is the original message that we have send.

Proof

How does we get back the message,

M = C2 – d * C1

'M' can be represented as 'C2 – d * C1'
C2 – d * C1 = (M + k * Q) – d * (k * P) (C2 = M + k * Q and C1 = k * P)

= M + k * d * P – d * k *P (canceling out k * d * P)

= M (Original Message)

ECC(in bits)	RSA(in bits)
106	512
112	768
132	1024
160	2048
210	3072
283	7680
409	15360
571	21000

Table 1: Key Size of ECC and RSA

Advantages of ECC over RSA

- Low on CPU consumption.
- Low on memory usage.
- Good protocols for authenticated key exchange.
- Moderately fast encryption and decryption.
- Smaller keys, cipher texts and signatures.
- Very fast key generation.

C. Data (SHA-1) Hash Value Algorithm

SHA-1(secure hash algorithm) is a revised version of SHA design by the national instititute of standard and technology(NIST) .The secure hash algorithm is a family of cryptographic hash function published by the BIST and FIPS .It includes the following variations.
SHA,SHA 0, SHA 1,SHA-2.Each creates a digest of length N from a multiple block message. each block is 512 bit in length.

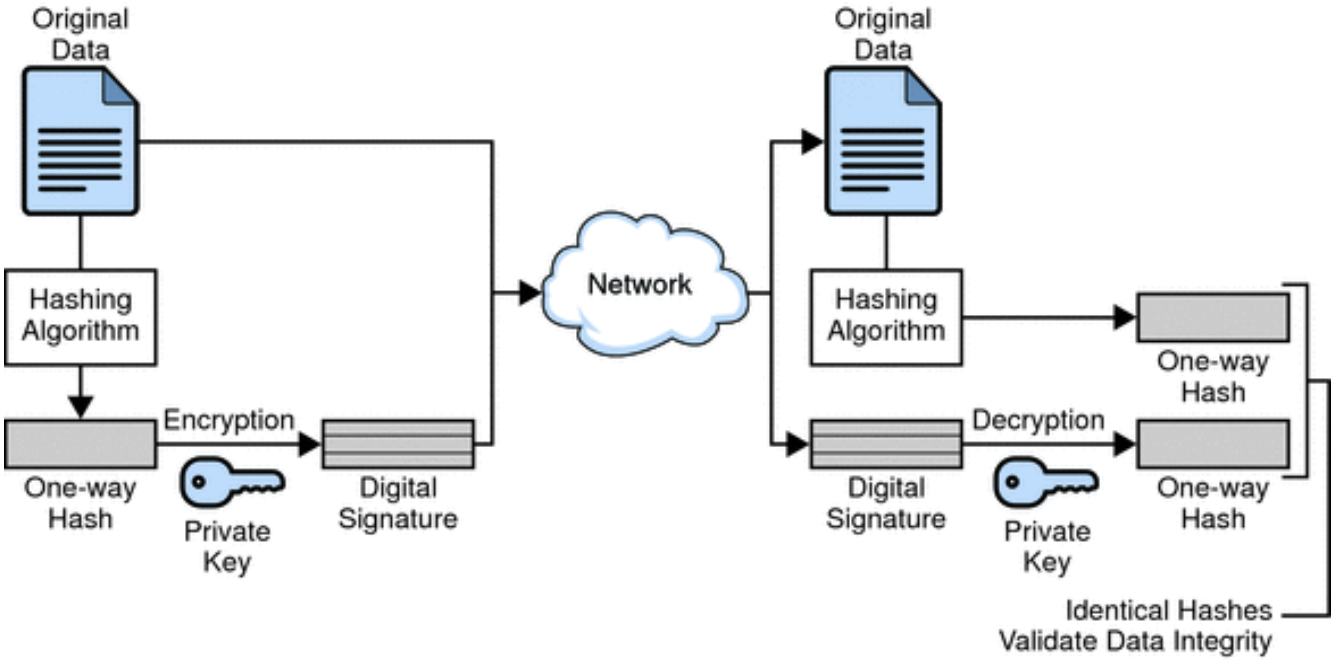

Fig 4: SHA-1 Hash Value

II. LITERATURE REVIEW

A. *ECC with Data Integrity*

The cloud architecture proposed in this paper brings suitable way to store and access files provided with confidentiality, integrity and authentication properties. Data is encrypted before uploading to server storage, so confidentiality is preserved. On the receiver side the user can download and decrypt the files using the key stored in mail server at the time of encryption providing authenticity. Further the third party auditor reliefs user from the overhead of ensuring integrity of data by periodically verifying it. The proposed model proves that it is secure in terms of integrity and confidentiality through security analysis. Through, performance analysis and results proved that proposed scheme is efficient. Compared with previously proposed protocols, we have also proved that proposed scheme is more secure and efficient.

B. *Searching*

Search over encrypted data is a technique of great interest in the cloud computing era, because many believe that sensitive data has to be encrypted before outsourcing to the cloud servers in order to ensure user data privacy. In the literature many privacy preserving multi-keyword search requirements are defined in the cloud. The privacy preserving requirements are as follows:

C. *Survey of Searchable Encryption*

In the literature, searchable encryption is a helpful technique that treats encrypted data as documents and allows a user to securely search through a single keyword and retrieve documents of interest. However, direct application of these approaches to the secure large scale cloud data utilization system would not be necessarily suitable, as they are developed as crypto primitives and cannot accommodate such high service-level requirements like system usability, user searching experience, and easy information discovery. Although some recent designs have been proposed to support Boolean keyword search as an attempt to enrich the search flexibility, they are still not adequate to provide users with acceptable result ranking functionality .Our early works have been aware of this problem, and provide solutions to the secure ranked search over encrypted data problem but only for queries consisting of a single keyword. How to design an efficient encrypted data search mechanism that supports multi-keyword semantics without privacy breaches still remains a challenging open problem.

Sr.No	Title	Author	Year	Work Done
1	Privacy-Preserving Multi-Keyword Ranked Search over Encrypted Cloud Data	Ning Cao,Cong Wang	January 2014-IEEE Transaction	Algorithm:-Index privacy, keyword privacy, Access pattern. Security:-Cryptography,Single searchable Encryption. Database:-Static data set
2	Enabling Secure and Efficient Ranked Keyword Search over Outsourced Cloud Data	Dan Boneh,Giovanni Di Crescenzo	August 2011-IEEE Transaction	Algorithm:-Searchable Encryption ,Order preserving mapping ,Cloud Computing. Security:-Confidential Data,Symmetric Key. Data base:-One to many order preserving mapping.
3.	Secure Ranked Keyword Search over Encrypted Cloud Data	Ayad Ibrahim	9-11 Sept. 2010 International Conference	Algorithm:-Inverted Index,Rank Function. Security:-Boolean Keywords search,SSE (Symmetric search encryption).
4.	Toward Secure and Dependable Storage Services in Cloud Computing	Qian Wang Cong Wang,	April-June 2012 IEEE Transaction	Algorithm:-Data integrity, dependable distributed storage. Security:-error localization. Data base:-Dynamic data
5.	Similarity Search in High Dimensions via Hashing	Rajeev Motwaniz	Sept 2013 IEEE paper	Algorithm:- Nearest Neighbor Search(NNS),Locality sensitive hashing(LHS). Security:-Symmetric key. Data base:-Dynamic data set.
6.	Enhanced data security model for cloud using ECC algorithm and third party auditor	Niyati Jain1, Priya Jain2, Nikita Kapil3	*March 2016 (IJARCET)*	Algorithm:- Data confidentiality and integrity Security: Third Party Auditor (TPA). Elliptic curve cryptography algorithm *(ECC*), Cloud Service Provider (CSP)

Table 2: Literature Survey

III. PROJECT WORK

Our project work provide effective security and integrity by using Shamir's secret hashing algorithm and Elliptic Curve Cryptography (ECC) algorithm. It also provide data verifier scheme which assure data integrity not only at data user side but also assure data admin too about confidentiality and data security on a cloud server.

Data admin upload their data on a cloud and generate a hash value on original text file using SHA-1 algorithm ,admin can download the text file from a cloud and applied hashing algorithm on it again to check the data integrity. ,If hash value get match means no data get tampered confidentiality is maintain on cloud. server. Else if data admin want can directly upload text file on cloud without integrity checking. Data Verifier scheme used at data admin level which provide the data security on a cloud as well as assure admin about its uploaded data not get manipulated by attacker.

After uploading the text file on a cloud server, Elliptic Curve Cryptography(ECC) algorithm used to encrypt the text file using secrete key encryption. The key mail to a respective user whom data admin wants to share text file.

Data user can view a share files by data admin, and can download a share file too through secrete decryption key present on a data user given mail .Data user can again check the integrity of original download text file content by using SHA-1 hash value algorithm. This scheme is know as Data verifier which match the original uploaded text file hash value with downloaded text files hash value .Both hash value get match means no data get manipulated.

Data user can also search a files on a cloud server using a search term keyword which gives relevant files list to a data user which are not shared by data admin.

ISSN No: - 2456 – 2165

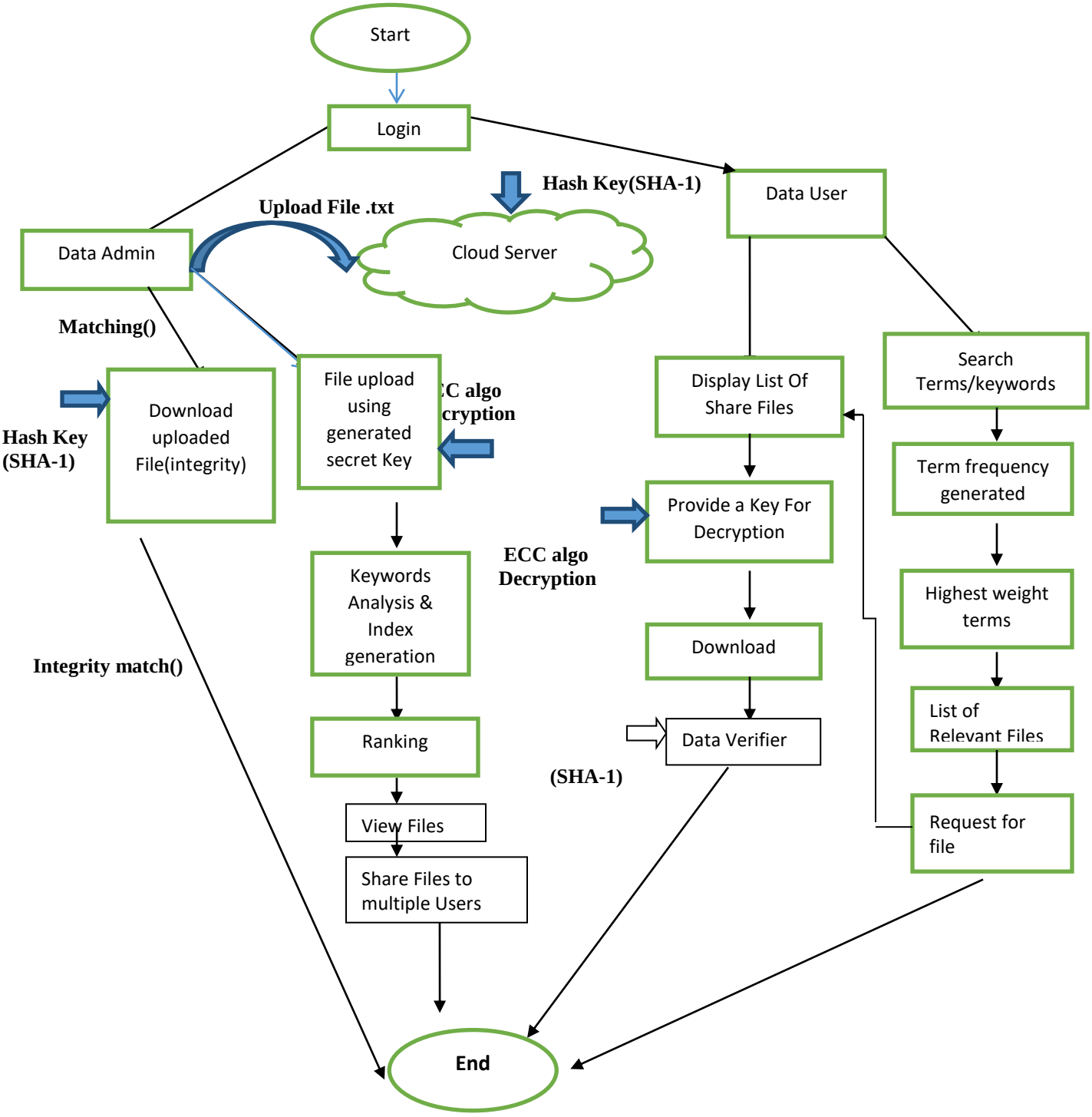

Fig 5: Project Work Flow

A. Data Admin Module

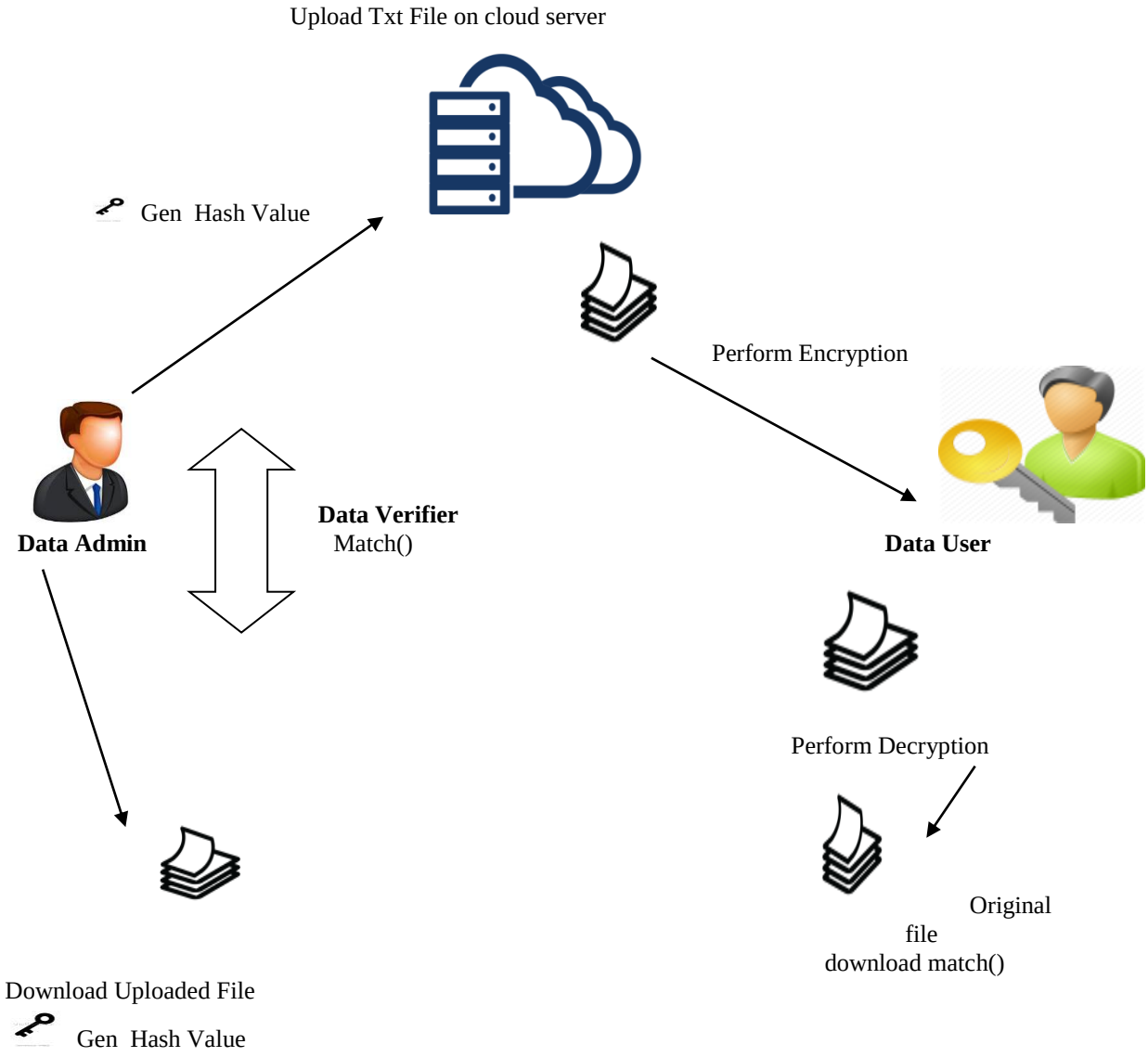

Fig 6: Data Admin and Data User Module

Data admin contain the entire process of admin ,it is a initial module from where work begin. Data admin upload the text file on a cloud server and check the uploaded file integrity by using SHA-1 hash value algorithm under data verifier module. After uploading a file on cloud server Elliptic Curve Cryptography (ECC) cryptographic algorithm perform. Which generate a encryption key .File uploading contain few mechanism are keyword analysis, indexing ,ranking etc .Now ,data admin can select the data users and share the encrypted file to them. Data user can view a files and download them by using decryption key. To perform decryption data user need a key which is present on mail. After downloading a original file data user can check the integrity of a text file by generating hash value SHA-1 on it.

a). Data Verifier

- *Dealing with Integrity at Data Admin level*

Data verifier is a module which concern about a data integrity at data admin and data user level. Major concern of data verifier to perform integrity checking while data admin upload text file on a cloud server .It generate hash value using SHA-1 algorithm. Data admin can download a uploaded text file a again generate hash value. Both the hash value should be match .Matching of hash value through SHA-1 algorithm

confirm integrity. Data on a cloud server not get manipulated by attacker or hacker.

b). File Uploading

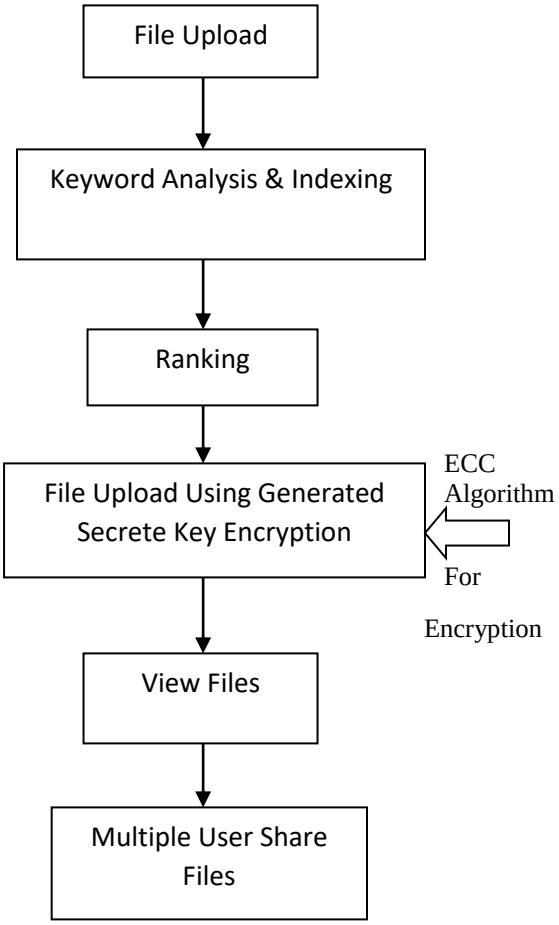

Fig 7: File Uploading

The file uploading is initial step on cloud server. Following are the steps how text file upload on a cloud server.

1.Retrieved File contents.

2.Tokenization(where sentences are break into words).

3.Stopword removal.

4.frequency count of a keywords

5.Store keywords into a data base.

Index Construction

A character-based or byte based searching method uses a character or characters (i.e., n-gram) as an indexing key unit. It stores each key (a character) with a list of all positions where it appears in a document collection

For each cluster , constructs an index.
1.Tokenization
2 stop word removal
3.Retrieves high frequency keywords in all files
5.concatenates all the keywords

Ranking

The ranking performed after the keyword get search from uploaded document by the data owner. Example fruits, tree, plant etc, keywords from different text files. while this keywords generated by the searching mechanism .

- Search files which contains keywords entered for search.
- From above step we will get list of files containing keywords.
- In search result files will be shown according to number of occurrences of keywords in selected files.

The following we have example ,how the ranking algorithm work in the project:
Eg:- apple, fruit, tree ,plant-(These are the search keywords)
Frequency Count:- apple-10

Fruit-8
Tree-11
Plant-4

Doc2 contain apple and fruit->10+8=18
doc1 → fruit, tree → 8 + 11 = 19
doc3 → plant → 4.

In the above mechanism of ranking, we have 3 documents (doc1, doc2, doc3) contains a search keywords. Each keywords having tag, of no. of time it appearances in the one document.(apple-10,fruits-8,tree-11,plant-4).If keywords repeated in more than one document .Then after as per frequency count the file will be ranked on a user file list.

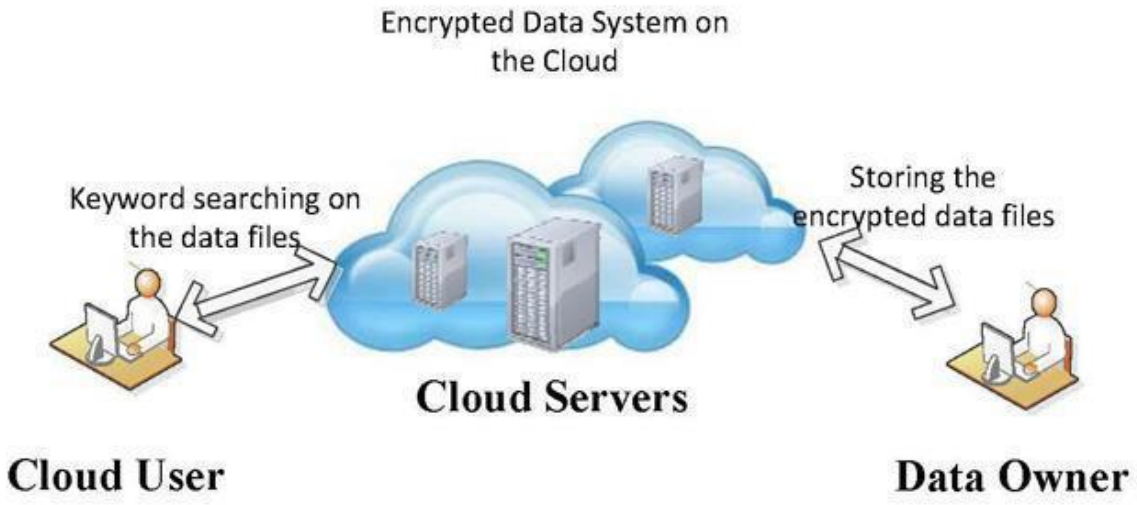

Fig 8: Encryption on a Cloud Server

Encryption

Encryption is the process of taking information in one form (usually human-readable), and converting it to another form (not usually human-readable). To perform encryption we used Elliptic Curve Cryptography (ECC) algorithm. To encrypt, the public key is applied to the target information . ECC encryption perform on a uploaded data (text file). Encrypt the data by using public key enhance data security on a cloud so that data will be move to Data user with full of confidentiality. The small key sizes make ECC very appealing for devices with limited storage or processing power, which are becoming increasingly common on the cloud services. The smaller key sizes can offer speedier and stronger security.

c). View Files

If data owner wants to see any specific file, this module provides the option. The output of search operation will provide the list of the files; the module will also have the option for deletion of specific file.

d). File Share

This module provides the option for file sharing. The module is somehow depends on the View file module. After searching files, the data owner can select specific file(s) and select specific user(s) with whom he want to share with.

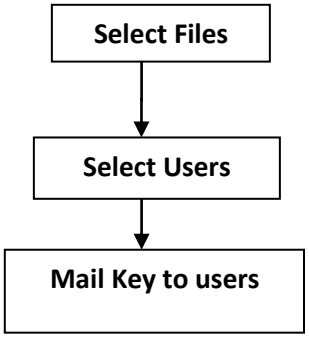

Fig 9: File Sharing Flow

B. Data User Module

Data user module is a second module of a project .where user logion and view a share files by data admin .Data user can select a file and decrypt the share file by using a secrete private key .The key mail to a data user, which is generated through a ECC algorithm.

After download a original file, data user can perform data verifier scheme. This scheme generate a hash value(SHA-1).The generated hash value ,now match with early generated(data uploading)hash value. If both the hash value get match integrity of a data is maintain.

ISSN No: - 2456 – 2165

The data user can also search a file on a cloud server by keywords analysis. To search a file on a cloud server following process will be perform.

1.Search a file
2.Map the keywords.
3.Identify related files with frequent count.
4.Ranking based on frequent count.
5.List of files arrived.

The data user can request for a files to a data admin and again files will be shared by data admin to a requested user .The data user can download a shared file.
The data user cannot do multiple request for a single file.

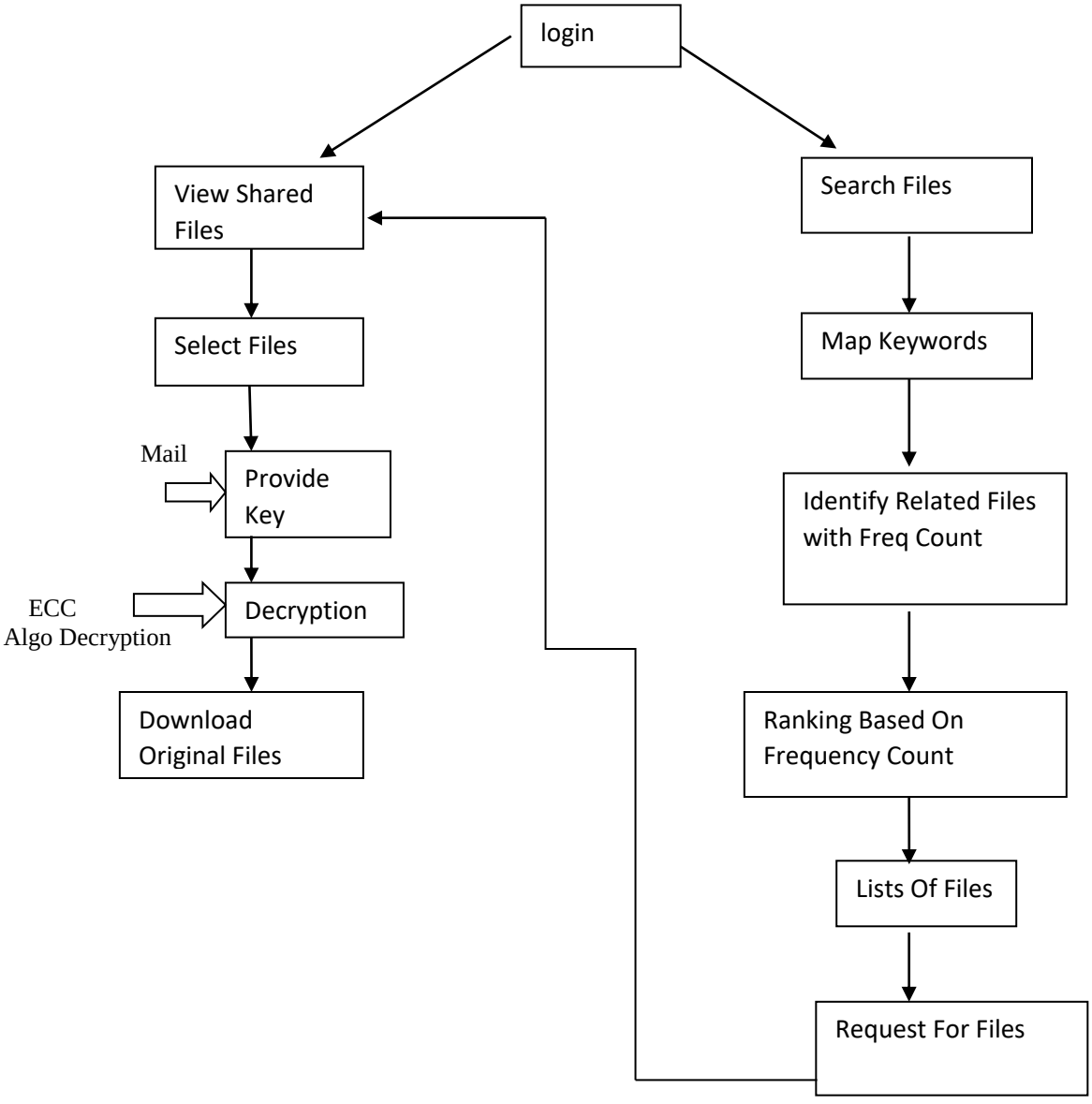

Fig 10: Data User Module

ISSN No: - 2456 – 2165

a). Data Verifier

- *Dealing with Integrity at Data User level:*

The encrypted text file using ECC algorithm get decrypted by secrete key. The data user download a text file. For checking integrity of a data user can generate a hash value using SHA-1.That hash value should be match with uploaded text file hash value .If both the key value get match. Means no original data get tampered.

The data verifier module contains integrity aspect of a data. That data will not get manipulated on cloud server as well as data user side.

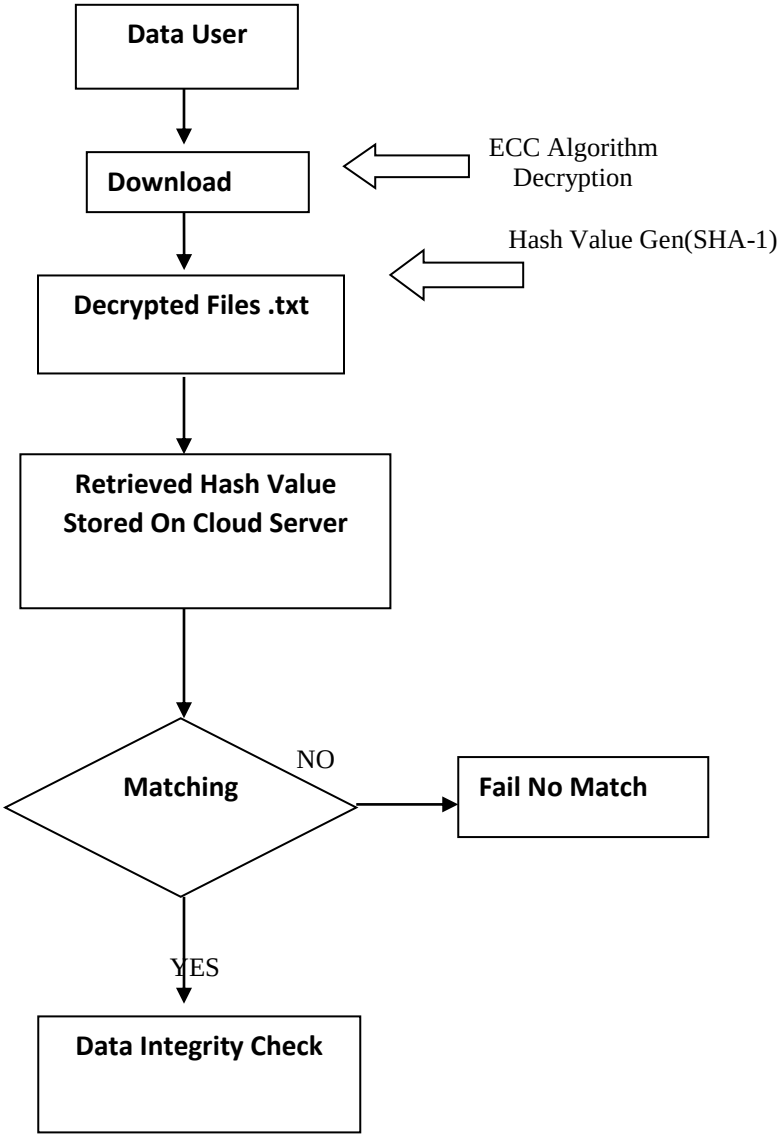

Fig 11: Data Verifier

ISSN No: - 2456 – 2165

IV. SYSTEM ARCHITECTURE

A. *System Model*

Cloud Data Storage Model The cloud storage model considering here is consists of three main components .

B. *Data Admin Login panel*

This module login into a system using username and password .If username and password match then admin get access to the application.

a). Manage User Module

Data admin add a new user over here ,with its full name, contact no ,Email id. only admin have a authority to add, edit or delete data user details.

b). Manage Files Module

Admin upload a text file with file _id and file _name field. After choosing a text file ,admin click
on encrypt option.

c). Share File Module

Data admin can share a encrypted text file to a data user. This module have Share _id, file_ name, user_ email fields.

C. *Data User Login Panel*

This module user login into a system using username and password .If username and password match then sender et access to the application.

a). View file Module

Data user can view a share files by data admin .This module have following fields are file_ id, file_ name, Download.

b). Request File Module

In this module data user can search a keyword, and relevant keyword file in a cloud will be arrived. the following fields are used file_ id, file_ name and request.

V. ANALYSIS RESULT

Short key size: ECC employs a relatively short encryption key, a value that must be fed into the encryption algorithm to decode an encrypted message. This short key is faster and requires less computing power than other algorithms. For example, a 160-bit ECC encryption key provides the same security as a 1024-bit RSA encryption key and can be up to 15 times faster, depending on the platform on which it is implemented fig 5 .Comparison of key generation time(ms) .

- More Complex: In spite of multiplication or exponentiation in finite field, ECC uses scalar multiplication. Solving Q= dP (utilized by ECC) is more difficult than solving factorization (used by RSA) and discrete logarithm (used by Diffie-Hellman (DH), EIGamal , Digital Signature Algorithm (DSA)).
- Power Consumption: ECC requires less power for its functioning so it is more suitable for low power applications such as handheld and mobile devices
- Computational Efficiency: Implementing scalar multiplication in software and hardware is much more feasible than performing multiplications or exponentiations in them. As ECC makes use of scalar multiplications so it is much more computationally efficient than RSA and Diffie-Hellman (DH) public schemes. So we can say without any doubt that ECC is the stronger and the faster (efficient) amongst the present techniques.

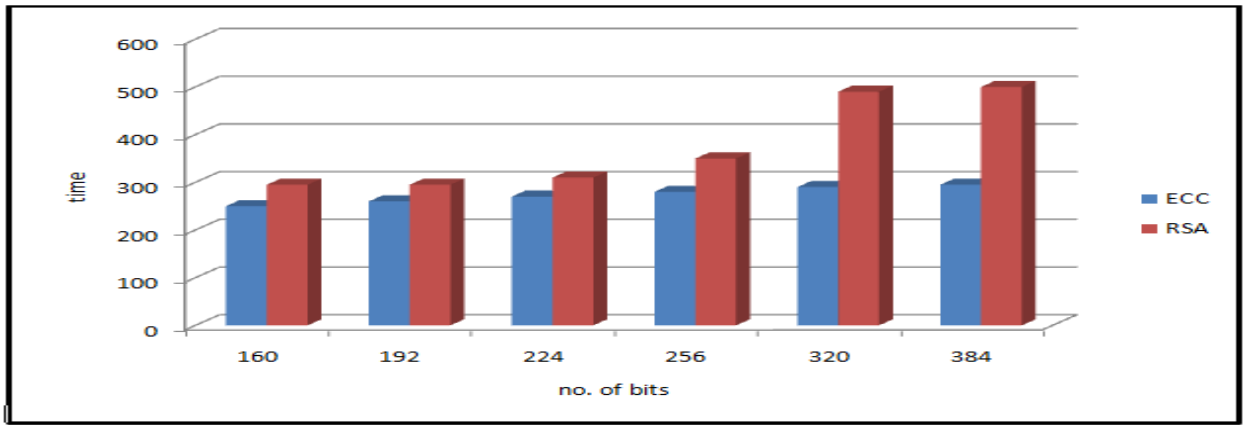

Fig 12: Comparison of Key Generation Time (Ms)

VI. CONCLUSION

The secure data storage over a cloud in this paper brings suitable way to store and access files provided with confidentiality, integrity and authentication properties. On the data admin side, data integrity checking perform before uploading on a cloud server .Then, cloud server have encrypted data, so confidentiality is preserved .Data user on the another have perform decryption to download the encrypted file into original format using the key stored in mail server at the time of encryption providing authenticity. further data verifier relief the data user by checking original data integrity. The project system model proves that it is secure in terms of integrity and confidentiality through security analysis. Through, performance analysis and results proved that project scheme is efficient. Compared with previously proposed protocols, we have also proved that project scheme is more secure and efficient.

REFERENCES

[1]. Ning Cao, Cong Wang, Ming Li, KuiRen, and Wenjing Lou, "Privacy-Preserving Multi-keyword Ranked Search over Encrypted Cloud Data" IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS VOL:25 NO:1 YEAR 2014

[2]. D. X. Song, D.Wagner, and A. Perrig. Practical techniques for searches on encrypted data. In Proceedings of the 2000 IEEE Symposium on Security and Privacy, Berkeley, CA, USA, May 2000.

[3]. Niyati Jain1, Priya Jain2, Nikita Kapil "Enhanced data security model for cloud using ECC algorithm and third party auditor " International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 5 Issue 3, March 2016

[4]. Gopinath V 1, Bhuvaneswaran R.S "Study on Secure Cloud Computing with Elliptic Curve.

[5]. Cryptography"IJCSI International Journal of Computer Science Issues, Vol. 11, Issue 5, No 2, September 2014.

[6]. T. Moataz and A. Shikfa. Boolean symmetric searchable encryption. In Proceedings of the 8th ACM SIGSAC Symposium on Information, Computer and Communications Security, Hangzhou, China, May 2013.

[7]. C. Orencik, M. Kantarcioglu, and E. Savas. A practical and secure multi-keyword search method over encrypted cloud data. In Proceedings of the 6th IEE International Conference on Cloud Computing, Santa Clara, CA, USA, June 2013.

[8]. D. Boneh and B. Waters. Conjunctive, subset, and range queries on encrypted data. In Proceedings of the 4th IACR Theory of Cryptography Conference, Amsterdam, The Netherlands, Feb. 2007.

[9]. N. Cao, C. Wang, M. Li, K. Ren, and W. Lou. Privacy-preserving multi-keyword ranked search over encrypted cloud data. In Proceedings of the 30th IEEE International Conference on Computer Communications, Shanghai, China, Apr. 2011.

[10]. C. Wang, N. Cao, K. Ren, and W. Lou. Enabling secure and efficient ranked keyword search over outsourced cloud data. IEEE Transactions on Parallel and Systems, 23(8):1467–1479, 2012.

[11]. D. Boneh, E. Kushilevitz, R. Ostrovsky, and W.E.S. III, "Public KeyEncryption That Allows PIR Queries," Proc. 27th Ann. Int'lCryptology Conf. Advances in Cryptology (CRYPTO '07), 2007.

[12]. P. Golle, J. Staddon, and B. Waters, "Secure Conjunctive Keyword Search over Encrypted Data," Proc. Applied Cryptography and Network Security, pp. 31-45, 2004.

[13]. L. Ballard, S. Kamara, and F. Monrose, "Achieving EfficientConjunctive Keyword Searches over Encrypted Data, Seventh Int'l Conf. Information and Comm. Security (ICICS '05),2005.

[14]. D. Boneh and B. Waters, "Conjunctive, Subset, and Range Querieson Encrypted Data," Proc. Fourth Conf. Theory Cryptography (TCC),pp. 535-554, 2007.

[15]. R. Brinkman, "Searching in Encrypted Data," PhD thesis, Univ. ofTwente, 2007.

[16]. Y. Hwang and P. Lee, "Public Key Encryption with ConjunctiveKeywordSearch and Its Extension to a Multi-User System,"Pairing, vol. 4575, pp. 2-22, 2007

[17]. J. Katz, A. Sahai, and B. Waters, "Predicate Encryption SupportingDisjunctions, Polynomial Equations, and Inner Products," Proc.27th Ann. Int'l Conf. Theory and Applications of CryptographicTechniques (EUROCRYPT), 2008.

[18]. A. Lewko, T. Okamoto, A. Sahai, K. Takashima, and B. Waters,"Fully Secure Functional Encryption: Attribute-Based Encryptionand (Hierarchical) Inner Product Encryption," Proc. 29th Ann.Int'l Conf. Theory and Applications of Cryptographic Techniques(EUROCRYPT '10), 2010.

[19]. E. Shen, E. Shi, and B. Waters, "Predicate Privacy in EncryptionSystems," Proc. Sixth Theory of Cryptography Conf. Theory ofCryptography (TCC), 2009.

[20]. M. Li, S. Yu, N. Cao, and W. Lou, "Authorized Private KeywordSearch over Encrypted Data in Cloud Computing," Proc. 31st Int'l Conf. Distributed Computing Systems (ICDCS '10), pp. 383-392, June 2011.

[21]. C. Wang, N. Cao, J. Li, K. Ren, and W. Lou, "Secure RankedKeyword Search over Encrypted Cloud Data," Proc. IEEE 30th Int'lConf. Distributed Computing Systems (ICDCS '10), 2010.

[22]. C. Wang, N. Cao, K. Ren, and W. Lou, "Enabling Secure andEfficient Ranked Keyword Search over Outsourced Cloud Data,"IEEE Trans. Parallel and Distributed Systems, vol. 23, no. 8, pp. 1467-1479, Aug. 2012.