Security and Privacy Preserving Keyword Search for Cipher Text Retrieval in Cloud Computing based on Oppositional Grasshopper optimization

Kasiviswanadham Y[1]

[1]Research Scholar, Department of CSE, S.V.U. College of Engineering, Tirupati, A.P, India
Email: ykvnath@gmail.com

Dr.Ch.D.V. Subbarao[2]

[2]Professor of CSE, Department of CSE, S.V.U. College of Engineering, Tirupati, A.P, India
Email: subbarao_chdv@hotmail.com

Abstract

Because more and more people and businesses are storing data in the cloud on a remote basis, cloud security is an essential research topic. Improving the data security and privacy our proposed work uses combination of Blowfish algorithm as well as Elliptic curve cryptography (Blowfish+ECC). This study mainly focusses on two stages; public key encryption as well as keyword retrieval. Initially, ECC provide access control for both data owner and authorized users. Porter stemming is used to extract keywords in the data pre-processing stage. The data owner uses the bloom filter to construct a secure index for each file. The data owner then encrypts the files using the blowfish encryption method and public keys. In the next stage, the trapdoor code is generated. After the user receives the information, the keyword search approach is utilized to discover matched files. The Oppositional based Grasshopper optimization Algorithm is used to select the optimal solution after gathering matched files. Finally, using the blowfish decryption method, the original files are decrypted.

Keywords: **Blow fish, Elliptic curve cryptography, Bloom filter, Trapdoor, Oppositional Grasshopper optimization.**

1. Introduction

The latest progress of distributed computing has displayed its capacity to change the advancing system a striking IT gear is envisioned and gotten [1]. In conveyed processing the enrolling organizations can be requested into Framework as-a-Administration (IaaS), Stage as-a-Administration (PaaS) and Programming as-a-Administration (SaaS) [2]. Capacity as-an administration has risen as a business elective for nearby information stockpiling because of its qualities incorporate less beginning foundation arrangement, alleviation from upkeep overhead and aggregate passage to the information independent of area and gadget [3].Cloud stockpiling, which is one of different cloud administrations, fills in as a down to earth apparatus and has made information redistributing to the cloud a developing pattern [4]. Distributed storage is a basic advantage of distributed computing, which has been logically basic since it can give negligible exertion and on-request use fitting to expansive capacity and handling assets [5]. As of late, distributed computing has risen as another stage for conveying, overseeing, and giving huge scope information administrations through an Internet-based framework. Fruitful models incorporate Amazon EC2, Google App Engine, and Microsoft Azure. By re-appropriating information and administrations, cloud clients appreciate a versatile excellent assistance in a financial and productive way since they can powerfully build their extra room as and when required without purchasing any capacity gadgets [6]. They are: (1) the clients can get to the distantly put away information whenever, from anyplace and allows approving clients and to get shared info. (2) On a local level, clients can be safeguarded from the board's weight of capacity, (3) capital expenditures on equipment and programming can be avoided, and so on. [7]. At the point when access is fueled by distributed computing, clients can utilize complete arrangements of instruments for evaluating different applications, stockpiling and stages through the Internet, just as utilizing the administrations offered by cloud makers [8].

There are two principle assaults for information put away in cloud under such a condition, i.e., outer assaults started by unapproved pariahs and inward assaults started by conniving CSPs. Sometimes, we can't

completely confide in a CSP, yet at the same time need its administrations. It is, in this way, essential to give adequate safety efforts to shielding the put away information both from malevolent untouchable assaults and the specialist organization itself, and this absence of trust is critical as it brings new security issues towards the cloud condition [9]. Subsequently, a few systems are expected to ensure the client information security and the client questioning protection in a cloud domain [10]. Thus information proprietors request elevated levels of security and classification when they re-appropriate their information to a cloud; in spite of the fact that they as a rule scramble their information while putting away it in a cloud worker, they despite everything need power over it, for instance, in the event that they much of the time update it. Direct work of conventional cryptographic natives can't accomplish the information security required [11]. In spite of the specialized and monetary favorable circumstances of distributed computing, numerous potential cloud buyers are as yet reluctant to receive distributed computing because of apprehensions about one's safety and security Because the vast majority of security measures and processes used by "cloud service providers (CSPs)" are not fully understood by "cloud service users (CSUs)," these security and protection challenges occur [12].Throughout the years diverse encoding approaches have been created and utilized viably for the security insurance of such touchy datasets. In any case, these methodologies ended up being unreasonable, costly and wasteful. Really, the assurance of the cloud datasets by means of encryption is exceptionally troublesome and testing in light of the fact that the majority of the current applications depend on decoded datasets [13].

These days there are many propelled encryption techniques for large information protection saving plans and are recorded as follows, Attribute-based encryption (ABE) is imagined as an exceptionally encouraging open key crude for acknowledging adaptable and fine-grained get to control frameworks, where differential yet adaptable access rights can be allocated to singular clients. Particularly, cipher text-policy attribute-based encryption (CP-ABE) empowers information proprietors to indicate an entrance strategy over a vast expanse of characteristics and encode the information under the entrance strategy with the relating open key segments. Decoding is empowered if and just if the client's qualities coordinate the relating access strategy [14]. At that point in the distributed computing and huge information situations, Order-preserving encryption (OPE) will be more valuable, in light of the fact that redistributed database has pulled in much consideration as of late because of the rise of distributed computing, be that as it may, how to ensure the re-appropriated information putting away in the untrusted cloud worker turns into a difficult issue. Since request safeguarding, OPE permits untrusted worker to perform database tasks, for example, correlation and range question over encoded information, without unscrambling them [15]. Notwithstanding of the propelled strategies created it is trying to plan and acknowledge strong protection safeguarding frameworks. This test starts from the fighting main impetuses that framework architects need to all the while consider [16]. Also anyway applying these customary ways to deal with huge information anonymization presents adaptability and effectiveness challenges as a result of the "3Vs", i.e., Volume, Velocity and Variety [17]. Most importantly, it is as yet a test to effectively accomplish protection conservation over appropriated and steady information within the sight of information refreshes [18].

1.1 Contribution

The contribution of this work is described as follows,

- Enhance the data security as well as privacy our proposed work using mixture of Blowfish algorithm and Elliptic curve cryptography.
- Public key encryption as well as keyword retrieval is the two main stages used in this research.
- Elliptic Curve Cryptography provides access control for both data owner and authorized users.
- Porter stemming is used to extract keywords in the stage of data pre-processing.
- Oppositional based Grasshopper enhancement Calculation is utilized to track down the ideal arrangement among the coordinated with documents.
- In terms of cost and energy time, our results show that our proposed technique outperforms existing efforts.

The rest of this paper is structured as follows. Section 2 looks at some earlier literature that is pertinent to our investigation. Section 3 proposes an alternative Grasshopper optimization-based security and privacy-preserving keyword-based search for cypher text retrieval in cloud computing. Section 4 delves into the findings of our proposed approach's experiments. Section 5 brings this research to a close.

2. Literature Survey

Yong Yu et al. [19] There has been an advancement in (I-D) Remote Data Integrity Checking (RDIC) convention due to the use of key-homomorphic cryptographic fundamentals to reduce the complexity of the framework and the cost of showing and keeping the open key tribute structure. An outsider verifier is confronted with a zero-accomplishment enigma as ID-based RDIC and its security model become official.... RDIC procedures using ID-

based RDIC protocols do not divulge any information about spared information to the verifier. The novel construct is proven in the classic gathering model to be secure against hostile employees and to acquire zero information mystery near a verifier. The outcomes of a thorough security examination and permission are revealed.

Xuefeng Liu et al. [20] has introduced a novel message-locked integrity auditing technique that is applicable to both record level and bump level duplication systems and does not require the involvement of a third-party mediator. This particular design is capable of removing the superfluous code text. It offers trustworthiness label deduplication through a message-determined marking key, resulting in a low customer-side computation cost. By combining the intermediary re-signature approach, you can complete the integrity investigation across any customer's distributed storage without reservation. In the discretionary prophet model, there is a "Computational Diffie-Hellman" uncertainty that proves that the innovative technique will not leak the information possession data. Last but not least, we have the execution appraisal.

Yue Zhang et al. [21] Storage method that utilizes cloud-based storage without being bound by the cloud-based record squares of a customer's data. To achieve this purpose, a unique key age technique and a new private key update mechanism can both be used in conjunction. In contrast to the authenticators of the revoked client, only the non-repudiated total client's private keys can be revived. If the authenticators aren't turned back on, it's possible to examine the veracity of a customer's data that has been rejected. It relies on personal data cryptography, which reduces the complex verification process that is typical of PKI frameworks.

Raman Kumar et al. [22] Job basis access control, encryption, and mark check for securing documents are all part of this security architecture's three-tiered security system design. To this end, a more robust and secure dynamic examination convention is presented that can accurately store information in the cloud. An independent auditor (TPA) and the data combiner can verify that their data is of the highest quality. As a result, the advanced secure dynamic evaluation standard that has been designed is safe and effective against a variety of connivance threats.

Yang et al. [23] passed on a successful public auditing system that may protect individual privacy while also allowing a group of persons to be identified. They initially envisioned a new framework for cloud-based information sharing and defined the meaning of the open evaluation plan for shared cloud data that supports the protection and detection of people's identities. And afterward they developed such a plan, in which a gathering administrator was acquainted with assistance part's created authenticators to ensure the character protection and two records were utilized to record the individuals who played out the most recent adjustment on each square to accomplish the personality discernibility.

3. Proposed Methodology

In cloud computing environments, the most significant challenge that cloud providers and their clients face is security. Data security, data integrity, and data storage are critical in this case. On the other side, data misbehavior is becoming a rising problem for data owners and cloud service providers. To solve the issue of cloud data security, we created a system that combines the Blowfish algorithm and Elliptic curve cryptography (Blowfish+ECC). This study mainly focusses on two stages; public key encryption as well as keyword retrieval. Initially, ECC provide access control for both data owner and authorized users. Porter stemming is used to extract keywords in the data pre-processing stage. Data owner creates a secure index for each file using bloom filter. After that data owner encrypt the files based on blowfish encryption algorithm with the help of public keys. In the next phase, trapdoor code is generated. After receiving trapdoor from the user matching files are searched using keyword search technique. After retrieving matched files Oppositional Grasshopper optimization Algorithm is used to find the optimal solution. Finally decrypts the original files based on blowfish decryption algorithm.

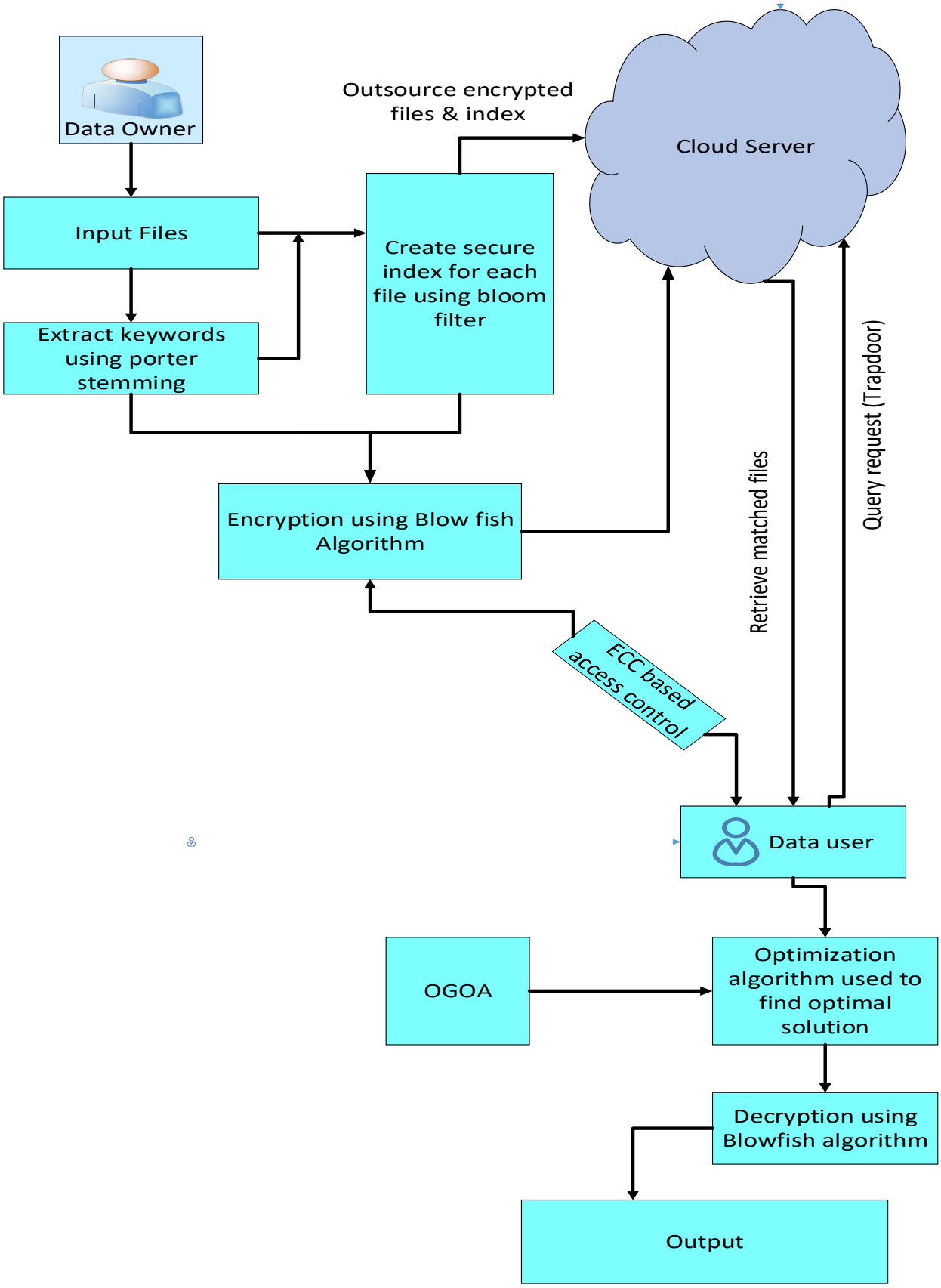

Figure 1: Overall architecture of proposed methodology

As shown in Figure 1; the data owner has some input files from those files, the key words are extracted based on the Porter stemming process. After completing the extraction process create secure index for each files using bloom filter. The detailed explanation of this is discussed as follows;

3.1. Pre-processing:

Data owner having large amount of input files $IF=\{f_1, f_2, f_3,..f_n\}$ to outsource the input files on the cloud in cipher text form for that purpose, first creates searchable index SI from a specific keyword set $KW=\{w_1, w_2, w_3,..w_m\}$ which is extracted from the input file collection of IF. Here porter stemming is used to extract the keywords from input file (IF).

3.2. Porter Stemming Process:

For example, troubled and troubles are stemmed by lowering their inflection to their root form (e.g. trouble). There is a possibility that the "root" in this example is not an actual root word, but rather a canonical form of the original term. Following is an illustration of Porter's stemming method.

Original-word and Stemmed-words

Original-word	Stemmed-words
“Connect”	“Connect”
Connected	“Connect”
Connection	“Connect”
Connections	“Connect”
Connects	“Connect”

Original-word and Stemmed-word

Original-word	Stemmed-words
Trouble	Trouble
Troubled	Trouble
Troubles	Trouble
Troublesome	Troublesome

Above process which specifies the original word, stemmed words and original-word, stemmed-word differences. After completing stemming process data owner creates a secure index for each files based on the bloom filter. The concept include bloom filter is explain detailed as follows;

3.3. Bloom Filter for create secure index file:

Bloom filter is an array of m bits to represent a set $F=\{f_1, f_2, f_3,...f_n\}$, Having n items, where m bits are initially set to 0. Hash functions with ranges of 1, 2, 3, 4, 5, and 6 are used to create the filter's hash values. Using these hash functions, each item in F is mapped to a random number uniformly distributed in the range of 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, and m. As a result, the same array index can be set to 1 numerous times for each item in the array with index of 1 for 1 I k. Check whether the bits with index hi(x) are set to 1 to see if an item y is in F. Assume that y is in F if all bits with index hi(x) are set to 1. In this case, a Bloom filter may provide a false positive. is the likelihood of receiving an incorrect diagnosis. In order to obtain a minimum false positive probability, the expression is minimized when the false positive probability is $FPB=1-e^{-kn/m})^k$. The expression $E=(e^{-kn/m})^k$ is minimized when $k = \ln 2 * (m/n)$, so the minimum false positive probability is $FPBmin=(1/2)^{\ln 2*(m/n)} \approx (0.6185)^{m/n}$.

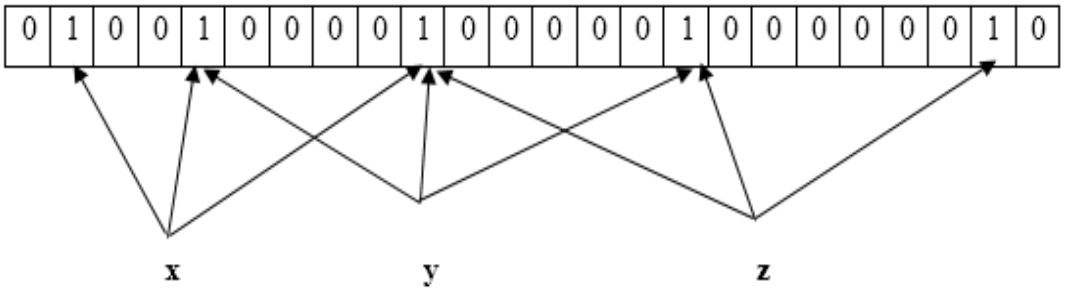

Figure 2: Secure index creation format using m-bit bloom filter

Above figure 2 specifies the secure index creation for each input files using m-bit bloom filter. After creating index, all the encrypted files as well as index are outsourcing to the cloud. The concept include encryption is explain detailed as follows; here, encryption using combined blowfish and ECC algorithm.

3.4. Encryption Using Combined Blowfish and Elliptic Curve Cryptography Algorithm:

The blowfish encryption algorithm uses only private keys for encryption purposes. Because symmetric algorithms are known as private key cryptography that uses the single private key for encryption and decryption. But here, we are using combined Blowfish and ECC for that way ECC provide public keys for decryption process. The important characteristics of Blowfish encryption algorithms are represented as follows:

Fast: It encrypts data at a pace of twenty-six clock cycles per computer memory unit on large 32-bit microprocessors.

Compact: It will just need 5K of memory to run. It is straightforward: it employs addition, XOR, and a search table with 32-bit operands. Secure: The key length is adjustable, ranging from 32448 bits to the default value of 128 bits.

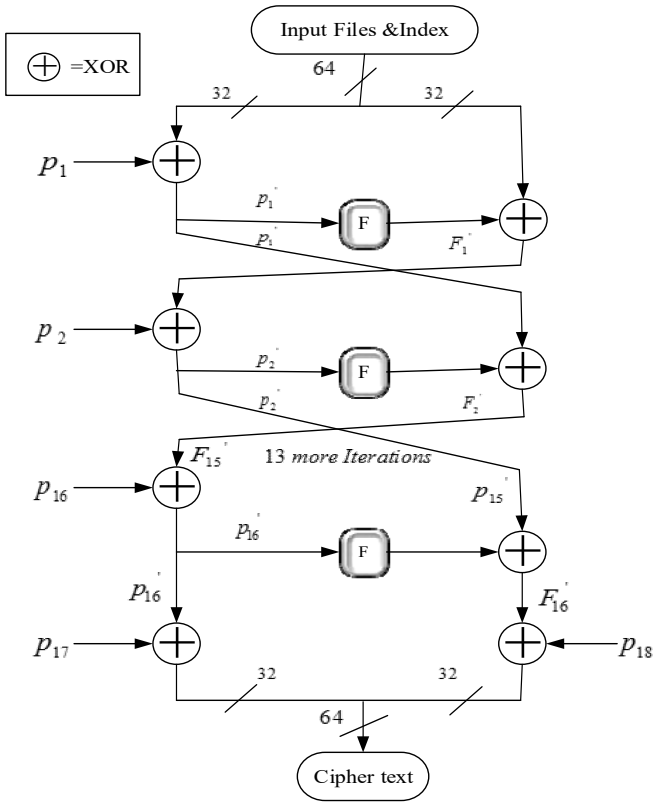

Figure 3: Blowfish Encryption Algorithm

The key-expansion and data-encryption elements of Blowfish are separate. A 4168-byte subkey array of the input key is created during the key expansion stage. There are eighteen 32-bit boxes in the P array and four 256 entry

32-bit arrays in the S array. Initializing the string consists of XORing P1's first 32 bits against the key's first 32 bits (the first 32-bit box in the P-array). Keep going backwards and forwards between key bits until you've XORed all 448 key bits, or until you've XORed the full P-array. Create a 64-bit block using the Blowfish method to encrypt all zero strings. P1 should be replaced with the first 32 bits of output, and P2 should be replaced with the second 32 bits, respectively (from the 64-bit block). A new 64-bit block will be generated when you feed the output 64 bit back into the Blowfish cypher as an input, as seen below. Following values in P-array should be replaced using block. For each P-array value and each S box, repeat the same process. These ways all the input files are encrypted using combined Blowfish and ECC. After completing the process of encryption, the encrypted data's as well as index are outsourced to the cloud server. The concept include cloud storage is explain detailed as follows;

Example:

Illustration of an Encryption Using Temporary Variables:
Step-1: Take the input
Input: Here 64 bits plain text can be taken
With keys and sub keys and with round function.
Step-2: Produce the output
Output:64 bits cipher text will be produces as output

Optimized Algorithm:

"(L0, R0) = T, dividing T into two 32-bit parts

L1 = L0 XOR P1

R2 = R0 XOR F (L1) XOR P2

.

.

.

Ln=LnXORpn

Rn=RnXORF(Ln)XORPn

C = (R18, L17)

3.5. Cloud Storage and Trapdoor:

Data encryption and indexing is done on the cloud. The next step is to generate the trapdoor code. It is possible to search by using a keyword trapdoor. An index searches for relevant files when a trapdoor query is received from a user and returns encrypted data to that user. Trapdoor users can utilize the keyword search method to find files that fit their search criteria (query request). If you have retrieved all the matching documents, you can utilize an optimization algorithm called "Oppositional Grasshopper". blowfish decrypts the original files. According to the oppositional grasshopper optimization algorithm's explanation,

3.6. Oppositional Grasshopper Optimization Algorithm (OGOA):

The Grasshopper Optimization Algorithm is used to determine the best solution. OGA stands for opposition-based grasshopper-optimization-algorithm and is an improved optimization algorithm. In this case, the "Grasshopper Optimization Algorithm (GOA)" is an improved version of a system that imitated grasshopper swarming behaviors. An "OBL" algorithm, which stands for "opposition-based learning," has been added to the Optimization Algorithm of Grasshopper. Oppositional Grasshopper Optimization, an optimization algorithm, finds the optimal solution after obtaining the matched files. For the greatest potential result, the following actions must be taken.

Step 1: Initialization

Come up with the greatest solution you can! The grass hoop algorithm is based on opposition. Optimizing algorithmic processes relies heavily on initialization, or creating solutions. In order to arrive at the optimal solution, this is a crucial step to take. This is done by encrypting his information using a public key in order to keep it safe from prying eyes and hackers. Defined trapdoor codes are necessary, therefore. In order to discover matching files as soon as the user submits their request, a keyword search approach is implemented. After retrieving matched files Oppositional Grasshopper optimization Algorithm is used to find the optimal solution.

Finally decrypts the original files based on blowfish decryption algorithm. In this step, encrypted data and the matched files are represented as follows; $ED = \{F_1, F_2, F_3, ...F_N\}$ where, F_N is the N^{th} encrypted data. $MD = \{S_1, S_2, S_3, ...F_M\}$ Where, S_M is the M^{th} matched data. After initialize the solution, the obtained solution is given for the next step i.e. oppositional solution.

Step 2: oppositional solution

To improve the searching ability, in this, the initial stage produces an initial population as well as its oppositional using OBL methodology and the subsequent stage utilizes OBL as an additional stage to update the GOA populace in every iteration. In this, oppositional solution to the initial solution is defined by the OBL methodology. After that it uses the fitness function values to decide if the oppositional is superior to the current solution. The oppositional value as well as the current values is calculated as follows;

$$\overline{a} = x + v - a \tag{1}$$

By utilizing the subsequent equation, the above definition could be generalized into n-dimensions which can be characterized mathematically as (2).

$$\overline{a_k} = x_k + v_k - a_k, \qquad k = 1,2, \dots, N \tag{2}$$

Anywhere $\overline{a} \in S^n$ is the oppositional vector from the actual vector $a \in S^n$.In this, two solutions are compared through the optimization process. The solutions are represented as (a as well as $\overline{a}$) compared to the fitness function (f) the best solutions are stored and the other is removed. The obtained solution is given for the next step i.e. fitness evaluation.

Step 3: Fitness Calculation

After generating the solution, the fitness function is evaluated and then chooses the best solution. Optimization algorithm for the most part relies upon its fitness function to acquire the best solution. The selection of the fitness is a fundamental aspect in OGOA.

$$Fitness_i = Min[\lambda 1(T) + \lambda 2(C)] \tag{3}$$

Where,

C- Cost

T- Time

task T^i_{Total} is given in equation (4).

$$T^i_{Total} = \sum T_R + \sum T_P + \sum T_W \tag{4}$$

Where,

$T_P \rightarrow$ Processing time of subtask,

$T_R \rightarrow$ receiving time of subtask,

$T_W \rightarrow$ waiting time

$$C = \sum_{i=1}^{CD} \left(\frac{No.of\ \ matched\ \ \ files}{Total\ \ encrypted\ \ \ files} \right) \tag{5}$$

Step 4: OGOA based Updating solution

Revise the answer with "Oppositional Grasshopper Algorithm based weighted fair sharing (OGWFS)" after the fitness evaluation. We can make changes to the solution by using an equation (7). The grasshopper's say x_i position can be numerically characterized using the equation below.

$$x_i = S_i + G_i + A_i \ , \qquad i = 1,2,...,N \tag{7}$$

The societal interaction of the ith grasshopper is S I which can be stated mathematically as follows:

$$S_i = \sum_{j=1, i \neq j}^{N} S(d_{ij}) \hat{d}_{ij}, \quad d_{ij} = | x_i - x_j | \tag{8}$$

D ij everywhere signifies the distance between the ith and jth grasshoppers, whereas s denotes the strength of the social forces function, which can be stated mathematically as follows:

$$S(y) = fe^{\frac{-y}{l}} - e^{-y} \tag{9}$$

Where,

$f \longrightarrow$ Attraction of intensity,

$l \longrightarrow$ Attractive length scale,

In equation 6, G_i and A_i gives the ith grasshopper's gravity force and wind advection, which may be stated mathematically as,

$$G_i = -g\hat{e}_g ,$$

$$A_i = u\hat{e}_w \quad (10)$$

Anywhere,

$g \longrightarrow$ Gravitational constant,

$u \longrightarrow$ Constant drift,

Although $e_g$and $e_w$symbolize the The wind direction and the earth's centre unity vector, respectively. Still, equation 7 couldn't be used to solve the optimization problem right away, so rewrite it as follows:

$$x_i = c\left(\sum_{j=1, i\neq j}^{N} c^{\frac{u-l}{2}} S(|x_j - x_i|)\frac{x_j - x_i}{d_{ij}}\right) + \hat{T}d \quad (11)$$

Where,

$u \longrightarrow$ Upper bound of the search space

$l \longrightarrow$ Lower bound of the search space

$Td \longrightarrow$ Best solution value

Step 5: Termination Phase:

The optimization process discontinues its execution when it achieves the most extreme number of matched files as well as when the best solution is found. Once the optimal solutions are attained means then move to the decryption process. The procedure involved in decryption phase is discussed as follows;

3.7. Decryption Phase

In this phase, the matched files are decrypted based on blowfish decryption algorithm. Blowfish decryption algorithm is used to decrypt the matched files and give the original result. Blowfish is a symmetric algorithm so that, the same procedure is used for decryption as well as encryption, except that p1, p2... p18 are used in reversed order. The only difference is that the input to the encryption is plain text and for decryption, the input is cipher text.

Example:

Illustration of an Encryption Using Temporary Variables:

Step-1: Take the input

Input: Here 64 bits cipher text can be taken

With keys and sub keys and with round function.

Step-2: Produce the output

Output:64 bits plain text will be produces as output

Optimized Algorithm:

"(L0, R0) = T, dividing T into two 32-bit parts

L1 = L0 XOR P1

R2 = R0 XOR F (L1) XOR P2

.

.

.

Ln=LnXORpn

Rn=RnXORF(Ln)XORPn

P = (R18, L17)

4. Result and Discussion

Our experiments are conducted in a Cloud Sim with JAVA. Our proposed research is applied over Security and Privacy Preserving Keyword Search for Cipher Text Retrieval in Cloud Computing based on Oppositional Grasshopper optimization using Java and a series of experiments. They were performed on a PC with Windows XP Operating system at 2 GHZ dual core PC machine with 4 GB main memory running a 64-bit version of Windows 2007. The experiment result of the proposed work is described below:

4.1. Experiment result and performance analysis:

The main idea of our proposed methodology is to improving the data security and privacy. Our proposed work uses Security and Privacy Preserving Keyword Search for Cipher Text Retrieval in Cloud Computing based on Oppositional Grasshopper optimization. The proposed result is compared with existing ESPP as well as TRSE algorithms [7]. The following figure 4 to 13 shows the performance of the proposed approach using this configuration,

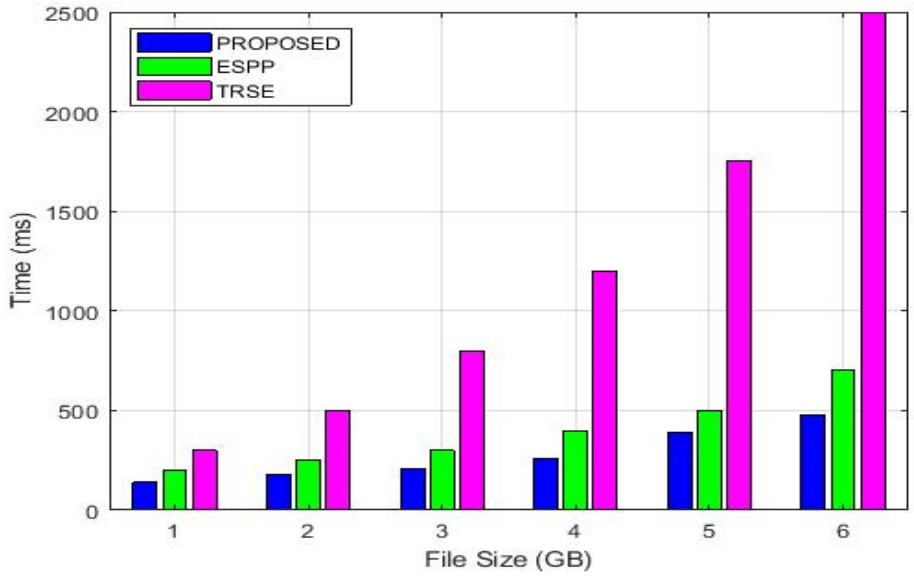

Figure 4: Cost of Computation (Encryption)

Figure 4 shows the presentation investigation of proposed against existing utilizing the calculation cost of information proprietor for encoding the record. Here, our proposed approach used least expense to accomplish the objective. Figure 4 shows that proposed encryption process is very productive in light of the fact that it takes just 1 modular augmentation to scramble h pieces of plaintext. By contrasting ESPP and TRSE encryption strategy, the proposed encryption takes less calculation cost for longer records (GB). The calculation cost of key age and file creation is unimportant contrast with encryption process.

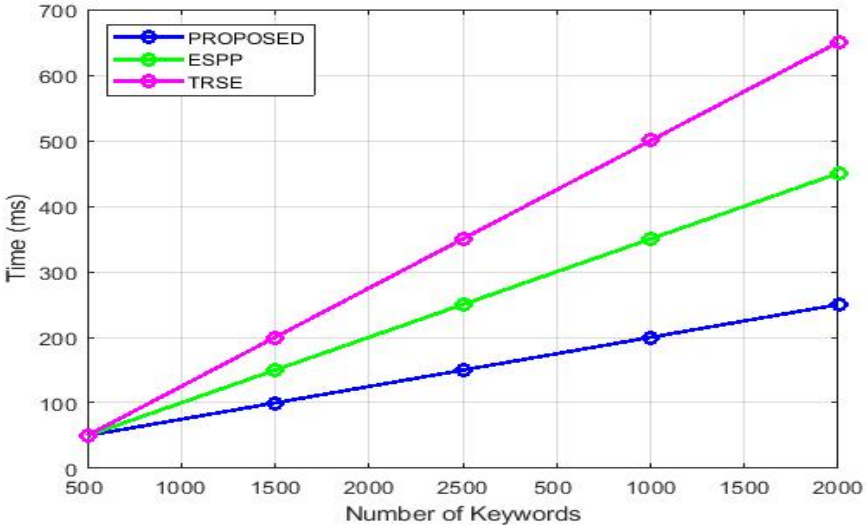

Figure 5: The time to generate Trapdoor on different number of keywords

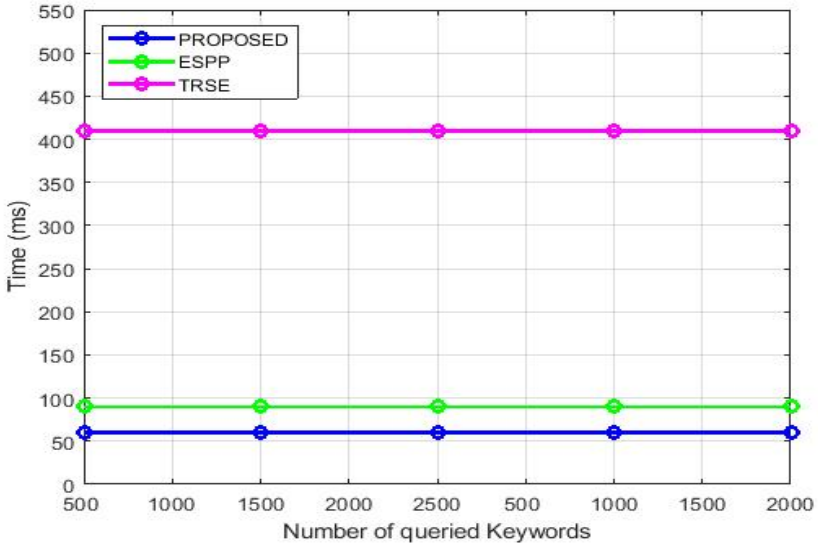

Figure 6: The time to generate Trapdoor for different number of queried keywords

Figure 5 shows an opportunity to produce a secret entryway of various lengths of catchphrases and Figure 6 shows trapdoor for questioned watchwords where number of catchphrases 4000.In the two cases, our proposed plot can be proficient than ESPP just as TRSE calculations, since they encode the trapdoor.

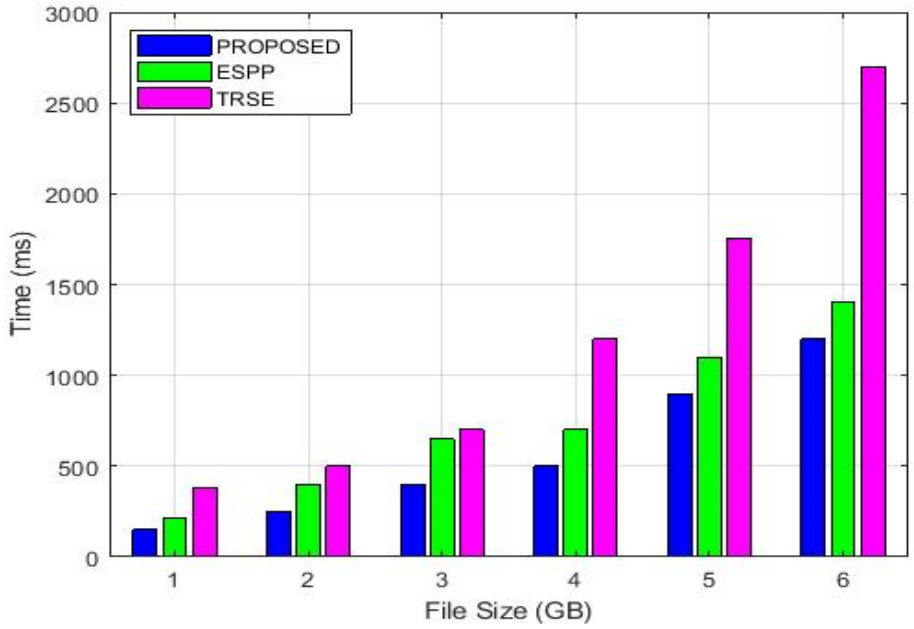

Figure 7: Computation Cost (Decryption)

Similar to the TRSE plot, the main difference between proposed and existing ESPPP is that the former uses homomorphic encryption and decoding calculations to encode and unscramble the records, requiring more calculation cost, whereas the latter uses probabilistic public key encryption and decoding calculations, requiring less calculation cost. We guarantee that our proposed scheme decreases the computational overhead of a small customer; as a result, it is more adaptive to asset-demanded mobile phones in the cloud than ESPP and TRSE.

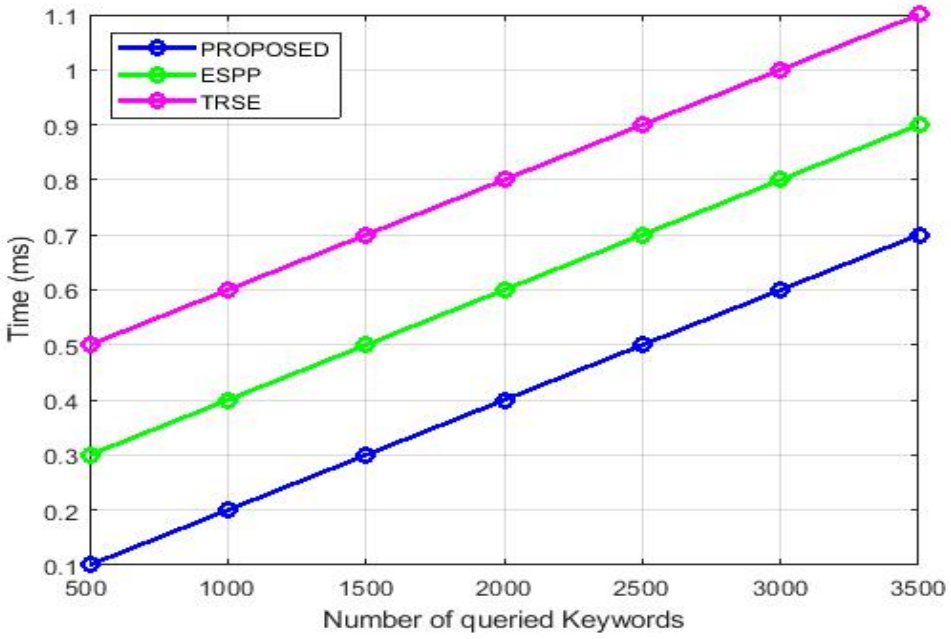

Figure 8: The time of the server to search the files based on queried keywords

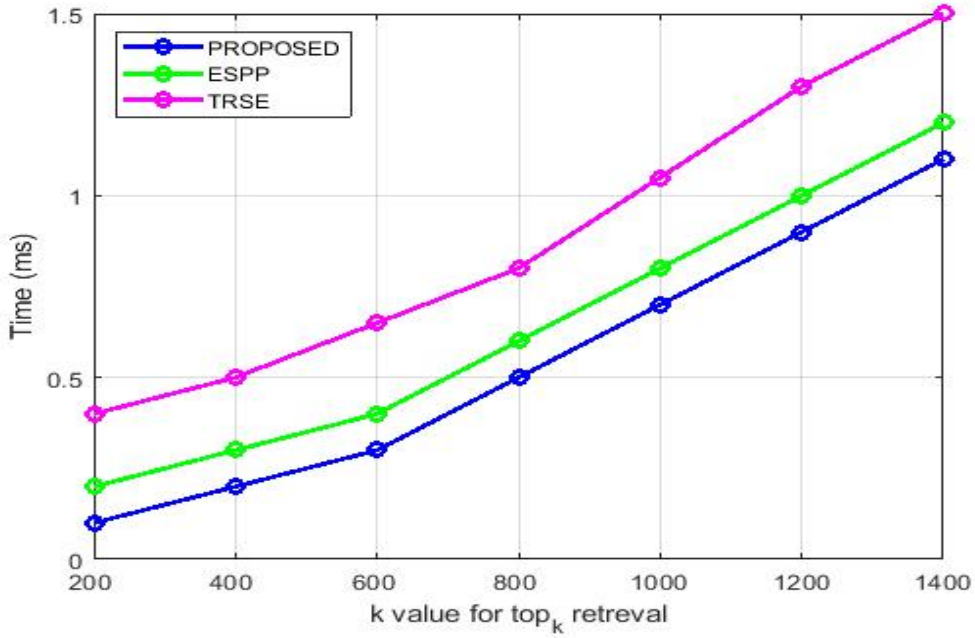

Figure 9: The time of server for selecting Top-k files where total number of files

Figure 8 shows an opportunity to look through the records dependent on trapdoor;the search time incorporates getting the document passage list in the file utilizing Bþ tree.The over all hunt time of proposed is nearly as proficient as thought about existing ESPP and TRSE. Essentially, Figure 7 shows the hour of worker to choose the top-k documents from the every single coordinated record dependent on correlations center registered by information proprietor. From Figure 9, we can see that top-k record recovery time and against the estimation of k increments for a similar file of our proposed shapes better than ESPPA and TRSE calculation cost of worker for recovery of top-k documents from entire record assortment is less contrasted with existing ESPP and TRSE scheme.computation cost of worker for retrievalof top-k documents from entire document assortment islesscompared to existing plans.

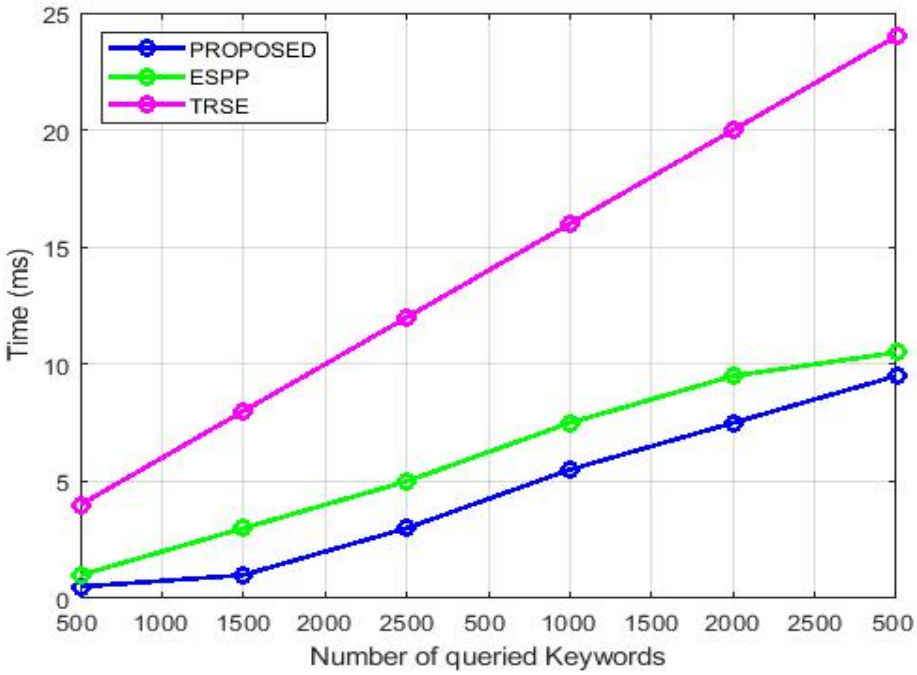

Figure 10: Communication cost

We investigate the communication cost of proposed conspire between the approved client and worker during document recovery process. In Figure 8, we show correspondence cost of our plan and contrast results and ESPP and TRSE conspire. From Figure 8, we can see that our proposed conspire inconceivably decreases the

correspondence overhead weight contrasted with the current plans, in light of the fact that proposed utilize one round correspondence between the client and worker to recover the coordinated records back, whereas existing plan utilizes two round correspondence forms between the client and worker.

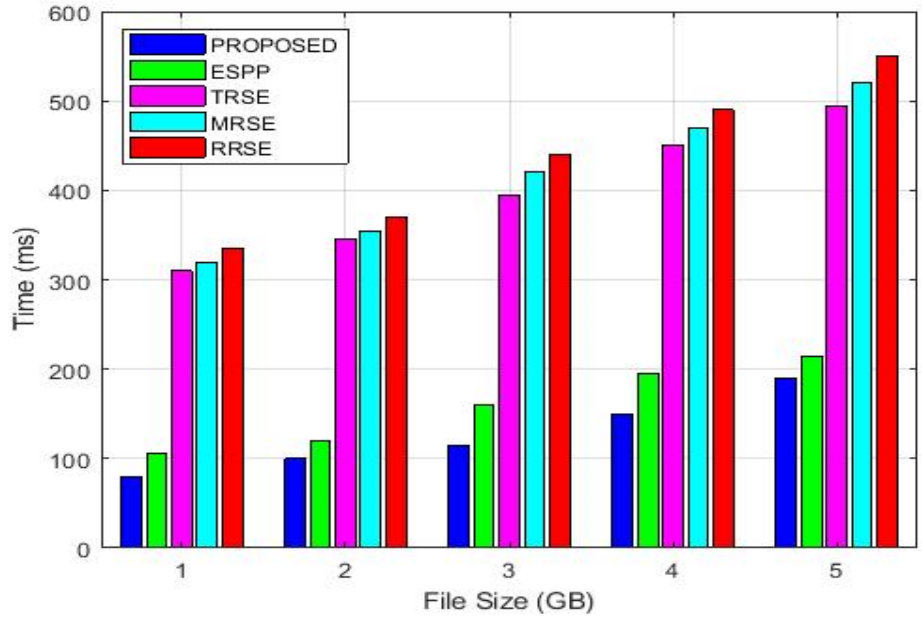

Figure 11: Computation Cost of Data Owner for encrypting the files on real smart phone

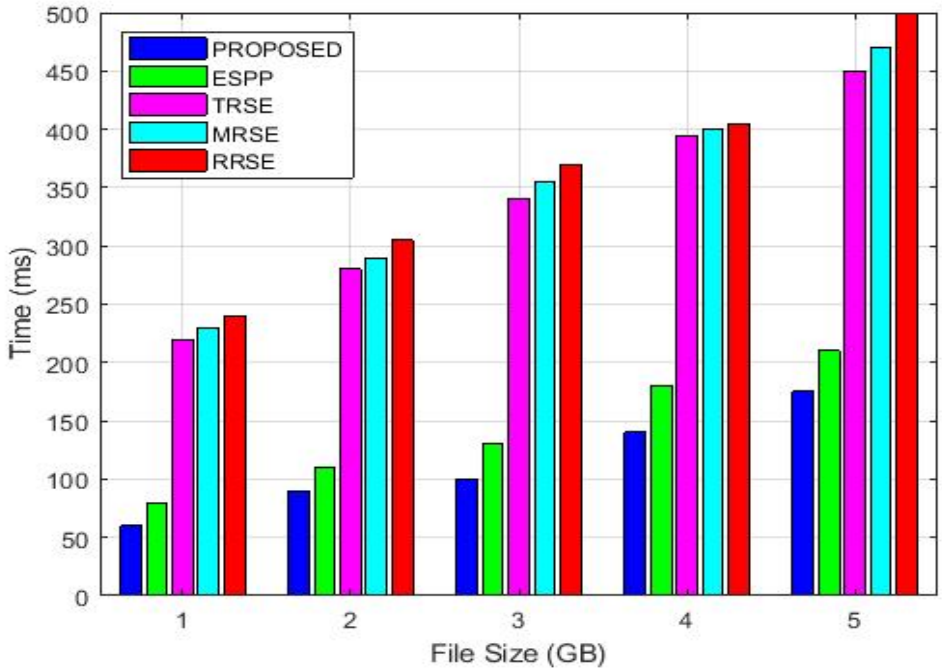

Figure 12: Computation Cost of Authorized user for decrypting the files on real smart phone

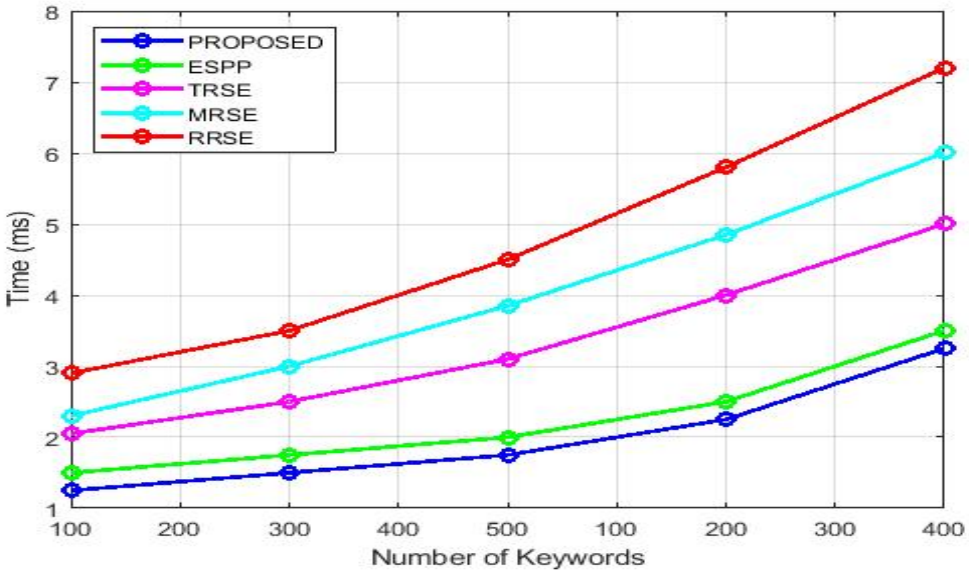

Figure 13: The time to generate Trapdoor on different number of keywords on real smart phone

Figure 11 and 12 show that calculation cost of proposed which is less proficient than existing strategies, for example, EPSS, TRSE, MRSE and RRSE. Since, the proposed approach utilizes the probabilistic open key encryption and Decryption calculations though existing strategies utilizes deterministic encryption and unscrambling calculations. Figure 13 shows an opportunity to create a hidden entrance of various lengths of watchwords. As indicated Figure 13, an opportunity to create secret entryway is constant when the quantity of questioned watchwords increments. The all-out time cost of secret entrance age of proposed is effective than existing methodologies ESPP, TRSE, MRSE and RRSE. Since existing utilize enormous key size for producing trapdoor.

5. Conclusion

In this paper, we introduce the utilization of Security and Privacy Preserving Keyword Search for Cipher Text Retrieval based on Oppositional Grasshopper optimization. In order to reduce the problem of security in cloud data our proposed methodology using combination of Blowfish algorithm and Elliptic curve cryptography (Blowfish+ECC). Our research study mainly focusses on two stages of working process like; public key encryption as well as keyword retrieval. In the phase of public key encryption, ECC provide access control, Porter stemming is used to extract keywords, Data owner creates a secure index for each file using bloom filter. After that data owner encrypt the files. In the next phase, trapdoor code is generated. Matching files are searched using keyword search technique and Oppositional Grasshopper optimization Algorithm is used to find the optimal solution. At last, decrypts the original files based on blowfish decryption algorithm. Compared to the existing Security and Privacy Preserving Keyword Search our proposed achieves higher outcomes.

Reference:

[1] Wei, Lifei, Haojin Zhu, Zhenfu Cao, Xiaolei Dong, Weiwei Jia, Yunlu Chen, and Athanasios V. Vasilakos. "Security and privacy for storage and computation in cloud computing." Information Sciences 258 (2014): 371-386.

[2] Wang, Boyang, Baochun Li, and Hui Li. "Panda: Public auditing for shared data with efficient user revocation in the cloud." IEEE Transactions on services computing 8, no. 1 (2015): 92-106.

[3] Liu, Chang, Rajiv Ranjan, Chi Yang, Xuyun Zhang, Lizhe Wang, and Jinjun Chen. "MuR-DPA: Top-down levelled multi-replica merkle hash tree based secure public auditing for dynamic big data storage on cloud." IEEE Transactions on Computers 64, no. 9 (2015): 2609-2622.

[4] Kumar, Raman, and Gurpreet Singh. "Analysis and design of an optimized secure auditing protocol for storing data dynamically in cloud computing." Materials Today: Proceedings 5, no. 1 (2018): 1037-1047.

[5] Ni, Jianbing, Yong Yu, Yi Mu, and Qi Xia. "On the security of an efficient dynamic auditing protocol in cloud storage." IEEE Transactions on Parallel and Distributed Systems 25, no. 10 (2014): 2760-2761.

[6] Song, Wei, Bing Wang, Qian Wang, Zhiyong Peng, Wenjing Lou and Yihui Cui, "A privacy-preserved full-text retrieval algorithm over encrypted data for cloud storage applications", Elsevier on Journal of Parallel and Distributed Computing, pp.1-25, 2016.

[7] Pasupuleti, Syam Kumar, Subramanian Ramalingam and RajkumarBuyya, "An efficient and secure privacy-preserving approach for outsourced data of resource constrained mobile devices in cloud computing", Elsevier on Journal of Network and Computer Applications, Vol.64, pp.12-22, 2016.

[8] AqeelSahi, David Lai and Yan Li, "Security and privacy preserving approaches in the Health clouds with disaster recovery plan", Elsevier Journal of Computers in Biology and Medicine, Vol.78, pp.1-8, 2016.

[9] Worku, Solomon Guadie, Chunxiang Xu, Jining Zhao and Xiaohu He, "Secure and efficient privacy-preserving public auditing scheme for cloud storage", Elsevier on Computers & Electrical Engineering, Vol.40, No.5, pp.1703-1713, 2014.

[10] Liu, Qin, Guojun Wang and Jie Wu, "Secure and privacy preserving keyword searching for cloud storage services", Elsevier on Journal of network and computer applications, Vol.35, No.3, pp.927-933, 2012.
[11] Dong, Xin, Jiadi Yu, Yuan Luo, Yingying Chen, GuangtaoXue and Minglu Li, "Achieving an effective, scalable and privacy-preserving data sharing service in cloud computing", Elsevier on computers & security, Vol.42, pp.151-164, 2014.
[12] Razaque, Abdul and Syed S. Rizvi, "Triangular data privacy-preserving model for authenticating all key stakeholders in a cloud environment", Computers & Security, Vol.62, pp.328-347, 2016.
[13] Aldeen, Yousra Abdul Alsahib S., MazleenaSalleh and YazanAljeroudi, "An innovative privacy preserving technique for incremental datasets on cloud computing", Elsevier on Journal of Biomedical Informatics, Vol.62, pp.107-116, 2016.
[14] Zhang, Yinghui, Xiaofeng Chen, Jin Li, Duncan S. Wong, Hui Li and Ilsun You, "Ensuring attribute privacy protection and fast decryption for outsourced data security in mobile cloud computing", Elsevier on Information Sciences, pp.1-20, 2016.
[15] Liu, Zheli, Xiaofeng Chen, Jun Yang, Chunfu Jia and Ilsun You, "New order preserving encryption model for outsourced databases in cloud environments", Elsevier on Journal of Network and Computer Applications, Vol.59, pp.198-207, 2016.
[16] Mortaza S. Bargh, Sunil Choenni and Ronald Meijer, "On design and deployment of two privacy-preserving procedures for judicial-data dissemination", Elsevier on Government Information Quarterly, pp.1-13, 2016.
[17] Vennila.S and J. Priyadarshini, "Scalable Privacy Preservation in Big Data a Survey", Elsevier on Procedia Computer Science, Vol.50, pp.369-373, 2015.
[18] Zhang, Xuyun, Chang Liu, Surya Nepal and Jinjun Chen, "An efficient quasi-identifier index based approach for privacy preservation over incremental data sets on cloud", Elsevier on Journal of Computer and System Sciences, Vol.79, No.5, pp.542-555, 2013.
[19] Yu, Yong, Man Ho Au, Giuseppe Ateniese, Xinyi Huang, Willy Susilo, Yuanshun Dai, and Geyong Min. "Identity-based remote data integrity checking with perfect data privacy preserving for cloud storage." IEEE Transactions on Information Forensics and Security 12, no. 4 (2017): 767-778.
[20] Liu, Xuefeng, Wenhai Sun, Wenjing Lou, Qingqi Pei, and Yuqing Zhang. "One-tag checker: Message-locked integrity auditing on encrypted cloud deduplication storage." In INFOCOM 2017-IEEE Conference on Computer Communications, IEEE, pp. 1-9. IEEE, 2017.
[21] Zhang, Yue, Jia Yu, RongHao, Cong Wang, and Kui Ren. "Enabling efficient user revocation in identity-based cloud storage auditing for shared big data." IEEE Transactions on Dependable and Secure Computing (2018).
[22] Kumar, Raman, and Gurpreet Singh. "Analysis and design of an optimized secure auditing protocol for storing data dynamically in cloud computing." Materials Today: Proceedings 5, no. 1 (2018): 1037-1047.

Author Biography

Kasiviswanadham Y received the B.Tech (CSIT) from Sri Krishna Devaraya University, Anantapur, India in 2004 and M.Tech (CSE) from S V University, Tirupati, India in 2008. He is pursuing Ph.D in S V University. He is working as Associate Professor in Gudlavalleru Engineering College. He has 17 Years of teaching experience. He has published 15 International Journals and Conferences. His interest areas are Information Security, Cloud Computing and Internet of Things. He is also member in IEEE, ACM, CSI, ISTE, IE (India), IAENG.

Dr Ch. D. V. Subba Rao received the B.Tech (CSE) from S V University college of Engineering, Tirupati, India IN 1991, ME (CSE) from M K University, madras in 1998 and he was the first Ph.D. Award in CSE from S V University. Tirupati in 2008. He has got 30 Years of teaching experience. He served as the Head. Dept of Computer Science and Engineering, Tirupati, India during 2008-11. His areas of interest include Distributed Systems, Advanced Operating Systems and Advanced Computing. He chaired and served as reviewer of IAENG and IASTED international conferences. He has published more than 53 papers in international journals and 11 international conferences. These publications appear in IEEE ACM, Elsevier, Springer and other Digital libraries. He guided five Ph.Ds and guiding six Ph.Ds. He visited Austria, Netherlands, Belgium, Hon-Kong, Thailand, Germany and Singapore.