SECURE PHRASE SEARCH IN CLOUD IOT USING BLOWFISH ENCRYPTION SCHEME

SHAKEEL JUMAN TP
Assistant Professor in Computer Science and Application,
Centre for Computer Science and Information Technology (CCSIT)

Abstract: With the increasing number of cloud computing the storage also increases, to this condition the privacy of the data remains question, it is still insecure in terms of retrieving and searching process. In this phase search it provide ameans of searching a content to the user which allows retrieval of data from information systems. Phrase search is one of many search administrators that are standard in search engine method. While searching the data the security may influence in numerous factors, the Data security in the cloud has consistently been a major worry for both of its clients and cloud service providers. This is because of danger issues from the outsider of the data and from its internal partners, that is their workers. So keeping up this security flawless is continually being a migraine and challenging task in the cloud because of its colossal structure. Commonly, to accomplish this, data is constantly kept in encrypted format and take care that data truly isn't undermined during the audition process and other interior exercises. Searchable Encryption (SE) techniques have been broadly concentrated to empower searching on the data while they are encrypted.These techniques empower different kinds of search on the encrypted data and offer various security. In this paper proposed a Blowfish Encryption Scheme (BES) for secure phrase search in cloud IOT. The proposed works handles with data transferring, data indexing, data sliding, encryption, decryption, recovery and merging process. The BES algorithm was created to give the security to the huge data before storing it in to the cloud. This proposed work presents a novel methodology in the field of encrypted searching that permits encrypted phrase searches.

Keywords: Cloud, Security, Phrase Search, Encryption and Blowfish.

I. INTRODUCTION

Cloud computing is one of the quickest developing innovation. The Cloud depends on different various services that contain different advantages going from programming advancement stages, server, to storage and far off computing. This has all been conceivable because of the coming of the web and it is the way these services are connected to the client; through the web. There has been an enormous scope reception of this technique as it is exceptionally helpful. This has additionally brought about expanded research into this stage [1].

Because of expanded research here, there has been a ton of improvement as far as the foundation that bolsters this stage. The framework has been profoundly expanded to help the developing number of clients. A great deal of associations, just as individual clients, are embracing this stage, that is quickening the development of this foundation significantly further. A significant increment in the framework likewise encourages lower costs that can permit this innovation to enter the standard market [2].

Cloud computing is exceptionally helpful for various applications, going from putting away some data to setting up enormous programming computational focuses. Larger part of the cloud has been utilized as a storage elective. Being ready to let loose your neighborhood storage and as yet having the option to get to your records anyplace on any gadget his advantageous and profoundly worthwhile. This has taken clients leap over to the cloud for their storage purposes promptly [3]. The client can store and access their data consistently and furthermore have the option to have better client experience as the neighborhood storage on your gadget has been opened up and the

data is being accessible to the client anyplace on the planet on any gadget.

There are a couple of disadvantages to the utility gave by the Cloud. As the data from the associations and people have been moved to the cloud, the clients need to relinquish their control the data and hand it over to the cloud. To guarantee the wellbeing of the touchy data and forestall any data spillages, encryption is continually being promoted as the safeguard instrument that can be profoundly secure and strong. This is an extremely helpful element that would shield the data and defend it from the interlopers and assailants [4]. This is a serious shrewd component to be actualized to ensure the data transferred onto the cloud is sheltered. Yet, this is an excess component that lessens the User Experience for the security on the cloud.

Phrase searching permits clients to recover content from information frameworks, (for example, documents from record storage frameworks, records from databases, and site pages on the web) that contains a particular request and blend of words characterized by the client [5]. Encryption is a strategy that makes sure about information by making it indecipherable or undefined from arbitrary commotion to anybody that doesn't have some special information, a key. The act of utilizing cryptography to scramble delicate information has been around for times. For a huge number of years a majorette was that the encrypted information was unusable until decoded. This served well until later, when the tremendous number of documents waiting be encrypted has made decoding singular documents to discover query results infeasible by and by. Searchable encryption was created to take care of the issue of how to discover keywords in documents that are encrypted without unscrambling the whole corpus set [6].

Search engines right now incorporate an assortment of highlights that permit clients to alter their query for their task. For the task of discovering website pages on a point, straightforward keyword searching is accessible in two structures – "any of these terms" and "these terms". For the task of phrase source searching, for instance in citations or verses, phrase coordinating is accessible as "definite phrase". "Precise phrase" coordinating necessitates that all terms be available in the request given. "These terms" searches necessitate that the entirety of the terms gave be found in the recovered record, yet request doesn't make a difference [7, 8]. "Any of these terms" searching requires that in any event one of the terms gave be found in the recovered archive and request, once more, doesn't make a difference.

An information recovery framework utilizes phrases to index, recover, compose and depict documents. Phrases are distinguished that predict the presence of different phrases in documents or cloud. Documents are the indexed by their included phrases. Related phrases and phrase augmentations are additionally recognized. Phrases in a query are recognized and used to recover and rank documents. Phrases are additionally used to bunch documents in the search results, make record portrayals, and kill duplicate documents from the search results, and from the index.

II. LITERATURE REVIEW

Kathryn Patterson et al [9] Exact phrase coordinating is an incredible asset to rapidly recover results when an adequate segment of the content is precisely given as the query. In the event that the part of the content isn't totally precise, phrase searching will fizzle. A technique must be utilized which will uphold severe enough conditions to accomplish high precision while considering botches in the content gave. Here we build up a strategy utilizing nearness conditions to search for cites from films and think about the outcomes against the Vector Space Model. Starting outcomes show a promising exactness in overabundance of 78% for documents being effectively positioned inside the best ten outcomes.

Zittrower et al [10] As cloud computing is expanding in fame, it is hard to both keep up privacy in datasets while as yet giving satisfactory recovery and searching strategies. This paper presents a novel methodology in the field of encrypted searching that permits both encrypted phrase searches and vicinity positioned multi-keyword searches to encrypted datasets on untrusted cloud. By putting away encrypted keyword-area data alongside uncommonly shortened encrypted keyword indexes in a social database, we can take into

consideration a full scope of search highlights in our encrypted searches, something that has never been cultivated. Moreover, our methodology allows the encrypted corpus and index to both be put away on cloud data servers. We alter right now accessible open-source search engine programming to finish a model and give results from investigates a huge scope genuine world dataset that has the greater part 1,000,000 documents.

Li, Mingchu, et al [11] Security of cloud stockpiling has drawn a consistently expanding number of concerns. In the searchable encryption, various past game plans can let people recuperate the archives containing single keyword or conjunctive keywords by taking care of encoded records with data lists. In any case, searching reports with a phrase or successive keywords is up 'til now a remained open issue. In this paper, using the relative positions, we propose a gainful plan LPSSE with symmetric searchable encryption that can maintain scrambled phrase searches in cloud stockpiling. Our plan relies upon non-flexible security definition by R. Curtmola and with lower costs of transmission and capacity than existing systems. Furthermore, we join a couple of parts of right now beneficial search engines and our abilities to complete a model. The assessment results furthermore show that our plan LPSSE is available and profitable.

Shen, Meng, et al [12] propose P3, a powerful privacy-sparing phrase search conspire for clever encoded data preparing in cloud-based IoT. Our plan abuses the homomorphic encryption and bilinear manual for choose the territory relationship of various addressed keywords over scrambled data. It in like manner utilizes a probabilistic secret entryway age algorithm to guarantee customers' search plans. Thorough security examination shows the security guarantees achieved by P3. We execute a model and lead expansive tests on authentic world datasets. The appraisal results show that differentiated and existing multi-keyword search plans, P3 can amazingly improve the search precision with moderate overheads.

Poon, Hoi Ting et al [13] proposed answers for conjunctive keyword search, it is starting late that researchers began examining phrase search once again scrambled data. In this paper, we present a plan that melds the two functionalities. Our answer uses symmetric encryption, which gives computational and capacity profitability over plans subject to open key encryption. By pondering the quantifiable properties of ordinary lingos, we had the alternative to design files that on a very basic level diminish stockpiling cost when appeared differently in relation to existing plans. Our answer thinks about fundamental situating of results and requires a low stockpiling cost while giving report and keyword security. By using both the list and the scrambled records to performs searches, our plan is moreover right now the fundamental phrase search plot fit for searching for non-ordered keywords.

III. PROBLEM DEFINITION

Security of cloud stockpiling has drawn a consistently expanding number of concerns. In the searchable encryption, various past plans can let people recuperate the archives containing single keyword or conjunctive keywords by taking care of scrambled reports with data records [14]. Be that as it may, searching documents with a phrase or back to back keywords is as yet a stayed open issue. Searchable encryption is definitely not another idea yet all momentum strategies have fizzled in different perspectives that shield them from getting normal or standard. Most proposed strategies use progressed numerical structures, for example, Bloom channels or secret entryways however they ordinarily just take into account Boolean searches and don't uphold phrase searching. A keyword search utilizing terms of a phrase can make issues, if there are insufficient extraordinary and regularly happening terms gave in the query.

IV. PROPOSED WORK

The ability to perform phrase search in encoded data is starting late explored by researchers. Phrase search grants recovery of records containing an exact phrase, which expects a critical part in numerous AI applications for cloud-based IoT, for instance, shrewd clinical data examination [15]. In order to shield sensitive data from being spilled by specialist organizations, reports (e.g., focus records) are by and large encoded by data owners before being moved activities to the cloud. This, regardless, makes the search movement a very testing task. An ideal

searchable encryption conspire achieves anyway numerous features of flow search engines as could sensibly be normal and besides guarantees the data and list. However, most existing proposition fall flat in different angles and they basically uphold searching single keyword. Searchable encryption was imagined to take care of the issue of how to discover keywords in documents that are encrypted without decoding the whole corpus set. In this paper, we look at the issues of utilizing phrase searching to discover known content and proposed a Blowfish Encryption Scheme (BES) for secure phrase search in cloud. The proposed works handles with data transferring, cutting, indexing, encryption, dissemination, decryption, recovery and consolidating process. The BES algorithm was created to give the security to the enormous data before putting away it in to the cloud.

a. System Model

The privacy-preserving phrase search framework over encrypted data includes three substances, in particular a data proprietor, a cloud server, and one or numerous clients. The data proprietor produces a protected searchable index for the archive set and redistributes the safe index alongside the encrypted record set to the cloud server [16]. At the point when an approved client, state Alice, plays out a phrase search over the encrypted documents, she initially obtains the relating trapdoor from the data proprietor through the search control component (e.g., broadcast encryption), and afterward presents the trapdoor to the cloud server. We accept that both the client and the data proprietor have restricted calculation and storage limits on a handy premise. Existing key administration instruments can be utilized to deal with the encryption capacities of approved clients.

b. Security Model

Like the current searchable encryption arrangements, we consider the cloud server as a legit however inquisitive enemy. That is, the cloud server would sincerely follow the predesigned phrase search conventions and accurately offer the relating types of assistance to clients, in any case, it might be interested about the substance of the documents and endeavor to get familiar with extra information by examining the trapdoor and indexes [17]. For example, it would derive the keywords in the index and trapdoors, just as their areas in the documents. Our scheme targets securing privacy related with the phrase of search activity, which comprises of three kinds of privacy, in particular the report set privacy, the index privacy, and the trapdoor privacy. The report set privacy can be effortlessly accomplished by scrambling the documents utilizing a square code, for example, AES, before re-appropriating them to the cloud server.

c. Secure phrase search using blowfish

This section presents the proposed privacy-securing phrase search plot over scrambled data. To deal with the trial of choosing the positional relationship of addressed keywords over scrambled data, were sort to the homomorphic encryption and bilinear guide, which engages the client to get exact search results from a lone correspondence with the cloud worker. As the phrase search is interesting occurrence of multi-keyword search, our answer can similarly perform conjunctive multi-keyword search capably [18]. It equips more conspicuous security with less key length and there is no persuading inspiration to store any private key any place. Blowfish algorithm is more secure to consider other symmetric key checks, and pass on best outcome for less managing time and changes. To becomes the key size of blowfish assessment. The blowfish algorithm is more secure to look at other symmetric key figurings, and make best outcome for less preparing time and changes. The key size of blowfish tally 128 to 448, it gives more protection to the messages and gives fantastic security to look at other symmetric figurings. Blowfish is depicted as a symmetric square figure calculation. On a fundamental level it utilizes a relative riddle key to both the encryption and unscrambling system of messages. Here the square size for Blowfish is 64 pieces; messages that aren't an outcome of 64-bits in size must be walked. It utilizes a variable – length key respect, from 32 pieces to 448 pieces. It is hold for applications where the key isn't moved as habitually as could sensibly be normal. It is generously snappier than most encryption figurings when acted in 32-bit chip with immense data holds. Blowfish is a keyed symmetric square figure masterminded in 1993 by Bruce Schneider. Schneider orchestrated Blowfish as an overall supportive figuring, expected as a decision as opposed to the creating DES. Blowfish has a 64-digit square size and a variable key length from 32 pieces up to 448 pieces. 18 sub-keys are gotten from a solitary starting key. It requires mean 521 cycles to make all vital sub keys. It utilizes design arranging system to rank the turn of events and it is essential to

see the individual lead standards in progressive learning action with the assistance of the procedure mining methodology

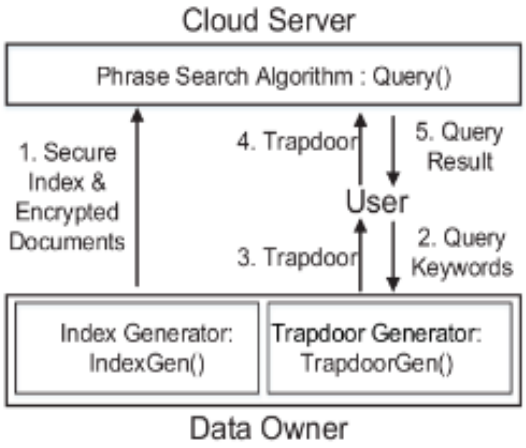

Fig 1: Proposed model

Index Generator: which is executed on the data owner side. It acknowledges the records as the data and yields the looking at secure index, similarly as the encoded archives.

Trapdoor Generator: which is also executed on the data owner side. Given a customer's addressed phrase, it creates the contrasting secure trapdoor and answers with the customer.

Phrase Search Algorithm: which is executed on the cloud worker side. In the wake of tolerating a trapdoor from a customer, it plays out a phrase search procedure over the secured index and returns the search results.

To help phrase search, we impact the turned around index structure and store the keyword zones close by the chronicle perceived. The phrase search philosophy can be portrayed as follows. Exactly when the cloud worker gets the trapdoor for a specific phrase inquiry from a customer, it at first finds the turned around records for the addressed keywords, and subsequently finds the archives that contain the aggregate of the addressed keywords. Starting now and into the foreseeable future, the cloud worker perceives whether the territories of the keywords are consecutive and returns simply the material archives that contain the particular phrase.

Algorithm 1: phrase search

1. Keygen(1^s) :

   Generate random secret key $\mathbf{SK}$: $x, y, z \leftarrow \{0,1\}^s$ and output $\mathbf{SK}$
2. BuildIndex(SK,**D**) :
   a) Generate linked list collection $\mathbf{L}$ from document collection $\mathbf{D}$
   b) Initialize a global counter $ctr \leftarrow 1$
   c) For each L_i in $\mathbf{L}$:
   - Generate $k_{i,0} \leftarrow \{0,1\}^l$
   - Add $(\pi_z(w_i), <\psi_x(\text{ctr}) \| k_{i,0} > f_y(w_i))$ into look-up table $\mathbf{T}$, where $\psi_x(\text{ctr})$ represents the index of $N_{i,1}$ in array $\mathbf{A}$

     For each $N_{i,j}$ in L_i :
     - Generate $k_{i,j} \leftarrow \{0,1\}^l$ and $value \leftarrow < id(D_{i,j}) \| k_{i,j} \| \psi_x(\text{ctr}+1) \| h(r) \| r' >$, where $\psi_x(\text{ctr}+1)$ represents the index of next node and for the last node in L_i set this segment to a invalid value.
     - Encrypt $value$ $\eta_{k_{i,j-1}}(value)$ and store it in $\mathbf{A}[\psi_x(\text{ctr})]$
     - $ctr \leftarrow ctr + 1$

   d) Output $\mathbf{I} \leftarrow (\mathbf{A}, \mathbf{T})$
3. Trapdoor(SK,w) : Output $T_w \leftarrow (\pi_z(w), f_y(w))$
4. Trapdoor′(SK, *phrase*) :
   a) Preprocess the *phrase* (tokenization, skip stop words and turn to lowercase)
   b) Output $T_{phrase} \leftarrow (T_{w_1}, ..., T_{w_m})$, where T_{w_i} equals $(\pi_z(w_i), f_y(w_i))$
5. Search($\mathbf{I}, T_{phrase}$) :
   a) If($|phrase| = 1$):
   - Let $(\alpha, \beta) \leftarrow T_{phrase}$, then retrieve α from table $\mathbf{T}$, $\omega \leftarrow \mathbf{T}[\alpha]$
   - $< \theta \| \kappa > \leftarrow \omega \oplus \beta$
   - Decrypt the node at $\mathbf{A}[\theta]$ with key κ, then get the nodes of linked list L one by one.
   - Output the document identifiers contained in L.

   b) If($|phrase| > 1$):
   - Search the trapdoors $(T_{w_1}, ..., T_{w_m})$ respectively, then return the result using intermediate result $(L_i, ..., L_m)$ and position information.

Phraseacknowledgment:To ensure the phrase area privacy, we encode the keyword area over the encryption scheme in this scheme, we just distribute $(n, G, G_T, e)$to the server as the public key. Expect that an and b speak to areas of two unique keywords in an equivalent record. Without loss of consensus, we likewise expect that a < b. On the off chance that these two keywords are continuous, we have a−b+ 1 = 0, i.e., b−a= 1. To decide the connection among an and b based on their ciphertextsgahr1andgbhr2, the cloud server sets x=a−b+ 1and changes this issue to a

comparable issue of deciding if x is the code text of 0 is appeared in Eq1.

$$E(x) = E(a - b + 1) = g^{a} h^{r_1} \cdot (g^{b} h^{r_2})^{-1} \cdot g^{1} h^{r_3} = g^{x} h^{r} \quad \text{(Eq 1)}$$

At that point, the cloud server further decides the connection among an and b relying upon the consequence of Eq 2

$$e(E(x), \lambda^{p}) = e(g^{x} h^{r}, \lambda^{p}) \quad \text{(Eq 2)}$$

Where$\lambda \in G$, p istheprivatekey,andλpisthescatteringfeature that cannot be an individuality ofG.Tillnow,thecloudserverhasknowng^{x}h^{r}andλp.Toreje cttherandomvaluer,itthencalculatese(g^{x}h^{r},λp)bybilinear maps.Remindthataandbsignifysuccessivepositionsifan donlyiftheoutcomeofEq.(2)isequalto1,ase(g^{x}h^{r},λp)=e(g ^{0}h^{r},λp)=e(h^{r},λp)=e(h,λ)rp=e(h^{r}p,λ) =e(1,λ) = 1.Theideaofsuchastrategyderivesfromthedetailthat wecanrejectthepresenceoftherandomvalue for(hr)p=urpq=urn= 1 (modn). Be that as it may, the phrase ID technique is performed by the cloud worker, a client can't ship off the cloud worker legitimately. Along these lines, the client arbitrarily picks a component $\lambda \in G$ and sends λ p to the cloud worker. Since λ and p are both mystery, the cloud worker can't surmise p from λ p. As of now we rapidly look at the improvement of the phrase affirmation measure. First and foremost, at a raised level, we have to make sure about the keyword territory data, rather than the keyword zone relationship in the phrase search. This is in light of the fact that important the keyword zone relationship is certain to perform phrase affirmation.Second, the acknowledgment strategy can decide an arbitrary span for two numbers. As such, in the event that we need to know whether the span between two areas an and b is d, we can simply send g^{x}h^{r} cloud server, where r is an arbitrary number. Moreover, the code messages for similar d over various questions are unique. This property can keep the cloud server from inducing the stretch d, on the grounds that the cloud server can't have the foggiest idea about the genuine estimation of d regardless of whether it discovers that an and b fulfil a specific relationship.

Intuitively,given a view created by the test system, if the cloud server, who has a few sets of questioned phrases and trapdoors, can't recognize it from the view that claims, we can say that the proposed phrase search scheme is secure under the realized foundation model.

V. EXPERIMENTAL RESULTS

Despite the way that the proposed conspire is more successful than the flow phrase search plot. In order to evaluate our scrambled phrase search model, we collect the encoded index with document arrangement which has over 2GB plain substance data corpus. Since the search cycle of our structure takes only one round of correspondence and the speed of the association gear is uncertain, we kill the association overhead, slack and move time related with correspondence when we play out our speed tests. Plus, we furthermore discard external variables, for instance, early on index saving and set up an ideal occasion to decrease the effect of the IO errands. Hence, we simply figure the time it takes for the cloud worker to run search questions and get back with report identifiers.

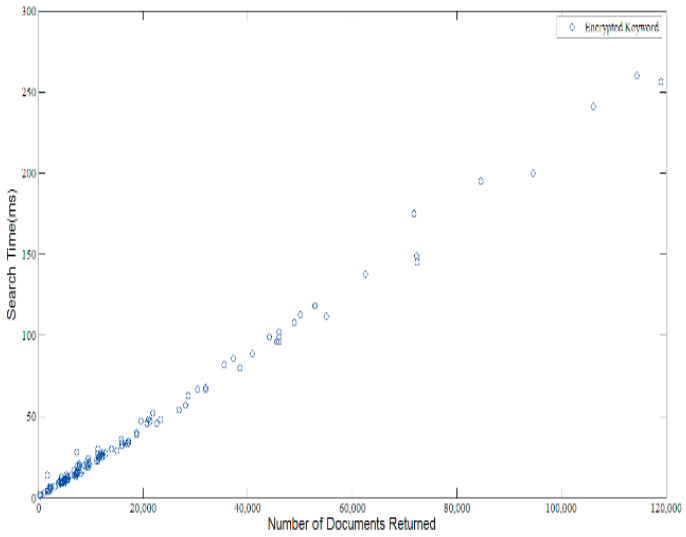

Chart 1: Search Time model

The search time of individual search inquiries fundamentally relies upon the quantity of documents containing the keywords in query and the quantity of keywords in the query.

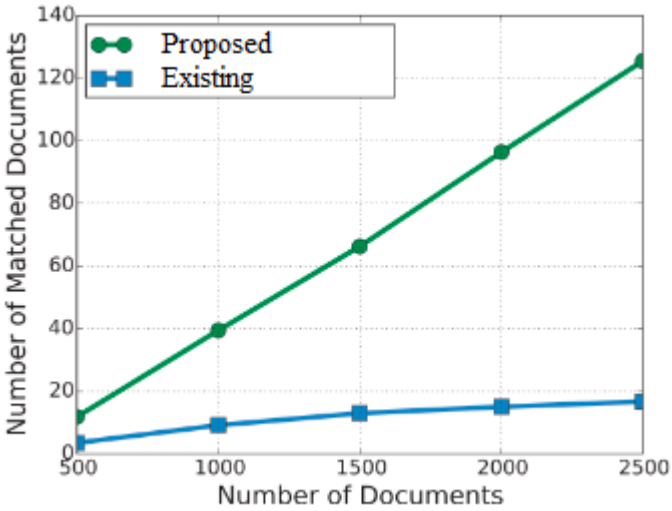

Chart 2: Matched Document

For the purpose of search, the procedure is search time and number of returned documents isn't as stable as searching a solitary keyword.

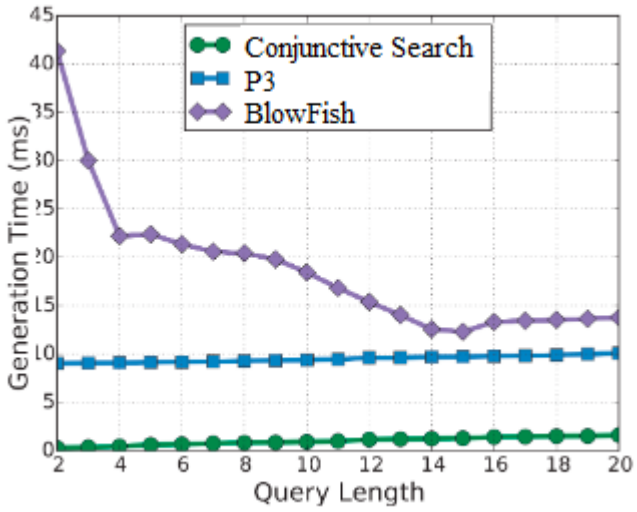

Chart 3: Generation time for query lengths

As shown, the proposed approach can accomplish critical reserve funds in cloud storage at a humble increment in client storage.

Discussion

Phrase searching can anyway fall flat, if inaccurate words are utilized or they show up out of order. For this situation, no outcomes will be found and unessential outcomes will be recovered. Phrase searching represented around 10% of questions in an investigation on the utilization of search engines on WebCrawler and Magellan. The inquiries are first contrasted against the rearranged index with discover which documents (and their comparing columns in the index) contain the keyword set. At that point these columns are dissected and the encrypted shortened keyword index and keyword areas are gotten back to the customer side server. As a phrase contains multi-keywords and the spots of keywords are determined, the amount of returned reports all things considered isn't actually searching single keyword. Since there are some etymological parts that influence the speed of searching a phrase. For example, if the phrase involves commonly ordinary keywords like "others", it requires longer exertion to reestablish the result than other phrase whose search result containing about comparative number of records. For such a phrases, the contained keywords that happen a great deal of times in the report variety, in this way the hour of telling whether the keywords are constant is any more.

VI. CONCLUSION

Phrase searching empowers clients to discover documents containing the full content that a client looks for. The more interesting the phrase is, the almost certain the right archive will show up in the profoundly positioned documents. Phrase searching is an exceptionally productive approach to find an ideal referred to message, all things considered definitely bound to accomplish the ideal outcome than playing out a keyword search. In this paper we applied a blowfish encryption scheme to give a safe search phrase. In cloud they were may data found, to locate a fast search of phrase, we may confront numerous battles, and furthermore security issue may raise because of open access of data. For giving those protections here blowfish is applied. This give high security more exactness which perceives the phrase subject to their keyword. It gives more noteworthy safety less key span and there is no convincing motivation to store any private key wherever. Blowfish algorithm is safer to take a gander at other balanced key computations, and make best result for less taking care of time and changes. To manufactures the key size of blowfish computation.

VII. REFERENCES

[1]. He Tuo and Ma Wenping, “An effective fuzzy keyword searchscheme in cloud computing,” inInternational Conference onIntelligent Networking and Collaborative Systems, 2013, pp.786–789.

[2]. A. Andrejev, D. Misev, P. Baumann and T. Risch, "Spatio-temporal gridded data processing on the semantic web", *Proceedings of the 2015 IEEE International Conference on Data Science and Data Intensive Systems*, pp. 38-45, Dec. 2015.

[3]. L. Zhu, X. Tang, M. Shen, X. Du, and M. Guizani. Privacy-preservingddos attack detection using cross-domain traffic in software defined net-works.IEEE Journal on Selected Areas in Communications, 36(3):628–643, March 2018

[4]. F. Gao, L. Zhu, M. Shen, K. Sharif, Z. Wan, and K. Ren.Ablockchain-based privacy-preserving payment mechanism for vehicle-to-grid networks.IEEE Network, pages 1–9, 2018.

[5]. C. Chen, X. Zhu, P. Shen, J. Hu, S. Guo, Z. Tari, and A. Y. Zomaya.An efficient privacy-preserving ranked keyword search method.TPDS,27(4):951–963, 2016

[6]. A. Anand, I. Mele, S. Bedathur, and K. Berberich.Phrasequeryoptimization on inverted indexes. InProc. of ACM CIKM, pages 1807–1810. ACM, 2014.

[7]. Liu, Z. Li, W. Guo, and C. Wu. Privacy-preserving multi-keywordranked search over encrypted big data. InInternational Conference onCyberspace Technology, 2016.

[8]. H. T. Poon and A. Miri. A low storage phrase search scheme basedon bloom filters for encrypted cloud services.In2015 IEEE 2ndInternational Conference on Cyber Security and Cloud Computing,pages 253–259, Nov 2015.

[9]. Patterson, Kathryn, Carolyn Watters, and Michael Shepherd. "Document Retrieval using Proximity-based Phrase Searching." In Proceedings of the 41st Annual Hawaii International Conference on System Sciences (HICSS 2008), pp. 137-137. IEEE, 2008.

[10]. Zittrower, Steven, and Cliff C. Zou. "Encrypted phrase searching in the cloud." In 2012 IEEE Global Communications Conference (GLOBECOM), pp. 764-770. IEEE, 2012.

[11]. Li, Mingchu, Wei Jia, Cheng Guo, Weifeng Sun, and Xing Tan. "LPSSE: lightweight phrase search with symmetric searchable encryption in cloud storage." In 2015 12th International Conference on Information Technology-New Generations, pp. 174-178. IEEE, 2015.

[12]. Shen, Meng, Baoli Ma, Liehuang Zhu, Xiaojiang Du, and Ke Xu. "Secure phrase search for intelligent processing of encrypted data in cloud-based IoT." IEEE Internet of Things Journal 6, no. 2 (2018): 1998-2008.

[13]. Poon, Hoi Ting, and Ali Miri. "An efficient conjunctive keyword and phrase search scheme for encrypted cloud storage systems." In 2015 IEEE 8th International Conference on Cloud Computing, pp. 508-515. IEEE, 2015.

[14]. MilindMathur, "Comparison between DES, 3DES, RC2, RC6, BLOWFISH and AES”, National Informatics Center Network NICNET, Vol. 1, No.3, pp. 143-148, 2013.

[15]. C. Nandini and B. Shylaja, *“Efficient Cryptographic key Generation from Fingerprint using Symmetric Hash Functions”*, International Journal of Research and Reviews in Computer Science (IJRRCS), Vol. 2, No. 4, August 2011.

[16]. M. Shen, B. Ma, L. Zhu, R. Mijumbi, X. Du, and J. Hu. Cloud-based approximate constrained shortest distance queries over encrypted graphswith privacyprotection.IEEE Transactions on Information Forensics and Security, 13(4):940–953, April 2018.

[17]. Jasmeet Singh, Harmandeep Singh, "Design and Development of a Rapid AES based Encryption Framework",International Journal of Engineering Research & Technology(IJERT),Vol.3,No.10,ISSN:2278-0181,2014.

[18]. M. Shen, M. Wei, L. Zhu, and M. Wang. Classification of encrypted traffic with second-order markov chains and application attribute bigrams. IEEE Transactions on Information Forensics and Security, 12(8):1830–1843, Aug 2017.