ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 2, March 2016

Privacy On Encrypted Cloud Data Multi – Keyword Ranked Search

K.Balasubramani.,M.E., Assistant Professor.,
Dept.of. CSE Tagore Institute of Engineering and Technology,
Salem,India.
kbsbala@gmail.com

M.Venkatesan.,M.E., Student Dept.of.CSE
Tagore Institute of Engineering and Technology,
Salem,India.
mvenkates.sri@gmail.com

Abstract—With the advent of cloud computing, data owners are motivated to outsource their complex data management systems from local sites to the commercial public cloud for great flexibility and economic savings. But for protecting data privacy, sensitive data have to be encrypted before outsourcing, which obsoletes traditional data utilization based on plaintext keyword search. Thus, enabling an encrypted cloud data search service is of paramount importance. Considering the large number of data users and documents in the cloud, it is necessary to allow multiple keywords in the search request and return documents in the order of their relevance to these keywords. Related works on searchable encryption focus on single keyword search or Boolean keyword search, and rarely sort the search results. In this paper, for the first time, we define and solve the challenging problem of privacy-preserving multi-keyword ranked search over encrypted data in cloud computing (MRSE). We establish a set of strict privacy requirements for such a secure cloud data utilization system. Among various multi-keyword semantics, we choose the efficient similarity measure of "coordinate matching," i.e., as many matches as possible, to capture the relevance of data documents to the search query. We further use "inner product similarity" to quantitatively evaluate such similarity measure. We first propose a basic idea for the MRSE based on secure inner product computation, and then give two significantly improved MRSE schemes to achieve various stringent privacy requirements in two different threat models. To improve search experience of the data search service, we further extend these two schemes to support more search semantics. Thorough analysis investigating privacy and efficiency guarantees of proposed schemes is given. Experiments on the real-world data set further show proposed schemes indeed introduce low overhead on computation and communication.

Index Terms—Cloud computing, searchable encryption, privacy-preserving, keyword search, ranked search

1 INTRODUCTION

Cloud computing is the long dreamed vision of computing as a utility, where cloud customers can remotely store their data into the cloud so as to enjoy the on-demand high-quality applications and services from a shared pool of configurable computing resources Its great flexibility and economic savings are motivating both individuals and enterprises to outsource their local complex data management system into the cloud. To protect data privacy and combat unsolicited accesses in the cloud and beyond, sensitive data, for example, e-mails, personal health records, photo albums, tax documents, financial transactions, and so on, may have to be encrypted by data owners before outsourcing to the commercial public cloud obsoletes the traditional data service based on plaintext keyword search. The trivial solution of downloading all the data and decrypting locally is clearly impractical, due to the huge amount of bandwidth cost in cloud scale systems. Moreover, aside from eliminating the local storage management, storing data into the cloud serves no purpose unless they can be easily searched and utilized. Thus, exploring privacypreserving and effective search service over encrypted cloud data is of paramount importance. Considering the potentially large number of on-demand data users and huge amount of outsourced data documents in the cloud, this problem is particularly challenging as it is extremely difficult to meet also the requirements of performance, system usability, and scalability.

On the one hand, to meet the effective data retrieval need, the large amount of documents demand the cloud server to perform result relevance ranking, instead of returning undifferentiated results. Such ranked search system enables data users to find the most relevant information quickly, rather than burdensomely sorting through every match in the content collection. Ranked search can also elegantly eliminate unnecessary network traffic by sending back only the most relevant data, which is highly desirable in the "pay-as-you-use" cloud paradigm.For privacy protection, such ranking operation,however, should not leak any keyword related information.On the other hand, to improve the search result accuracy as well as to enhance the user searching experience, it is also necessary for such ranking system to support multiple keywords search, as single keyword search often yields far too coarse results. As a common practice indicated bytoday's web search engines (e.g., Google search), data users may tend to provide a set of keywords instead of only one as the indicator of their search interest to retrieve the most relevant data. And each keyword in the search request is able to help narrow down the search result further."Coordinate matching" as many matches as possible, is an efficient similarity measure among such multi-keyword semantics to refine the result relevance,and has been widely used in the plaintext information retrieval (IR) community. However, how to apply it in the encrypted cloud data search system remains a very challenging task because of inherent security and privacy obstacles, including various strict requirements like the data privacy, the index privacy, the keyword privacy.

In two threat models with increased attack capabilities. Our Contributions are summarized as follows:

1. For the first time, we explore the problem of multikeyword and establish a set of strict privacy
2. Requirements for such a secure cloud data utilization system.

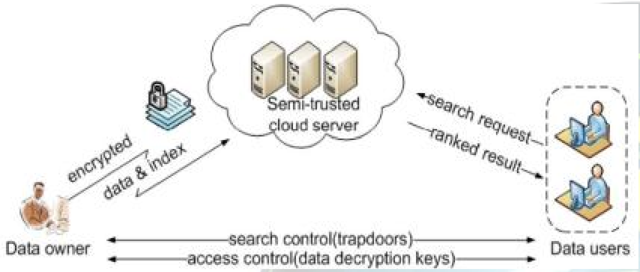

fig.1.architecture of the search over encrypted cloud data.

3. We propose two MRSE schemes based on the similarity measure of "coordinate matching" while meeting different privacy requirements in two different threat models.
4. We investigate some further enhancements of our ranked search mechanism to support more search semantics and dynamic data operations.
5. Thorough analysis investigating privacy and efficiency guarantees of the proposed schemes is given and experiments on the real-world data set further show the proposed schemes indeed introduce low overhead on computation and communication.

This version also studies the support of data/index dynamics in the mechanism design. Moreover, we improve the experimental works by adding the analysis and evaluation of two new schemes. In addition to these improvements, we add more analysis on secure inner product and the privacy part. The remainder of this paper is organized. In we introduce the system model, the threat model, our design goals, and the preliminary. The describes the MRSE framework and privacy requirements. Which describes the proposed schemes. Presents simulation results. We discuss related work on both single and Boolean keyword searchable encryption in

2 LITERATURE SURVEY

In the literature, searchable encryption [2], [3], [4], [5], 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 cryptoprimitives 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 [2], [3], [4], 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.

In this paper, for the first time, we define and solve the problem of multi-keyword ranked search over encrypted cloud data (MRSE) while preserving strict systemwise privacy in the cloud computing paradigm. Among various multi-keyword semantics, we choose the efficient similarity measure of "coordinate matching," i.e., as many matches as possible, to capture the relevance of data documents to the search query. Specifically, we use "inner product similarity", i.e., the number of query keywords appearing in a document, to quantitatively evaluate such similarity measure of that document to the search query. During the index construction, each document is associated with a binary vector as a subindex where each bit represents whether corresponding keyword is contained in the document. The search query is also described as a binary vector where each bit means whether corresponding keyword appears in this search request, so the similarity could be exactly measured by the inner product of the query vector with the data vector. However, directly outsourcing the data vector or the query vector will violate the index privacy or the search privacy. To meet the challenge of supporting such multikeyword semantic without privacy breaches, we propose a basic idea for the MRSE using secure inner product computation, which is adapted from a secure k-nearest neighbor (kNN) technique , and then give two significantly improved MRSE schemes in a step-by-step manner to achieve various stringent privacy requirements.

3 PROBLEM FORMULATIONS

3.1 System Model

Considering a cloud data hosting service involving three different entities, as illustrated in Fig. 1: the data owner, the data user, and the cloud server. The data owner has a collection of data documents F to be outsourced to the cloud server in the encrypted form C. To enable the searching capability over C for effective data utilization,the data owner, before outsourcing, will first build an encrypted searchable index I from F, and then outsource both the index I and the encrypted document collection C to the cloud server. To search the document collection for t given keywords, an authorized user acquires a corresponding trapdoor T through search control mechanisms, for example, broadcast encryption [4]. Upon receiving T from a data user, the cloud server is responsible to search the index I and return the corresponding set of encrypted documents. To improve the document retrieval accuracy, the search result should be ranked by the cloud server according to some ranking criteria (e.g., coordinate matching, as will be introduced shortly). Moreover, to reduce the communication cost, the data user may send an optional number k along with the trapdoor T so that the cloud server only sends back top-k documents that are most relevant to the search query. Finally, the access control mechanism is employed to manage decryption capabilities given to users and the data collection can be updated in terms of inserting new documents, updating existing documents, and deleting existing documents

3.2 Threat Model

The cloud server is considered as "honest-but-curious" in our model, which is consistent with related works on cloud security [12], Specifically, the cloud server acts in an "honest" fashion and correctly follows the designated protocol specification. Christo Ananth et al. [10] proposed a system which can achieve a higher throughput and higher energy efficiency. The S-BOX is designed by using Advanced Encryption Standard (AES). The AES is a symmetric key standard for encryption and decryption of blocks of data. In encryption, the AES accepts a plaintext input, which is limited to 128 bits, and a key that can be specified to be 128 bits to generate the Cipher text. In decryption, the cipher text is converted to original one. By using this AES technique the original text is highly secured and the information is not broken by the intruder. From that, the design of S-BOX is used to protect the message and also achieve a high throughput, high energy efficiency and occupy less area.As an instance of possible attacks in this case, the cloud server could use the known trapdoor information combined with document/keyword frequency [11] to deduce/identify certain keywords in the query.

3.3 Design Goals

To enable ranked search for effective utilization of outsourced cloud data under the aforementioned model, our system design should simultaneously achieve security and performance guarantees as follows..

1. Multi-keyword ranked search. To design search schemes which allow multi-keyword query and provide result similarity ranking for effective data retrieval, instead of returning undifferentiated results..
2. Privacy-preserving. To prevent the cloud server from learning additional information from the data set and the index, and to meet privacy requirements

3. Efficiency. Above goals on functionality and privacy should be achieved with low communication and computation overhead.

3.4 Notations

1. F – the plaintext document collection,denoted as a set of m data documents F=(F_1,F_2,….F_m).

2. C – the encrypted documents collection stored in the cloud server, denoted as C=(C_1,C_2….C_m)
3. W – the dictionary, i.e., the keyword set consisting of n keyword, denoted as W=(W_1, W_2,….W_n)

4. I – the searchable index associated with C,denoted as (I_1,I_2,….I_m) where each subindex I_i is built for F_i

5. $\hat{W}$ – the subset of W, representing the keywords in a search request, denoted as $\hat{W}$=(W_{j1},W_{j2},…W_{jt})

6. T $_{\hat{W}}$ – the trapdoor for the search request $\hat{W}$.

7. F $_{\hat{W}}$ – the ranked id list of all documents according to their relevance to $\hat{W}$.

3.5 Preliminary on Coordinate Matching

As a hybrid of conjunctive search and disjunctive search, "coordinate matching" is an intermediate similarity measure which uses the number of query keywords appearing in the document to quantify the relevance of that document to the query. When users know the exact subset of the data set to be retrieved, Boolean queries perform well with the precise search requirement specified by the user. In cloud computing, however, this is not the practical case, given the huge amount of outsourced data. Therefore, it is more flexible for users to specify a list of keywords indicating their interest and retrieve the most relevant documents with a rank order.

4 FRAMEWORK AND PRIVACY REQUIREMENTS FOR MRSE

we define the framework of multi-keyword ranked search over encrypted cloud data (MRSE) and establish various strict system wise privacy requirements for such a secure cloud data utilization system.

4.1 MRSE Framework

For easy presentation, operations on the data documents are not shown in the framework since the data owner could easily employ the traditional symmetric key cryptography to encrypt and then outsource data. With focus on the index and query, the MRSE system consists of four algorithms as follows:

1. **Setup** the data owner generates a n+2 – bit vector as X and two(n+2) x (n+2) invertible matrices {M1,M2}. The secret key SK is the form of a 3-tuple as{X,M1,M2}.

2. **BuildIndex**($A^{'}$,SK) The data owner extracts a term – document matrix $A^{'}$,Following, we multiply these three matrices to get the result matrix **A.**Taking privacy into consideration, it is necessary that the matrix **A** is encrypted before outsourcing. After applying dimension-extending, the original **A[j]** is extended to (n+2)-dimensions, instead of n. namely the (n+1)-th entry in **A[j]** is set to a random number $\sum_j$.and the (n+2) –th entry in A[j] is set to 1 during the dimension extending.Finally,A[j] can be represented as $((A[j])^T, \sum_j.1)^T$.The subindex $I_j=\{M_1^T,A'[j], M_2^T,A^{''}[j]\}$ build.

3. **Trapdoor(Ŵ)** With t keywords of interest in Ŵ input one binary vector Q is generated. The (n+1) – th entry in Q is set to a random number 1, and scaled by a random number r≠0, and the (n+2)-th in Q is set to another random number t.Q can ber represented as $(rQ^T,r,t)^T$.The trapdoor $T_{\hat{W}}$ is generated as$\{M_1^{-1},Q^{'},M_2^{-1},Q^{''}\}$.

4. **Query($T_{\hat{W}}$,I,I)**The inner product of $I_j$and $T_{\hat{W}}$ is calculated by the cloud server. After sorting all scores, the cloud server returns the top-l ranked id list to the data.

4.2 Privacy Requirements for MRSE

Keyword Privacy: hide what the user is searching, i.e., the keywords indicated by the corresponding trapdoor.

Trapdoor Privacy (Unlink ability): the cloud server should not able to link the relationship• of any given trapdoor, e.g., to determine whether the two • trapdoors are formed by the same search request.

Access Pattern Privacy: the sequence of search results the proposed scheme is not designed to protect access• pattern for the efficiency concerns

5. PROPOSED SYSTEM

For our system, we choose the principle of coordinate matching, to identify the similarity between search query and data documents.Specially,we use inner data correspondence, i.e., the number of query keywords appearing in a document, to evaluate the similarity of that document to the search query in coordinate matching principle. Each document is linked with a binary vector as a sub index where each bit represents whether corresponding keyword is contained in the document.[1] The search query is also described as a binary vector where each bit means whether corresponding keyword appears in this search request, so the similarity could be exactly measured by inner product of query vector with data vector. However, directly outsourcing data vector or query vector will violate index privacy or search privacy. To meet the challenge of supporting such multi-keyword semantic without privacy breaches, we propose a basic SMS scheme using secure inner product computation, which is adapted from a secure k-nearest neighbour (kNN) technique, and then improve it step by step to achieve various privacy requirements in two levels of threat models.

1. Showing the problem of Secured Multi-keyword search over encrypted cloud data .

2. Propose two schemes following the principle of coordinate matching and inner product similarity.

Algorithms Used

RSA Algorithm

This algorithm is used to encrypt n decrypt file contents. It is an asymmetric algorithm. The RSA algorithm involves three steps. Key generation, encryption and decryption.

Key generation

RSA involves a public key and a private key. The public key can be known to everyone and is used for encrypting messages. Messages encrypted with the public key can only be decrypted using the private key. The keys for the RSA algorithm are generated the following way.

1. Choose two distinct prime numbers a and b.

2. Compute n=ab n is used the modulus for both the public and private keys.
3. Compute ¢(n)=(a-1)(b-1),where ¢ is Euler's totient function.
4. Choose an integer e such that 1<e<¢(n)and greatest common divisor of(e, ¢(n))=1;i.e., e and ¢(n) are co prime.
   e is released as the public key exponent. Having a short bit-length.

Encryption

Alice transmit her public key(n,e) to bob and keeps the private
key secret. Bob then wishes to send message M to alice.He first turns M into an integer m, such that O<m <n by using an agreed-upon reversible the cipher text C corresponding to

$$C=m^{e} \ (\text{mod } n)$$

This can be done quickly using the method of exponentiation by squaring. Bob then transmits C to Alice. Note that at least nine values of m could yield a cipher text c equal to m, but this is very unlikely to occur in practice.

Decryption

Alice can recover m from c by using her private key exponent d via computing.

$$m=C^{d} \ (\text{mod } n)$$

Given m, she can recover the original message M by reversing the padding scheme.(In practice, there are more efficient methods of calculating C^{d} using the pre computed values below.)

K-Nearest Neighbour

K-nearest neighbor search identifies the top k nearest neighbors to the query. This technique is commonly used in predictive analytics to estimate or classify a point based on the consensus of its neighbors. K-nearest neighbor graphs are graphs in which every point is connected to its k nearest neighbors.

The basic idea of our new algorithm. The value of d_{max} is decreased keeping step with the ongoing exact evaluation of the object similarity distance for the candidates. At the end of the step by step refinement, d_{max} reaches the optimal query range Ed and prevents the method from producing more candidates than necessary thus fulfilling the r-optimality criterion.

Nearest Neighbor search(q,k)//optimal algorithm

```
1. Initialize ranking = index.increm-ranking(F(q).df)
2. Initialize result=new sorted-list(key,object)
3. Initilized  $d_{max}$=w
4. While o=ranking.getnext and d,(o,q) I d,,,do
5. If do @,s>s $d_{max}$ then result.insert(d,(o,q),o)
6. If result.length 2k then $d_{max=}$result[k].key
7. Remove all entries from result where key> $d_{max}$
8. End while report all entries from result where key I $d_{max}$
```

Module Description

1. **Encryption Module**
   This module is used to help the server to encrypt the document using RSA Algorithm and to convert the encrypted document to the Zip file with activation code and then activation code send to the user for download.
2. **Multi-keyword Module**
   This module is used to help the user to get the accurate result based on the multiple keyword concepts. The users can enter the multiple words query, the server is going to split that query into a single word after search that word file in our database. Finally, display the matched word list from the database and the user gets the file from that list.
3. **File upload Module**
   This module is used to help the server to view details and upload files with the security. Admin uses the log key to the login time. Before the admin logout, change the log key. The admin can change the password after the login and view the user downloading details and the counting of file request details on flowchart. The admin can upload the file after the conversion of the Zip file format.
4. **Data user module**
   This module includes the user registration login details .
5. **Data Owner Module**
   This module helps the owner to register them details And also include login details.

6. PERFORMANCE AND SECURITY ANALYSIS

we show a thorough experimental evaluation of the proposed technique on a real dataset: the *MED* dataset[17].The whole experiment is implemented by C++ language on a computer with Core 2.83GHz Processor, on Windows 7 system. For the proposed scheme, we will reduce to separate dimensions. The performance of our method is compared with the original MRSE scheme.

6.1 Efficiency

The proposed scheme is depicted in details in previous section, except the Key Gen algorithm. In our scheme, we adopt Gauss-Jordan to compute the inverse matrix. The time of generating key is decided by the scale of the matrix. Besides, the proposed scheme that processed by SVD algorithm will consume time. Other algorithms, such as index construction, trapdoor generation, query, which is put forward by us, are consistent with the original MRSE in time-consuming.

6.2 Measure

we still use the measure of traditional information retrieval. Before the introduction of the F-measures concept, we will firstly give the brief of the precision and recall. Precision is the fraction of retrieved instances that are relevant, while recall is the fraction of relevant instances that are retrieved. Both precision and recall are therefore based on an understanding and measure of relevance[18]. F-measure that combines precision and recall is the harmonic mean of precision and recall[19]. Here, we adopt F-measure to weigh the result of our experiments.

$$F= \frac{2.precision.recall}{Precision+ recall}$$

6.3 Performance Analysis

For a clear comparison, our proposed scheme attains score higher than the original *MRSE* in F-measure. Since the original scheme employs exact match, it must miss some similar words which is similar with the keywords. However, our scheme can make up for this disadvantage, and retrieve the most relevant files. Figure2 shows that our method achieves remarkable result.

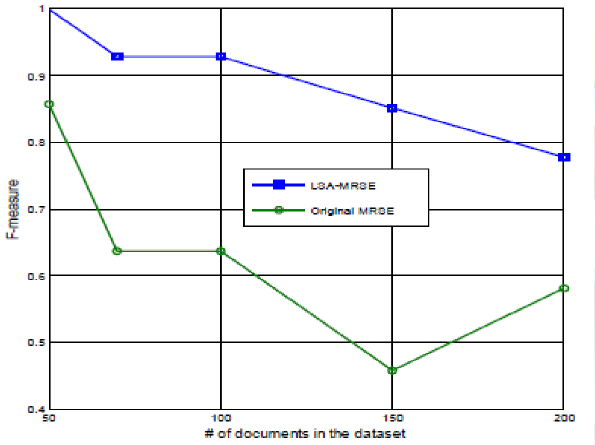

Fig. 2. Comparison of two schemes.

6.4 Security Analysis

We analyze our proposed scheme according to the predefined privacy requirements in design goals

1. Index Confidentiality .In our proposed scheme and are obfuscated vectors, which means the cloud server cannot infer the original data vector and the query vector without the secret key SK. As is proven in [14], the cloud server cannot deduce TF values from the result relevance scores. In other word, the index confidentiality is protected.
2. Trapdoor Unlink ability. The trapdoor of query vector is generated from a random splitting operation, which means the same search requests are transformed into different query trapdoors. And thus, the query Unlink ability well preserved.
3. Keyword Privacy. In the known background scheme, the cloud server is supposed to have more knowledge, such as the distribution TF values of keywords in the dataset. The cloud server is able to identify keywords by analyzing these specific distributions. In our scheme, the TF distributions of keywords will be leaked directly when there is only one query keyword. Thus, our proposed scheme is designed to obscure the TF distributions of keywords with the dummy values. That is to say, the keyword privacy is protected.

7. CONCLUSION

The first time we define and solve the problem of multi-keyword ranked search over encrypted cloud data, and establish a variety of privacy requirements. Among various multi-keyword semantics, we choose the efficient similarity measure of “coordinate matching,” i.e., as many matches as possible, to effectively capture the relevance of outsourced documents to the query keywords, and use “inner product similarity” to quantitatively evaluate such similarity measure. For meeting the challenge of supporting multi-keyword semantic without privacy breaches, we propose a basic idea of MRSE using secure inner product computation. Then, we give two improved MRSE schemes to achieve various stringent privacy requirements in two different threat models. We also investigate some further enhancements of our ranked search mechanism, including supporting more search semantics, i.e., TF _IDF, and dynamic data operations. Thorough analysis investigating privacy and efficiency guarantees of proposed schemes is given, and experiments on the real-world data set show our proposed schemes introduce low overhead on both computation and communication. In our future work, we will explore checking the integrity of the rank order in the search result assuming the cloud server is untrusted.

REFERENCES

[1] Cao.N, Yu.S, Yang.Z, Lou.W, and Y. Hou, ‘LT Codes-Based Secure and Reliable Cloud Storage Service’,Proc. IEEE INFOCOM,pp. 693-701, 2012.

[2] Song.D, Wagner.D, and Perrig.D, ‘Practical Techniques for Searches on Encrypted Data;, Proc. IEEE Symp. Security and Privacy, 2000.

[3] Chang.Y.C and Mitzenmacher.M,’Privacy Preserving Keyword Searches on Remote Encrypted Data’, Proc. Third Int’l Conf.Applied Cryptography and Network Security, 2005.

[4] Curtmola.R, Garay.J.A, Kamara.S, and Ostrovsky.S, 'Searchable Symmetric Encryption: Improved Definitions and Efficient Constructions,"Proc. 13th ACM Conf. Computer and Comm. Security (CCS '06), 2006.

[5] Bellare.M, Boldyreva.A, and ONeill.A, 'Deterministic and Efficiently Searchable Encryption,' Proc. 27th Ann. Int'l Cryptology Conf. Advances in Cryptology (CRYPTO '07), 2007.

[6] Vaquero.L.M, Rodero-Merino.L, Caceres.J, and Lindner.M, 'A Break in the Clouds: Towards a Cloud Definition', ACM SIGCOMM Comput. Commun. Rev., vol. 39, no. 1, pp. 50-55, 2009.

[7] Bellare.M, Boldyreva.A, and ONeill.A, "Deterministic and Efficiently searchable Encryption," Proc. 27th Ann. Int'l Cryptology Conf. Advances in Cryptology (CRYPTO '07), 2007.

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

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

[10] Christo Ananth, H. Anusuya Baby, "S-Box using AES Technique", International Journal of Engineering Research & Technology (IJERT), Vol. 3 Issue 3, March – 2014, pp 285-290

[11] Cao.N, Yang.Z, Wang.C, Ren.K, and Lou.W, "Privacypreserving Query over Encrypted Graph-Structured Data in Cloud Computing," Proc. Distributed Computing Systems (ICDCS),pp. 393-402, June, 2011.

[12] Wang.C, Wang.Q, Ren.K, and Lou.W, "Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing," Proc. IEEE INFOCOM, 2010.

[13] L.Ji, Wang.Q, Wang.C, Cao.N, Ren.K, and Lou.W, "Fuzzy Keyword Search Over Encrypted Data in Cloud Computing,"Proc. IEEE INFOCOM, Mar. 2010.

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

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

[16] Li.M, Yu.S, Cao.N, and Lou.W, "Authorized Private Keyword Search over Encrypted Data in Cloud Computing," Proc. 31st Int'l Conf. Distributed Computing Systems (ICDCS '10), pp. 383-392, June 201

[17] Boneh.D,Crescenzo.G.D, Ostrovsky,.R and Persiano.G, "Public Key Encryption with Keyword Search," Proc. Int'l Conf. Theory and Applications of Cryptographic Techniques (EUROCRYPT), 2004