ISSN No. 0976-5697

Volume 4, No. 3, March 2013 (Special Issue)

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Privacy Preserving Keyword Search for Encrypted Cloud Storage Data

Ms. Pooja D. Shah
P.G. Student, Dept. of Information Technology
Shantilal Shah College of Engineering
Bhavnagar, India
poosrk12@gmail.com

Mr. Gopal Pandey
In-charge head of the Information Technology
Department Sir Bhavsinhji Polytechnic Institute
Bhavnagar, India
mr.gopal.pandey@gmail.com

Abstract: Cloud storage services allow the users to outsource their data in the Cloud Storage Servers and retrieve them whenever and wherever required. This avoids the cost of building and maintaining their own data store. But the users need to provide privacy for the data and also needs to be able to search it without losing privacy. The users always search their documents through keyword in plaintext, which may leak privacy of users in Cloud Storage Environment. So allowing a Cloud Service Provider (CSP), whose purpose is mainly for making a profit, to take the custody of sensitive data, raises underlying security and privacy issues. To keep user data confidential against an untrusted CSP, a natural way is to apply cryptographic approaches, by disclosing the data decryption key only to authorized users. In this paper we propose an efficient, secure and privacy preserving keyword search scheme which supports multiple users with low computation cost and flexible key management and it is proved to be secure and flexible.

Keywords: Cloud Storage Services; Security; Searchable Encryption; Partial decipherment; Privacy Preserving.

I. INTRODAUCTION

Cloud Computing is online method to distributing the resources on demand of users. And manages and schedules the computing resources through network. It constitutes a large computing resources pool which can provide service to users on their demand. Cloud Storage Services Provides the large space of data storage resources and it stores the data on the remote servers based on the Cloud Computing. It uses the normal hard drive for storing the all kinds of the data on the remote servers. It includes database-like services and network attached storage. It is often billed for usage per gigabyte per month. For example, the Drop box simple storage service that provides 2GB free storage space and the after charges starting from $9.99 for 100, 200 and 500 GB per month.

The main purpose of using these services is outsourcing the data. Using this we can access the data from anywhere in the word by any devices that is connected to the network. So it provides the biggest flexibility. For Example the user can store the file or some documents on the Cloud Storage. Now after some time it want to retrieve this then it uses some keywords that define in that file. So user sends keyword to CSP. The CSP checks and finds this keyword and sends appropriate output to the user. So in this process the keyword must be secure from unauthorized user. For this task it decrypts the all documents. So it contains more memory power and large computation power. Also if it sends key as a plaintext than two main attacks is performed on this communication e.g. external attacks initiated by unauthorized outsiders and internal attacks initiated by untrustworthy CSPs. In some cases, we cannot fully trust a CSP, but still need its services. So privacy must be needed on searching keyword. Without proper protection for the user searching keyword privacy, an attacker may know the user's private interests and querying patterns. For above example if any unauthorized person knows user keyword than that person sends the request for searching this keyword specific document as a legal request. Than easily it gets the same results and shows the user private data.

To avoid this problem we can use the some natural approached like searchable encryption scheme. Now Qin liu introduced an privacy preserving keyword search scheme in Cloud Computing. That uses this searchable encryption scheme. In this approach we can use encrypted search keyword for finding the document. Now in this paper, we propose an efficient encrypted keyword search scheme suitable for Cloud Storage. It has the following advantages:

a. It supports keyword search in encrypted form. The Cloud Server could determine which all documents contain the specified keyword without knowing anything about the contents of document or the keyword searched.
b. The service provider will participate in the partial decipherment of the cipher text, thus reducing the computational overhead of the user.
c. Same keywords are encrypted to different cipher text for different documents thus reducing redundancy and avoiding the chance of statistical attack on keyword cipher text. So for that every time random key is generated.

II. RELATED WORK

It is an important question is that how to Cloud Service Provider to efficiently search the keyword in encrypted form on encrypted files and providing user data privacy at

CONFERENCE PAPER
II International Conference on
"Advance Computing and Creating Entrepreneurs (ACCE2013)"
On 19-20 Feb 2013
Organized by
2nd SIG-WNs, Div IV & Udaipur Chapter , CSI , IEEE Computer Society Chapter India Council ,
IEEE Student Chapter Geetanjali Institute of Technical Studies, Udaipur, Rajasthan, India

the same time. The one approach is a public key encryption with keyword search (PEKS) which supports encrypted keyword search. Here the document is encrypted with any public key encryption algorithm and the user needs to decrypt it completely by him. So it will use too much CPU and memory power of the client if documents are decrypted frequently and loose the critical value of Cloud Computing. Qin liu introduced an efficient privacy preserving keyword search which allows the service provider to participate in partial decipherment of the searched documents thus reducing the computational overhead of the user. In this scheme if the keyword encryption uses public key of all share users then that is used for multi user also. In multi user approach, the trapdoor is made as the keyword query with the partial of every computed public key. When the number of users increases, it has low efficiency and it is not so efficient for the actual application of Cloud. For this proposed a scheme shared and searchable encrypted data for untrusted servers. It supports multiple users. In this scheme encryption is not based on public key. In this scheme service provider performs partial decryption. But it can't resist the statistical attack on keywords. Here same keyword is encrypted to same cipher texts only for different documents.

III. BASIC SYSTEM MODEL

In this secure and privacy preserving search keyword on encrypted data approach contains three participants are there:

A. User :

Who are the authorized person that stores the file, update the file, and operate some functions like encryption decryption.

B. Key Server:

It is a trusted server which stores all the keys used for encrypting the document along with the signature of file names. The key server provides the key for decrypting a file to the user after verifying the signature of file name.

C. CSP:

The data centre who provides the storage service. The users stored the cipher text of their file, the cipher text of the metadata of the files and the cipher text of the keyword into the Cloud Storage Server, so that the server can know nothing about the information in the files and keywords.
The basic mode of this approach is described in following Figure 1.

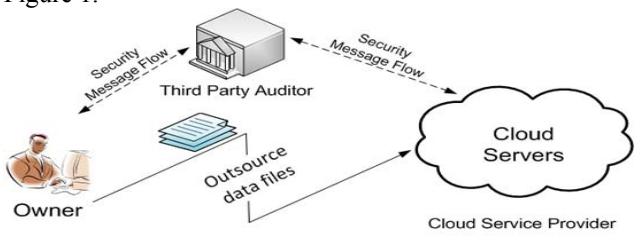

Figure 1 Secure Cloud Storage Basic System Model

IV. VARIOUS SCHEMAS

There are many different schemas are available that are described below.

A. Public Key Encryption With Keyword Searching(PEKS):

It is an asymmetric searchable encryption scheme, where encryption is done using a public key system. This was designed for the purpose of intelligent email routing. In this scheme, when a user requests a particular keyword, the server should retrieve the files or mails, which are in the server, but the server, should not know anything about the mail or the keyword. The four algorithms that are in this technique are given below:

a. Key Generation(K):

Takes a security parameter K and generates a public or private key pair Apub, Apriv.

b. PEKS(Apub, W):

For a public key Apub and a word W, produces a searchable encryption of W

c. Trapdoor (Apriv ,W):

With A's private key and a word W, a trapdoor TW is produced.

d. Test (Apub,S,TW)

With the public key of A, a searchable encryption S=PEKS(Apub, W□) and the trapdoor TW= Trapdoor(Apriv,W) then outputs =yes, if W=W otherwise no.
This PEKS scheme can easily applied when user stores its document on the CSP. In the following figure this applied by following way. 1) If Alice and Bob are the same User 'U' want to store its document on the CSP then first it runs the Key generation algorithm to generate the private and public key. 2) Now it uses the searchable encryption algorithm or PEKS for encrypting the document and respectively keywords. 3) Now 'U' wants to retrieve this document that containing the keyword W then runs the Trapdoor algorithm to compute Trapdoor for W and sends to CSP. 4) After receiving trapdoor the CSP uses the Test function for finding documents that contains same keyword. The working process of this is described in Figure 2.

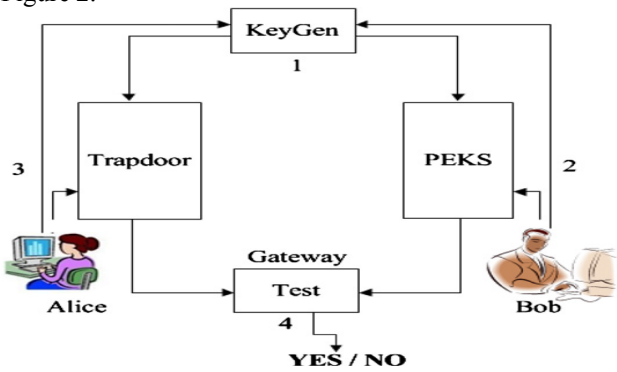

Figure 2 The working process of the PEKS scheme

CONFERENCE PAPER
II International Conference on
"Advance Computing and Creating Entrepreneurs (ACCE2013)"
On 19-20 Feb 2013
Organized by
2nd SIG-WNs, Div IV & Udaipur Chapter , CSI , IEEE Computer Society Chapter India Council ,
IEEE Student Chapter Geetanjali Institute of Technical Studies, Udaipur, Rajasthan, India

PKES also implies Identity Based Encryption it is proven to be semantically secured against an adaptive chosen keyword attack. But it has many issues. Like

a. The CSP finding the resulting document as an output to user but it is in the encrypted format. So user decrypts it before using this. So user memory power is used. For this reason every user must be need proper suitable CPU that has require configuration.
b. It is not used when multiple keywords are used for searching. These two issues are solved by the following advance schemas.

B. Efficient and Privacy Preserving Keyword Search(EPPKS):

When the encryption is not searchable the service provider will not be able to know which files are containing the keywords that are requested by the user. In this situation, the service provider will return all the encrypted files. If the user is using a thin client with a limited bandwidth, it will not be able to handle such situations, because of its limited bandwidth and memory.

So this schema used the partial decipherment, which will reduce the client's computational overhead, and enable the service provider to search through the encrypted files for the requested keywords in order to protect the user data privacy. It based on the BDH assumption; the CSP cannot know file contents and keywords. Seven randomized polynomial time algorithms are used here, they consists of the following:

a. Key Generation:

It takes large security parameter K1 and generates the public key pair for user kwon as (Upub, Upriv). Now also take another large security parameter K2 and generate another public key pair for CSP known as (Spub, Spriv).

b. Email Encryption (EMBEnc):

It uses the public key encryption algorithm.It takes two key Upub, Spub and message M as a input and encrypt the message that known as Cm. So We write EMBEnc (Upub; S pub;m) = Cm.

c. Keyword Encryption (KWEnc):

It is also performs the public key encryption algorithm. It takes the public key Upub and keyword Wi Є W (i Є Z+) as a input and produces cipher text CWi's Є CW. So we can write KWEnc(Upub;Wi) = CWi .

d. Trapdoor Computing (TCompute):

It takes the Upriv key and keyword like Wj Є W (j Є Z+) as a input and generate the Wj's trapdoor known as TWj. So we can write TCompute(Upriv, Wj) = TWj.

e. Testing:

Now it uses the CWi, TWj and Upub as a input and checks the finding keyword. It returns 1 if CWi=TWj otherwise it returns 0.

f. Decryption:

It takes Upub, Spriv, and Cm as an input and generates the intermediate result Cp.

g. Recovery:

It uses the Upriv to decrypt the Cp. So it takes Upriv and Cp as an input and generates the final output plain text M.

If any user that want to stores its email on CSP. That time if it uses this schema then first key generation party generate two pairs of the public key that known as (Upriv,Upub) and (Spriv, Spub). The email is now encrypted with Upub and Spub key before store on the CSP. And keywords of this document also encrypted with keyword encryption algorithm.

When user accesses his email then it sends the request to CSP with trapdoor of searching keyword that is generated with TCompute function. The CSP test all document of user and find that contain matching keyword and generate the intermediate output and sends to user. Here user decrypts this email document with his private key and gets simple plain text format email. So this scenario overcomes the user computational memory power issue of PEKS.

However there are still some unsolved issues are there for example user sends n bits length cipher text to CSP which increase the cost of the communication to a certain degree. Also here it assumes that the length of the email is same or shorter than n that is not possible in real environment. These all issues are overcome by SPKS.

C. Secure and Privacy Preserving Keyword Search(SPKS):

The SPKS scheme enables the CPS's to participate in the partial decipherment, this will reduce the computational overhead on users, without leaking any information about the plain text. It also supports keyword searching on encrypted data. This scheme will enable the CSP to determine whether the keyword specified by the user is in the email, but it will not be aware of the information contained in the email, nor the keyword that was searched. It is proven to be semantically secure under the Bilinear Diffie Hellman assumption and the random oracle model. The working process of this schema is described in the following Figure 3.

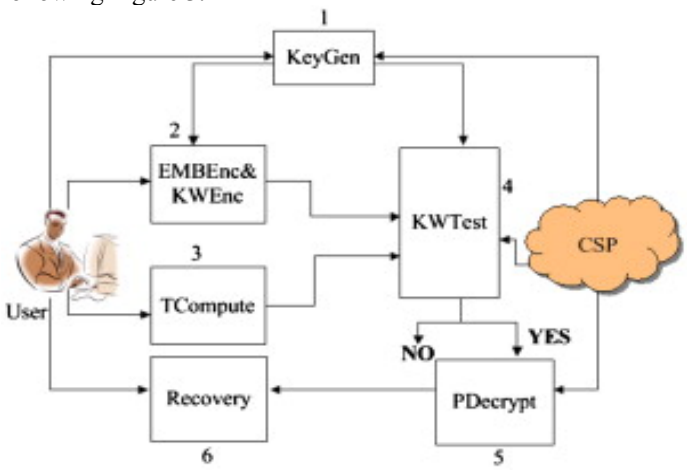

Figure 3 The working process of the SPKS scheme.

This scheme consists of seven randomized polynomial time algorithms, which are the similar to that of in Efficient Privacy Preserving Keyword Searching Scheme. They are as follows:

a. Key Generation:

It takes large security parameter K1 and generates the public key pair for user kwon as (Upub, Upriv). Now also take another large security parameter K2 and generate another public key pair for CSP known as (Spub, Spriv).

b. Email Encryption (EMBEnc):

It uses the public key encryption algorithm.It takes two key Upub, Spub and message M as a input and encrypt the message that known as Cm. So We write EMBEnc (Upub; S pub;m) = Cm.

c. Keyword Encryption (KWEnc):

It is also performs the public key encryption algorithm. It takes the public key Upub and keyword Wi Є W (i Є Z+) as a input and produces cipher text CWi's Є CW. So we can write KWEnc(Upub;Wi) = CWi .

d. Trapdoor Computing (TCompute):

It takes the Upriv key and keyword like Wj Є W (j Є Z+) as a input and generate the Wj's trapdoor known as TWj. So we can write TCompute(Upriv, Wj) = TWj.

e. Testing:

Now it uses the CWi, TWj and Upub as a input and checks the finding keyword. It returns 1 if CWi=TWj otherwise it returns 0.

f. PDecryption:

It takes Upub, Spriv, and Cm as an input and generates the intermediate result Cp. We can write PDecrept(Upub, Spriv, Cm) = Cp.

g. Recovery:

It uses the Upriv to decrypt the Cp. So it takes Upriv and Cp as a input and generate the final output plain text M.

The all randomized polynomial time algorithm is worked as same as EPPKS except PDecryption that generate the partial decryption result.

The user U and the CSP runs the KeyGen algorithm to generate their public or private key pairs that known as (Upub, Upriv) and S(Spriv, Spub). When U wants to store an email M containing keywords W1....Wk on Cloud Servers, U first runs the EMBEnc algorithm to encrypt the email using the Upub and Spub, and then runs KWEnc to encrypt all the keywords using Upub, and finally sends both the ciphertext of the email and keywords to the CSP.

When U wants to retrieve emails containing keyword, he runs the TCompute algorithm to generate Wj's trapdoor Twj using Upriv and sends it to the CSP. On receiving the trapdoor the CSP runs the KWTest algorithm to determine whether a given email contains keyword Wj specified by U.

Before returning the results to U, the CSP runs PDecrypt to calculate an intermediate result for the decipherment using the Upub and Spriv key. After that it returns along with the encrypted emails. When a ciphertext and is given, U runs the Recovery algorithm to recover the plain text.

V. COMPARISION

In SPKS and EPPKS schema, which one is better that identified by comparing the computation and communication cost during encryption as well as decryption.

Case 1: If n is sufficiently large such that it is larger than or the same as the maximal length L of any email. Here we pad the shorter messages to make all the emails to have the equal length n. The results are given in table 1 and table 2.

Table: 1 Computational Cost of Encryption

Operation	*PEKS*	*SPKS*
Map	0	1
Mul	2	2
Exp	0	1
Mod	1	0
Hash	0	3
Xor	0	2

Table: 2 Computational Cost of Decryption

Operation	*PEKS*	*SPKS*
Map	0	0
Mul	1	0
Exp	0	1
Inv	1	0
Mod	1	0
Hash	0	1
Or	0	1

Case 2: Here n is a relatively small number in comparison with the average length of most emails. In this case, we need to split a longer email into several segments and pad the last segment to make each segment to have the equal length n.

Table: 3 Computation Cost of Encryption

Operation	*PEKS*	*SPKS*
Map	0	1
Mul	2	2
Exp	0	1
Mod	M	0
Hash	0	3
Xor	0	M+1

Table: 4 Computational Cost of Decryption

Operation	*PEKS*	*SPKS*
Map	0	0
Mul	1	0
Exp	0	1
Inv	1	0
Mod	M	0
Hash	0	1
Xor	0	M

VI. CONCLUSION

The study of different keyword searching schemes offers a way to overcome one technique's disadvantage. The keyword searching techniques improve the security of the user keyword searching privacy. Now SPKS allows the CSP to participate in the decipherment, thus a user could pay less computational overhead for decryption. It is a searchable encryption scheme, thus the CSP could search the encrypted files efficiently without leaking any information. So from this various approaches like PEKS, SPKS etc, the SPKS is more efficient and privacy preserving keyword search schema.

VII. REFERENCES

[1] Seny Kamara and Kristin Lauter, "Cryptographic Cloud Storage," Microsoft research.

[2] Tritty Mamachan1, and Roshni. M. Thankar, "Survey on keyword searching in Cloud Storages," International Journal of Emerging Technology and Advanced Engineering, ISSN 2250-2459, Volume 2, Issue 11, November 2012.

[3] Liu Hong-xia, Dai Jia-zhu, and Jiang Chao, "Research on Privacy Preserving Keyword Search in Cloud Storage," IEEE publication, 978-1-4244-5540-9/10, 2010.

[4] Qin Liuy, Guojun Wang, and Jie Wuz, "An Efficient Privacy Preserving Keyword Search Scheme in Cloud Computing," Computational Science and Engginerring, IEEE publication, 29-31 Aug 2009.

[5] Qin Liuy, Guojun Wang, and Jie Wuz, "Secure and privacy preserving keyword searching for Cloud Storage Services," Journal of Network and Computer Applications, 9 March 2011.