ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 3, March 2015

Two Round Searchable Encrypted Data using Multi-Keywords for Cloud Computing

Mr. H.D.Gadade[1], Miss.Pranita U. Shinganwade[2], Miss. Punam S. Wanare[3],Miss. Vibha Badhe[4]

Department of Computer Engineering, Government College of Engineering, Jalgaon-India[1,2,3,4]

Abstract*:* Cloud Computing represents vital role in Information Technology. Cloud Computing provides high security for managing & storing large scale data in internet-based Infrastructure. Cloud provides large storage space & make user friendly for fast accessing data. Searchable Symmetric Encryption (SSE) use to secure and retrieve data from the cloud and it also focus on addressing data privacy issues. We observe that server side ranking based on Order Preserving Encryption (OPE) certainly leaks data privacy.OPE use Boolean search technique. To remove the leakage of data, we propose Two Round Searchable Encryption (TRSE) scheme that supports top-k multi-keyword retrieval. In TRSE, we use Vector Space Model & Homomorphic Encryption. This proposed scheme guarantees high security and practical efficiency.

Keywords*:* Cloud, Data privacy, Homomorphic Encryption, Vector Space Model, SSE, OPE

I. INTRODUCTION

Cloud computing is a long dreamed vision of computing as a utility, where cloud customers can remotely store their data into the cloud as to enjoy the on-demand high-quality application and services from a shared pool of configurable computing resources. User store data on cloud, the cloud provide services to control & monitor the data and the communication between client and the cloud. The cloud itself provides data privacy. Public clouds provide third party service & use application from different customers. Private clouds are built to use of one client & manage by organization’s own administrator. Hybrid cloud is combination of public & private cloud model. For security of data, user’s uses encrypt the data before storing the data on cloud. If, it is possible to user to access the encrypted data then user need to communicate with the cloud and it allows other user to use encrypted data, which causes leakage of sensitive information. In Cloud computing, data owners may share their stored data with many users & they might retrieve only those data files that they required. For retrieval of cloud data keyword based technique mainly used. At the time of retrieval system gets ranked to data files in the order of relevance by user’s interest and only those files with the highest relevance are send back to user. If user has not sufficient space to those files then user may have to pay some amount to retrieve those files from the cloud. Novel technologies in the cryptography community and information retrieval community are employed, including homomorphic encryption and vector space model. In the proposed scheme, the majority of computing work is done on the cloud while the user takes part in ranking, which guarantees top k multi-keyword retrieval over encrypted cloud data with high security and practical efficiency.

II. EXISTING SYSTEM

A lot of research is done in cloud data security with number of techniques. In the work [1], describes the cryptographic schemes for the problem of searching on encrypted data and provide proofs of security for the resulting crypto systems. In the work [2], they are considering the problem of Searching on data that is encrypted using a public key system. Consider user Bob who sends email to user Alice encrypted under Alice's public key. An email gateway wants to test whether the email contains the keyword \urgent" so that it could route the email accordingly. In the work [3], introduces a new framework for confidentiality preserving rank-ordered search and retrieval over large document collections. In the work [4], 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 has to be encrypted before outsourcing, which obsoletes traditional data utilization based on plaintext keyword search. In the work [5], Query processing that preserves both the data privacy of the owner and the query privacy of the client is a new research problem. It shows increasing importance as cloud computing drives more businesses to outsource their data and querying services. However, most existing studies, including those on data outsourcing, address the data privacy and query privacy separately and cannot be applied to this problem. We believe this work steps towards practical applications of privacy homomorphism to secure query processing on large-scale, structured datasets. As for future work, we plan to extend this work to other query types, including top-k queries, skyline queries and multi-way joins. [6]

III. PROPOSE SYSTEM

We introduce the concepts of similarity relevance and scheme robustness to formulate the privacy issue in searchable encryption schemes, and then solve the insecurity problem by proposing a two-round searchable encryption (TRSE) scheme. Novel technologies in the cryptography community and information retrieval community are employed, including homo-morphic encryption and vector space model.

DOI 10.17148/IJARCCE.2015.4338

In the proposed scheme, the majority of computing work is done on the cloud while the user takes part in ranking, which guarantees top k multi-keyword retrieval over encrypted cloud data with high security and practical efficiency. Contributions can be summarized as follows**:**

1. We propose the concepts of similarity relevance and scheme robustness. We, thus, perform the first attempt to formulate the privacy issue in searchable encryption, and we show server-side ranking based on order-preserving encryption (OPE) inevitably violates data privacy.
2. We propose a TRSE scheme, which fulfills the secure multi-keyword top-k retrieval over encrypted cloud data. Specifically, for the first time, we employ relevance score to support multikeyword top-k retrieval.
3. Thorough analysis on security demonstrates the proposed scheme guarantees high data privacy. Furthermore, performance analysis and experimental results show that our scheme is efficient for practical utilization. [7]
4. If user has to access the data from cloud and there is no space in user system then user has to access cloud data by paying some charges for that data.

IV. ADVANTAGES

The concepts of similarity relevance and scheme robustness. It perform the first attempt to formulate the privacy issue in searchable encryption, and we show server side ranking based on order-preserving encryption (OPE) inevitably violates data privacy The two-round searchable encryption (TRSE) scheme, which fulfills the secure multi-keyword top-k retrieval over encrypted cloud data. Specifically, for the first time we employ relevance score to support multi-keyword top-k retrieval. Thorough analysis on security demonstrates the proposed scheme guarantees high data privacy. Furthermore, performance analysis and experimental results show that our scheme is efficient for practical utilization. [8]

V. SYSTEM ARCHITECTURE

In Fig 1, the various components present in the architecture of proposed system are actual user stores the data on cloud. Cloud server stores the encrypted data and searching indexes. Data user retrieves the file from the cloud server using top-k multikeyword ranked search. [6]

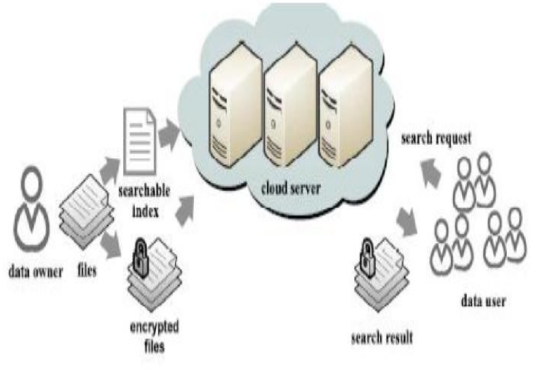

Fig.1. Component and Architecture of System

In this system data owner store their files with encrypted files and searchable index which is also encrypted. This generate private and public key and it store on cloud. At the time of decryption data user send search request with query and decrypt files with their private key.

VI. TRSE DESIGN

Existing Searchable Symmetric Encryption (SSE) scheme works server-side ranking based on Order Preserving Encryption (OPE) to improve the efficiency of retrieval of encrypted cloud data. Hence, server-side ranking based on OPE is not sufficient for privacy of sensitive data. To improve data privacy, ranking has to be given to the user side. In user-side ranking, the user has load heavy computational burden and high computation, due to the communication between server and user by adding searchable index return and calculate rank score. The user-side ranking is used for practical used. To provide high privacy server side ranking is might be better.

To improve privacy of data, we propose new Searchable Encryption technique, by using the Homomorphic Encryption and Vector Space Model. In proposed technique the data owner encrypts the searchable index using Homomorphic Encryption. When cloud server takes a query providing of multikeywords, it calculates the score from encrypted index and returns the encrypted scores of file to the user. Then user decrypt the score and picks out a top-k highest score files identifier to request to the cloud server. The retrieval takes a two round communication between the cloud server and the user. For that this technique is called the Two Round Searchable Encryption (TRSE) scheme, this ranking is done on user-side and score calculation is done at the server-side.

The TRSE scheme includes five phases:

- Setup Phase
- Index Build Phase
- Trapdoor Gen Phase
- Score Calculate Phase
- Rank Phase

A. Setup Phase:

In this phase, Cryptography Community is used. The data owner generates the secret key and public key. By using this secret key we encrypt the stored data on cloud.

B. Index Build Phase:

The data owner creates the searchable index from the data and also gives the security to it. Information Retrieval community is used to create search index. By using the collection of files, extracts the collection of words and gives the similarity between these words. The secrete key is used to encrypt the search index with its similarities. The encrypted data and encrypted search index are stored on the cloud.

C. TrapdoorGen Phase:

The user used query with keywords to request the data from the cloud. The keywords are formed such as 0 and 1 called as vector query. This query generates secure trapdoor by using homomorphic encryption. Then this encrypted vector query sends to the cloud.

D. Score Calculate Phase:

The cloud receives the request from the user. The cloud gives the score for each file on cloud using the request query, for giving the score Vector Model is applied, from that we get the Encrypted Vector. This Encrypted Vector is return back to the user.

E. Rank Phase:

The Encrypted Vector is received by the user and applied the Homomorphic decryption for top-k scores to the related files.

VII. ALGORITHM

A. RSA Algorithm :

i. Key Generation Algorithm :

Generate an RSA key pair.

INPUT: Required modulus bit length, *k*.

OUTPUT: An RSA key pair ((N,e), d) where N is the modulus, the product of two primes (N=pq) not exceeding *k* bits in length; e is the public exponent, a number less than and coprime to (p-1)(q-1); and d is the private exponent such that ed≡1(mod(p-1)(q-1)).

```
1.    Select a value of e from {3, 5, 17, 257, 65537}
2.    repeat
3.       p ← genprime(k/2)
4.    until (p mod e) ≠ 1
5.    repeat
6.       q ← genprime(k - k/2)
7.    until (q mod e) ≠ 1
8.    N ← pq
9.    L ← (p-1)(q-1)
10.   d ← modinv(e, L)
11.   return (N, e, d)
```

The function genprime(b) returns a prime of exactly b bits, with the bth bit set to 1. Note that the operation k/2 is integer division giving the integer quotient with no fraction.

If you've chosen e = 65537 then the chances are that the first prime returned in steps (3) and (6) will pass the tests in steps (4) and (7), so each repeat-until loop will most likely just take one iteration. The final value of N may have a bit length slightly short of the target k. This actually does not matter too much (providing the message m is always < N), but some schemes require a modulus of exact length. If this is the case, then just repeat the entire algorithm until you get one. It should not take too many goes. Alternatively, use the trick setting the two highest bits in the prime candidates described in [9].

ii. Encryption :

Sender A does the following:-

1. Obtains the recipient B's public key (n, e).
2. Represents the plaintext message as a positive integer *m*, 1 < m < n .
3. Computes the ciphertext c = m^e mod n.
4. Sends the ciphertext *c* to B.

iii. Decryption :

Recipient B does the following:-

1. Uses his private key (n, d) to compute m = c^d mod n.
2. Extracts the plaintext from the message representative *m*.[9]

B. *Stemming Algorithm* :

```
integers ( p1 p2 )
booleans ( Y_found )

routines (
   shortv
   R1 R2
   Step_1a Step_1b Step_1c Step_2 Step_3 Step_4
Step_5a Step_5b
)

externals ( stem )

groupings ( v v_WXY )

define v        'aeiouy'
define v_WXY     v + 'wxY'

backwardmode (

    define shortv as ( non-v_WXY v non-v )

    define R1 as $p1 <= cursor
    define R2 as $p2 <= cursor

    define Step_1a as (
        [substring] among (
            'sses' (<-'ss')
            'ies'  (<-'i')
            'ss'   ()
            's'    (delete)
        )
    )

    define Step_1b as (
        [substring] among (
            'eed'  (R1 <-'ee')
            'ed'
            'ing' (
                test gopast v  delete
                test substring among(
                    'at' 'bl' 'iz'
                         (<+ 'e')
                    'bb' 'dd' 'ff' 'gg' 'mm' 'nn' 'pp' 'rr' 'tt'
                    // ignoring double c, h, j, k, q, v, w, and x
                         ([next]  delete)
                    ''   (atmark p1  test shortv  <+ 'e')
                )
```

DOI 10.17148/IJARCCE.2015.4338

```
      )
   )
)

define Step_1c as (
   ['y' or 'Y']
   gopast v
   <-'i'
)

define Step_2 as (
   [substring] R1 among (
      'tional'  (<-'tion')
      'enci'    (<-'ence')
      'anci'    (<-'ance')
      'abli'    (<-'able')
      'entli'   (<-'ent')
      'eli'     (<-'e')
      'izer' 'ization'
              (<-'ize')
      'ational' 'ation' 'ator'
              (<-'ate')
      'alli'    (<-'al')
      'alism' 'aliti'
              (<-'al')
      'fulness' (<-'ful')
      'ousli' 'ousness'
              (<-'ous')
      'iveness' 'iviti'
              (<-'ive')
      'biliti'  (<-'ble')
   )
)

define Step_3 as (
   [substring] R1 among (
      'alize'   (<-'al')
      'icate' 'iciti' 'ical'
              (<-'ic')
      'ative' 'ful' 'ness'
              (delete)
   )
)

define Step_4 as (
   [substring] R2 among (
      'al' 'ance' 'ence' 'er' 'ic' 'able' 'ible' 'ant' 'ement'
      'ment' 'ent' 'ou' 'ism' 'ate' 'iti' 'ous' 'ive' 'ize'
              (delete)
      'ion'     ('s' or 't' delete)
   )
)

define Step_5a as (
   ['e']
   R2 or (R1 not shortv)
   delete
)

define Step_5b as (
      ['l']
      R2 'l'
      delete
   )
)

define stem as (

   unset Y_found
   do ( ['y'] <-'Y' set Y_found)
   do repeat(goto (v ['y']) <-'Y' set Y_found)

   $p1 = limit
   $p2 = limit
   do(
      gopast v  gopast non-v  setmark p1
      gopast v  gopast non-v  setmark p2
   )

   backwards (
      do Step_1a
      do Step_1b
      do Step_1c
      do Step_2
      do Step_3
      do Step_4
      do Step_5a
      do Step_5b
   )

   do(Y_found  repeat(goto (['Y']) <-'y'))

) [10]
```

VIII. CONCLUSION

We motivate and solve the problem of secure multikeyword top-k retrieval over encrypted cloud data. We define similarity relevance and scheme robustness. Based on OPE invisibly leaking sensitive information, we devise a server-side ranking SSE scheme. We then propose a TRSE scheme employing the fully homomorphic encryption, which fulfills the security requirements of multikeyword top-k retrieval over the encrypted cloud data. By security analysis, we show that the proposed scheme guarantees data privacy. According to the efficiency evaluation of the proposed scheme over a real data set, extensive experimental results demonstrate that our scheme ensures practical efficiency.

REFERENCES

[1] D. Song, D. Wagner, and A. Perrig, “Practical Techniques for Searches on Encrypted Data,” Proc. IEEE Symp. Security and Privacy, 2000.

[2] D. Boneh, G. Crescenzo, R. Ostrovsky, and G. Persiano, “Public-Key Encryption with Keyword Search,” Proc. Int’l Conf. Theory and Applications of Cryptographic Techniques (Eurocrypt), 2004.

[3] A. Swaminathan, Y. Mao, G.-M. Su, H. Gou, A.L. Varna, S. He, M.Wu, and D.W. Oard, “Confidentiality-Preserving Rank-Ordered Search,” Proc. Workshop Storage Security and Survivability, 2007.

DOI 10.17148/IJARCCE.2015.4338

[4] N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, "Privacy-Preserving Multikeyword Ranked Search over Encrypted Cloud Data," Proc.IEEE INFOCOM, 2011.

[5] H. Hu, J. Xu, C. Ren, and B. Choi, "Processing Private Queries over Untrusted Data Cloud through Privacy Homomorphism," Proc. IEEE 27th Int'l Conf. Data Eng. (ICDE), 2011.

[6] C.Rajeshkumar and Dr.K.Rubasoundar ''Retrieval of Encrypted cloud data using multikeyword'' [IJIRCCE-Vol.2, Special Issue 1, March 2014]

[7] Juvairiya K P and Rasheeda Z Khan ''Revival of Secure Top-k Multi-Keyword over Encrypted Cloud Data'' [*IJCTT – volume 9 number 2– Mar 2014*]

[8] Mrs. P.Shanmuga Priya M.E(Ph.d), Preethi.D, Priya.J and shanthini.B ''Retrival of Encrypted Data Using Multi Keyword Top –K Algorithm'' [IJSRP- Volume 4, Issue 4, April 2014]

[9] http://www.di-mgt.com.au/rsa_alg.html

[10] http://snowball.tartarus.org/algorithms/porter/stemmer.html

BIOGRAPHIES

Mr.H.D.Gadade, ME(Computer Engg),Government College of Engineering ,Jalgaon

Miss Pranita U.Shinganwade, BE(Computer Engg.),Government College of Engineering ,Jalgaon

Miss. Punam S. Wanare, BE(Computer Engg.),Government College of Engineering ,Jalgaon

DOI 10.17148/IJARCCE.2015.4338