Impact Factor (SJIF): 3.632

International Journal of Advance Research in Engineering, Science & Technology

e-ISSN: 2393-9877, p-ISSN: 2394-2444

Volume 3, Issue 5, May-2016

A Survey on Encrypted Searching

Neel Shah[1]

Information Technology[1]*, PIET*

Abstract: Goverment and enterprise producing large amount of data which start a new trend of outsourcing data to cloud. Ubiquitous nature of cloud making it more popular. But, In real world third party data-center is not completely trustworthy. For privacy preserving and security all data should be encrypted. But encrypted data took away all capability to search and processing on data by both unauthorized and authorized users. Becaused of that new trend encrypted searching emerged.

Keyword: Encrypted searching, privacy preserving , cloud storage.

I. Introduction

As outsourced data get bigger, it become more informative and privacy sensitive. But storing of data to cloud(third party data-center) make it valuable, because administrator or hacker with root rights have full access to data. Imagine, your company outsource all sensitive data to cloud which is stored to untrusted third party data-center and they have full access to your data. To secure that data, data has to be encrypted. Data encryption make it unaccessible for insider and outsider. But it also disable searching functionality on data. To provide searching functionality, hole data is to be downloaded than decrypt it locally and search for needed data. Which is impractical way. Another way is to decrypt data by server and search on it, but it allow server to know about it. Instead, providing searching functionality on the server side without decrypting data. Which is called encrypted searching.

II. Different approach for encrypted searching

Currently, encrypted searching scheme construction achieve between three trade off security, query expressiveness and efficient search on large database.

2.1 Security and Expressiveness VS Efficiency

At present data is in large size stored at cloud. Then there is a need of higher efficiency to search on large amount of encrypted data. When query expressiveness and security maximized, the best solution based on full homo-morphic(FHE) encryption and ORAM. But this approach are very inefficient for a practical use. Present all FHE scheme require linear time to search. ORAM schemes are better than FHE but not practical for large amount of data.

2.2 Efficiency and Expressiveness VS Security:

When query expressiveness and efficiency are maximized, best solution is given through deterministic(Property preserving) encryption. But use of deterministic encryption leak critical information about data. So it is not used for encrypted searching because of weak security.

2.3 Security and Efficiency VS Expressiveness:

When security and efficiency are maximized, best solution is given through structural encryption. Benifit of structural encryption is sub-linear searching with leakage of very small amount of information. But it is not provide high level SQL query which limit query expressiveness but it is good for noSQL data.

2.4 Security definition:

Sang[] first introduced security definition for encrypted searching. Their constructed scheme is even valuable against chosen plaintext attack(IND-CPA). Scheme is IND-CPA secure if an adversary can not distinguish encryption of two arbitrary message, even if adversary can adaptively query on encryption oracle.

Goh[] introduced second definition for encrypted searching, who define semantic security against adaptive chosen keyword attack(IND-CKA1). IND-CKA1 secure scheme means adversary deduce the document's content from its index.

Custmala[] introduced introduced adaptive sematic security against adaptive chosen keyword attack(IND-CKA2). It means security for trapdoors and guarantee that trapdoors do not leak information about the keywords in index. It is widely used security standard for encrypted searching.

III. Survey of basic scheme

Song, Wanger, Perrig have proposed SWP[10] scheme to search over encrypted data which first scheme of encrypted searching. SWP scheme uses bitwise exclusive OR between plaintext and sequence of pseudo-random number. Using specific structure user can search over encrypted data. It supports only one user to perform read and write operation. Scheme takes linear time for searching. It is not providing IND-CPA security.

Linear time is required for searching encrypted keyword in SWP scheme. This time is reduced in Goh's[12] scheme which uses Bloom Filter (BF) for every document. Using of BF gives a result of membership query in constant time. Pre-computation is costly because computing n keyword per document takes linear time for creating an index. Even for small document number of the keyword are numerous. And encryption of index makes a relatively large static index in size. Usage of BF results in a possibility of false positive which is depend on a number of hash function, array size and element in the document. This scheme is proven IND1-CKA secure.

Curtmola[1] proposed scheme consists of two construction CGK-1 and CGK-2. The idea behind this scheme is to use an inverted index. It means index per word instead of a document makes time complexity sub-linear. Index array constructed using link list. Each node in link list contains three fields: document identifier, the key to encrypt next node and pointer to next node. Using trapdoor user can find the correct position of the file and decrypt it. Because of link list property server can find all matching nodes and decrypt it all. Worst case to search file depends on a maximum number of the file in a database. Updates are costly because of link list structure. This scheme is good for the static dataset. It is IND-CKA2 secure.

Chang and Mitzenmacher[11] scheme use index per document by existing dictionary for searching keyword per document. M-bit array is used to create an index which initially set to 0 & position of each keyword is mapped to the dictionary. 1 bit is set when the keyword is present in a document. The index is stored in the home machine in case of computer or at server side in case of a mobile device which increases overhead. Creating indexes in advance, results in increased time and computational overhead since index per file is needed. Searching file relies on a number of document present in the database that gives linear time in worst case scenario. This scheme is provided IND2-CKA2 security.

Amanatidis[2] construct scheme using deterministic MAC & encryption. The user generates MAC of each keyword. MAC and encrypted keyword at a server. The index is made using MAC which uses to search correct answer. Index generating take linear time per document. Search file takes logarithmic time which depends on the database size. This scheme is IND-EASE secure proven. The scheme is not secure against chosen keyword attack.

Van Liesdonk[3] scheme provides efficient search and update. The idea behind this scheme related to one index per keyword logic. To generate index per document is depend on distinct words in linear time. Search time to find the file with a keyword is in logarithmic. This scheme also provides an efficient update. This scheme is IND-CKA2 proven.

Kamara and Chase's[4] scheme are adaptively secure construction. The scheme is based on CGK[paper-4]. In the scheme, an inverted index is generated with permuted dictionary. To create dictionary hash is use which gives the result in optimal time. Linear time is required to generate an index for a distinct keyword of document in the database. Searching is done through the inverted index which contains hash compare them with user requested hash. This scheme hides the data structure and IND-CKA2 proven.

Kurosawa and Ohtaki[5] scheme MAC inside index for query expression. It uses MAC and PRF to generate an index. Index generation takes linear time depends on a number of the document. To search document, it compares with all document in the database which falls in linear time complexity. It is a verifiable scheme and provide keyword privacy. It is IND-CKA2 proven secure.

Kamara[6] proposed a scheme to perform add, delete or modified the document in an efficient manner. It is an extension of CGK. In scheme deletion array is add for tracking of update or modification of the document. It uses homomorphic encryption to encrypt array pointers to update pointer without decryption. It also maintains free array which has a list of free position in search array. This scheme uses PRF and XOR for searching. To search thy perform XOR with all node of an array which falls in linear time complexity. It also performs eight rounds for index generation. It leaks little information during an update operation. It is IND-CKA2 proven.

Cash[7] proposed conjunctive quires for arbitrarily structured data scheme. It uses inverted index of Curtmola which scalable for large data size. But the user needs decryption of IDs before retrieving searched documents. For index

generation takes one exponentiation computation performed per document. Search query performs XOR-bitwise operation per document. It falls in sublinear time complexity. It is proven IND-CKA2 secure.

Contructions	Technique	Security	Searching Efficiency
SWP[10]	Bitwise exclusive OR	Not support IND-CPA	Linear
Goh[12]	Bloom Filter (BF)	IND-CKA1	Linear
CGK-2[1]	Inverted index	IND-CKA2	Sub-Linear
Chang and Mitzenmacher[11]	M-bit array with index	IND-CKA2	Linear
Amanatidis[2]	Deterministic MAC & encryption.	IND-CPA	Sub-Linear
Van Liesdonk[3]	Index per keyword with deterministic ecryption	IND-CKA2	Sub-Linear
Kamara and Chase[4]	Inverted index with hash	IND-CKA2	Sub-Linear
Kurosawa and Ohtaki[5]	MAC inside index for query expression	IND-CKA2	Linear
Kamara[6]	Hmomorphic encryption	IND-CKA2	Linear
Cash[7]	Inverted index	IND-CKA2	Sub-Linear

Table 1: Survey of scheme

IV. Future work

We observe most schemes are achieved higher security regardless of their efficiency drawback and lack of query expressiveness which make them hard for deployment in the practical world.

References:

[1]R. Curtmola, J. Garay, S. Kamara and R. Ostrovsky, "Searchable symmetric encryption", *Proceedings of the 13th ACM conference on Computer and communications security - CCS '06*, 2006.

[2]G. Amanatidis, A. Boldyreva and A. O'Neill, "Provably-Secure Schemes for Basic Query Support in Outsourced Databases", *Data and Applications Security XXI*, pp. 14-30.

[3]S. Sedghi, P. van Liesdonk, S. Nikova, P. Hartel and W. Jonker, "Searching Keywords with Wildcards on Encrypted Data", *Lecture Notes in Computer Science*, pp. 138-153, 2010.

[4]M. Chase and S. Kamara, "Structured Encryption and Controlled Disclosure", *Advances in Cryptology - ASIACRYPT 2010*, pp. 577-594, 2010.

[5]K. Kurosawa and Y. Ohtaki, "UC-Secure Searchable Symmetric Encryption", *Financial Cryptography and Data Security*, pp. 285-298, 2012.

[6]S. Kamara, C. Papamanthou and T. Roeder, "Dynamic searchable symmetric encryption", *Proceedings of the 2012 ACM conference on Computer and communications security - CCS '12*, 2012.

[7]D. Cash, S. Jarecki, C. Jutla, H. Krawczyk, M. Roşu and M. Steiner, "Highly-Scalable Searchable Symmetric Encryption with Support for Boolean Queries", *Advances in Cryptology – CRYPTO 2013*, pp. 353-373, 2013.

[8]S. Kamara, "Encrypted Search", *XRDS: Crossroads, The ACM Magazine for Students*, vol. 21, no. 3, pp. 30-34, 2015.

[9]C. Bösch, P. Hartel, W. Jonker and A. Peter, "A Survey of Provably Secure Searchable Encryption", *CSUR*, vol. 47, no. 2, pp. 1-51, 2014.

[10] Dawn Xiaoding Song, D. Wagner and A. Perrig, "Practical techniques for searches on encrypted data", *Proceeding 2000 IEEE Symposium on Security and Privacy. S&P 2000*.

[11]Y. Chang and M. Mitzenmacher, "Privacy Preserving Keyword Searches on Remote Encrypted Data", *Applied Cryptography and Network Security*, pp. 442-455, 2005.

[12]Goh, Eu-Jin. "Secure Indexes." 2004.