Literature Review on Enhance Performance of Private Information Retrieval in Outsourcing Database Services

[1]Dr. Raghav Mehra, [2]Durga Pooja

[1]Asso. Prof., [2]Research Scholar
[1,2]Computer Science & Engineering
[1]Bhagwant University, Ajmer, Rajasthan ,India.

Abstract : In this paper , We suggest a scheme for outsourcing (PIR) Private Information Retrieval to untreated servers while shielding the isolation of the record owner as well as that of the database client. This introduces several protection issues related to database. Many organizations use data in various forms such as those in image processing, Geographical Information Systems (GISs), medical records, financial transaction records, employee information and other applications. For most clients (e.g., organizations), databases play a grave role in their continuation and development. In outsourced databases, the parties those subcontract their data will be referred as data owners also clients hereafter. The service providers storing data will be referred to as servers. In outsourced database scenario, the clients rely on servers for storage, retrieval, and maintenance of their databases. Outsourcing alleviates their need to procure expensive hardware and software, or to pay to professionals to deploy, maintain, and upgrade the systems, which are now taken care of by the external server whose security level is normally not fully trusted. Hence, outsourced database model introduces abundant security examine challenges. To ensure data discretion, outsourced data is typically encrypted and then querying is conceded out with the support of trusted client front-ends or secure coprocessors. However, encrypting data is only a revenue of securing data. Several other security issues such as data privacy, efficient query processing over encrypted databases, key management, and data integrity remain in this model. Although ensuring security related issues is primary requirement, addressing performance guarantee is also required.
In this model method encryption maintains the data secrecy and a convenience that help to construct PIR reply which maintains data isolation. Single-file PIR schemes are generally unusable because of its expensiveness from computational point of view. This model suggests use of (GPU) Graphics Processing Unit service to process database. This model will help to accomplish practical discharge of single-database PIR scheme which uses encryption algorithm to preserve data secrecy as secondary objective and it serves to generate database reply to maintain data privacy as primary goal. In this paper we shows some real time consequences that proves necessitate of concurrency in P.I.R. algorithms. It paper describes use of trellis based single-database PIR protocol with GPU as processing unit to achieve high speed-up & hence, the performance of system.

Index Terms **- Outsourced Database, secretive Information Retrieval, cloud computing, User Privacy, Data Privacy.**

I. INTRODUCTION

Database outsourcing has become popular due to advances in networking technologies and unrelenting growth of the Internet. It has triggered a new trend of outsourcing data management and information technology needs to external service providers. Many organizations use data in various forms such as those in image processing, Geographical Information Systems (GISs), time-series databases, Computer Aided Design (CAD) or Computer Aided Manufacturing (CAM), and other applications. For most clients (e.g., organizations), databases play a crucial role in their existence and development. In outsourced databases, clients rely on service providers for storage, retrieval, and maintenance of their databases. Outsourcing alleviates their need to purchase pricey hardware and software, or to pay professionals to deploy, preserve, and upgrade the systems, which are now taken care of by the service provider. It is indispensable to offer adequate security measures to protect this stored data from both malicious outsiders and the service provider itself. Encryption is a popular technique for ensuring secrecy of sensitive data. While data encryption shall be able to enhance security greatly, that can impose substantial overhead on the performance of system in stipulations of data management.

PIR schemes usually require the an gigantic amount of computational power, but allowing for the huge number of applications these protocols have, it is imperative to develop practically implemented protocols that provide acceptable performances for as many applications as possible[1] . The cost computational for a server replying forn PIR uncertainty is therefore linear on database size. Moreover, theoretic-number schemes have a very pricey cost per bit burgeoning over a bulky modulus in database. This limits both the database size & throughput shared by users.

II. LITERATURE REVIEW

This paper deals during an updated survey and review of different protection concerns of database outsourcing scenarios. A focused approach was taken to carry out the literature survey by dividing the survey into four parts:

1) PIR- Private Information Retrieval Protocol
2) Protection of indexes in outsourced databases
3) Searching on encrypted databases
4) Protection issues associated with database outsourcing model

The Search PIR (Private Information Retrieval) offers a solution to the information escape problem by thrashing access patterns, independent of an encryption mechanism. In PIR, a database store at a server holds n strings each of size l bits, and a user can doubt for one l bit string without leaking which string to the database. A slight way to realize this is to simply transmit the entire database to the user. However, this is communication inefficient, and explore focuses on achieving lower communication bounds. In disparity, computational cost of every PIR design is essentially $O(n \cdot l)$, as the server necessity “contact” every piece of the database if the server is to stay put unconscious of the requested piece.

1 PIR Protocol : It is the assignment of enticing an item from a database server not including the server ethnicity which item the client is involved . In the context of PIR, an "item" is regularly deliberation of as a single bit out of an n-bit database, but it could also be a "block" of size b bits. There are a wide range of applications such as those dealing with medical records, video or song databases, in which one may want to retrieve information in a database without the database knowing which information is being retrieved. The main performance determine used for these schemes is communication cost, disregard computational cost. A many applications have been anticipated for PIR, including rights and pharmaceutical databases . A insignificant solution to the problem PIR is that the client asks the server for the whole database and looks up the needed bit or blocks itself. Note that in the information-theoretic case, the uncontrolled power is only to be used to try to compromise client's privacy; in either case we immobile insist that the client & servers use only polynomial-time computation to achieve the protocol. Information heoretic sub-linear PIR is possible when there are ℓ servers, each with copy of database-assuming that the servers do not collude to determine the client's query. A t-private ℓ-server PIR is a PIR system in which the isolation of the query is order hypothetically sheltered, even if up to t of the ℓ servers collud. Beimel and Stahl investigated the case where servers can fail to take actions. In this event, it is important that the client immobile be able to salvage her answer. If only k of the ℓ servers need to take action, and no federation of up to t servers can learn any information about the uncertainty, they call such a system t-private k-out-of-ℓ PIR. Ian Golberg presented a Byzantinerobust PIR protocol which provides information-theoretic privacy protection against federation of up to all but one of the respond servers to return inaccurate information while immobile enable the user to compute the exact result.

2 Protection of Indexes in Outsourced Databases: The indexing scheme anticipated in suggests encrypting the complete database row and conveying a set identifier to each value in this row. When searching a unambiguous value, its set identifier is calculated and then passed to the server, who, in turn, returns to the client a anthology of all rows with values assigned to the same set. Finally, the client searches the explicit value in the returned compilation and retrieves the required rows. However, only client can now present the B-Tree traversal, by executing sequence of the queries. Each query salvage a node positioned at a deeper level of B-Tree.

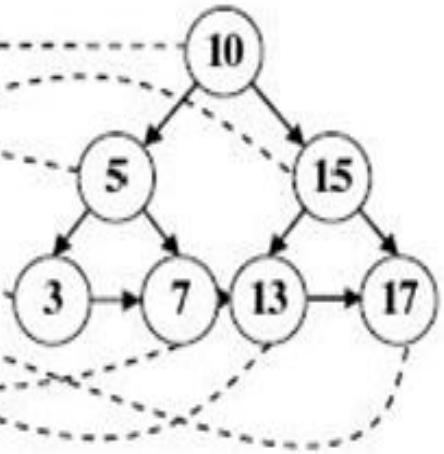
The scheme indexing provided in proposes encrypting each page index using a dissimilar depending key on the page number. However, these schemes, which are implement at the level of the operating system, are not acceptable, since in most cases it is not feasible to transform the operating system functioning. Moreover, in these schemes, it is not possible to encrypt different portions of the database using different keys.

The encryption database scheme in suggest encrypting each database cell with its inimitable cell coordinate μ (T, R, C) and each index value concatenated with its inimitable row identifier, as illustrated in Figure 2.1 .

a) Table T before Encryption

Row	C
0	10
1	5
2	3
3	15
4	17
5	13
6	7

b) Index before Encryption



c) Encryption of Table T in [4]

Row	C
0	$E_k(10 \oplus \mu(T,0,C))$
1	$E_k(5 \oplus \mu(T,1,C))$
2	$E_k(3 \oplus \mu(T,2,C))$
3	$E_k(15 \oplus \mu(T,3,C))$
4	$E_k(17 \oplus \mu(T,4,C))$
5	$E_k(13 \oplus \mu(T,5,C))$
6	$E_k(7 \oplus \mu(T,6,C))$

d) Encryption of the Index in [4]

Row	Struc.	Data
0	1,2	$E_k(10 \| 0)$
1	3,4	$E_k(5 \| 1)$
2	5,6	$E_k(15 \| 3)$
3	6	$E_k(3 \| 2)$
4	5	$E_k(7 \| 6)$
5	4	$E_k(13 \| 5)$
6	-	$E_k(17 \| 4)$

Figure 1: Database and Index Encryption

3 Searching on Encrypted Databases:

Damiani, E., et al. have proposed a solution to the problem of database outsourcing on untrusted servers by providing a hash-based method for database encryption suitable for selection queries.

Hacig¨um¨us, H., et al. have proposed a database service provider that provides users with power to create, store, adjust, and repossess data from somewhere in the world as long as they are allied to the Internet. They proposed a solution in which data at the service contributor is stored in encrypted form and it can be decrypted simply by the owner. Their technique deploys a "coarse index", which allows partial carrying out of an SQL query on the bringer side. The result of this query is sent to the client. The acceptable result of query is found by decrypting data, & executing reward query at client site. Technique operates the SQL query, and splits it into a server query and client query. Hence the client gets total privacy, at the cost of cooperate in query implementation with the package provider.

Dawn Song, et al. presented various practical techniques for remote probing on Encrypted Data using an untreated server. They described various schemes cryptographic for the problem of penetrating on data encrypted and provided proofs of protection for the resultant cryptosystems.

Dang Tran Khanh has discussed protection issues in outsourced databases that come mutually with search trees and presented technique to ensure confidentiality in the implementation of these basic operations on the untreated server. Basic operations of search trees include search and updates. Privacy means the outsourced tree structure and data as well as user's queries are all hidden from unauthorized users.

4 Protection Issues Associated with Outsourcing Database as Service Model:

The model suggests encryption of outsourced data as a means to secure it. However, encrypting data is only a means of securing data. Several other associated issues that still remain to be addressed are described below. The first issue is that of management of keys among users who need to access the database. Different users have dissimilar levels of access on the data contained in the database. A straightforward approach is to identify the common patterns of data sharing so that the number of keys used to encrypt the data can be reduced. This problem is similar to that of role-engineering in role-based access control systems, where the problem is to group the users into a minimum number of roles. However, the problem in data sharing is more complicated as the data sharing pattern are more dense and intricate and the solutions for role-engineering do not apply directly to this problem..

The second issue is that of thesceneryof information storage and retrieval. Typically, the possessor of the database would prefer that the data is stored in such a manner that any query on the data does not reveal either the scenery of the query or the scenery of the results. Furthermore, queries should be performed in an efficient manner observance in intelligence the communication overhead. This involves protecting the type of indexes used in the database system, obfuscating the query execution and protecting the results from inference. Indexes of the database system are one of the primal means to gather valuable information about the structure of the database and need to be protected for safeguarding the database itself. Query execution can be obfuscated to prevent attackers from inferring the sceneryof the database based on increased activity in the database. Typically, the database should show uniform transaction activity regardless of the type of doubt being executed. This would prevent a user from trying to infer the kind of data contained in that part of the database and/or trying to summarize the type of doubt that was executed.

The third issue is that of data integrity. Typically, databases support write operations like append and update. Due to susceptible scenery of the data, it is required that users add information like cryptographic checksums to ensure integrity of data. This again brings in the issue of key management. Users would typically like to reduce the number of such cryptographic operations they perform as it affects the recital of the database. Due to atomicsceneryof these transactions, these operations need to be efficient. In outsourced databases, users typically want message communication overhead to be smaller as integrity checksums are calculated on the client area before sending the data over to the outsourced database for updates.

III. RELATED WORK

Searchable symmetric encryption (SSE) has been extensively studied [28, 15,16,10,13,11,23] (see [13,11] for more on related work). Most SSE research focused on single-keyword search, and after several solutions with density linear in the database size, Curtmola et al. [13] present the first solution for single-keyword search whose density is linear in the number of identical documents. They also improve on previous security models, in particular by providing an adaptive security definition and a solution in this model.

Extending single-keyword SSE to request by conjunctions of keywords was considered in [16,7,2], but all these schemes had O(|DB|) search complexity. The first SSE which can knob very large database and ropes conjunctive queries is OXT protocol discussed above, given by Cash et al.[9].MC-SSE & OSPIR schemes we current are based on this protocol & they preserve its performance and privacy characteristics.

Extension of the two-party client-server model of SSE to setting multi-client was considered by Curtmola et al, [13], but their model disallowed per-query interaction between the data owner and the client, leading to a relatively inefficient implementation based on broadcast encryption. Multi-client SSE setting which allows such interaction was considered by Chase & Kamara [11] as SSE that "illicit disclosure", and by Kamara and Lauter [22], as "virtual confidential storage", but both consider only single-keyword queries and did not support query privacy from the data owner. De Cristofaro et al. [12] extended multi- client SSE to OSPIR setting, that supports query privacy, but only for case of single-keyword queries. In recent independent work, Pappas et al. [26] provide support for boolean queries in a setting similar to our OSPIR setting (but with honest-but-curious clients).SSE schemes which support efficient updates of the encrypted database appeared in [29,23] for single-keyword SSE. The OXT SSE scheme of [9] that supports arbitrary boolean queries, has been extended to dynamic case in [8], & the same techniques apply to MC-SSE and OSPIR schemes presented in this paper.

Recently Islam et al. [19] showed that frequency analysis revealed by access control pat- terns in SSE schemes can be used to predict single-keyword queries. Such attacks, although harder to stage, are possible for conjunctive queries as well, but the general masking and padding countermeasures suggested in [19] are applicable to the MC-OXT and OSPIR-OXT protocols.

In other directions, SSE was extended to the public key setting, allowing any party to encrypt into the database, first for single-keyword search [5,30,1,3,6,27], & later for conjunctive as well queries [6], but all these PKSE schemes have O(|DB|) search complexity. Universally composable SSE were introduced by [24], also with O(|DB|) search complexity.

Multi-client SSE and OSPIR models are related to work on multi-client ORAM, e.g. see the recent work of Huang & Goldberg [17], which aims for stronger privacy protection of client's queries from server E , but multi-client ORAM supports DB lookups by (single) indexes instead of (boolean formulas on) keywords, and they can currently support much smaller DB sizes.

IV. CONCLUSION

Our model of PIR gives an idea of data isolation as well as data confidentiality. It also propose the use of Genera l-Purpose computation using Graphics Processing Units (GPGPU) for accomplishment to enhance speed of implementation of protocol, which will leads to convenient usability of PIR schemes in real world. By looking systematically at the body of work done on outsourcing databases, it is clear that several areas have received deep treatment and some clear recommendations for outsourcing databases as a service have emerged. For each approach, numerous methods have been studied in isolation and the currently proposed techniques fall short of the desirable protection issues. One should identify several performance metrics like cryptographic overhead, message overhead and transaction overhead related to different database transactions. One should identify protocols that will reduce this overhead especially for outsourced databases research and define protocols that will reduce the cryptographic overhead for maintaining the integrity of data in such solutions. Many existing protection techniques including encryption, data fragmentation or slicing etc. compromise data integrity. Thus future developments in this issue needs to be considered and propose solutions for protecting the constitution of database from inference attacks. Towards this, one should propose protocols that will randomize queries and results so that no information is leaked to the attackers. One idea to reduce the inference is to make the transaction activity uniform across all queries.

REFERENCES

[1] Carlos Aguilar Melchor, "High-Speed Single-Database PIR Implementation" in 2008.

[2] L. Ballard, S. Kamara, and F. Monrose. Achieving efficient conjunctive keyword searches over encrypted data. In S. Qing, W. Mao, J. López, and G. Wang, editors, ICICS 05, volume 3783 of LNCS, pages 414–426. Springer, Dec. 2005.

[3] M. Bellare, A. Boldyreva, and A. O'Neill. Deterministic and efficiently searchable encryption. In A. Menezes, editor, CRYPTO 2007, volume 4622 of LNCS, pages 535–552. Springer, Aug. 2007.

[4] D. J. Bernstein. Faster square roots in annoying finite fields.http://cr.yp.to/ papers/sqroot.pdf, 2001.

[5] D. Boneh, G. Di Crescenzo, R. Ostrovsky, and G. Persiano. Public key encryption with keyword search. In C. Cachin and J. Camenisch, editors, EUROCRYPT 2004, volume 3027 of LNCS, pages 506–522. Springer, May 2004.

[6] D. Boneh and B. Waters. Conjunctive, subset, and range queries on encrypted data. In S. P. Vadhan, editor, TCC 2007, volume 4392 of LNCS, pages 535–554. Springer, Feb. 2007.

[7] J. W. Byun, D. H. Lee, and J. Lim. Efficient conjunctive keyword search on encrypted data storage system. In EuroPKI, pages 184–196, 2006.

[8] D. Cash, J. Jaeger, S. Jarecki, C. Jutla, H. Krawczyk, M.-C. Roşu, and M. Steiner. Dynamic Searchable Encryption in Very Large Databases: Data Structures and Imple- mentation. 21st Annual Network and Distributed System Security Symposium, NDSS 2014, San Diego, 2014., 2014.

[9] D. Cash, S. Jarecki, C. Jutla, H. Krawczyk, M. Rosu, and M. Steiner. Highly-scalable searchable symmetric encryption with support for boolean queries. Crypto'2013. Cryp- tology ePrint Archive, Report 2013/169, Mar. 2013. http://eprint.iacr.org/2013/ 169.

[10] Y.-C. Chang and M. Mitzenmacher. Privacy preserving keyword searches on remote encrypted data. In J. Ioannidis, A. Keromytis, and M. Yung, editors, ACNS 05, volume 3531 of LNCS, pages 442–455. Springer, June 2005.

[11] M. Chase and S. Kamara. Structured encryption and controlled disclosure. In ASI- ACRYPT 2010, LNCS, pages 577–594. Springer, Dec. 2010.

[12] E. D. Cristofaro, Y. Lu, and G. Tsudik. Efficient techniques for privacy-preserving sharing of sensitive information. In J. M. McCune, B. Balacheff, A. Perrig, A.-R. Sadeghi, A. Sasse, and Y. Beres, editors, TRUST, volume 6740 of Lecture Notes in Computer Science, pages 239–253. Springer, 2011.

[13] R. Curtmola, J. A. Garay, S. Kamara, and R. Ostrovsky. Searchable symmetric en- cryption: improved definitions and efficient constructions. In A. Juels, R. N. Wright, and S. Vimercati, editors, ACM CCS 06, pages 79–88. ACM Press, Oct. / Nov. 2006.

[14] M. J. Freedman, Y. Ishai, B. Pinkas, and O. Reingold. Keyword search and oblivious pseudorandom functions. In J. Kilian, editor, TCC 2005, volume 3378 of LNCS, pages 303–324. Springer, Feb. 2005.

[15] E.-J. Goh. Secure indexes. Cryptology ePrint Archive, Report 2003/216, 2003.http://eprint.iacr.org/.

[16] [16]P. Golle, J. Staddon, and B. R. Waters. Secure conjunctive keyword search over en- crypted data. In M. Jakobsson, M. Yung, and J. Zhou, editors, ACNS 04, volume 3089 of LNCS, pages 31–45. Springer, June 2004.

[17] Y. Huang and I. Goldberg. Outsourced private information retrieval with pricing and access control. Technical Report 2013-11, Centre for Applied Cryptographic Research (CACR), University of Waterloo, Feb. 2013.

[18] IARPA. Security and Privacy Assurance Research (SPAR) Program - BAA, 2011.http://www.iarpa.gov/solicitations_spar.html/.

[19] M. Islam, M. Kuzu, and M. Kantarcioglu. Access pattern disclosure on searchable encryption: Ramification, attack and mitigation. In Proceedings of the Symposium on Network and Distributed Systems Security (NDSS 2012), San Diego, CA, Feb. 2012. Internet Society.

[20] S. Jarecki and X. Liu. Efficient oblivious pseudorandom function with applications to adaptive OT and secure computation of set intersection. In O. Reingold, editor, TCC 2009, volume 5444 of LNCS, pages 577–594. Springer, Mar. 2009.

[21] S. Jarecki and X. Liu. Fast secure computation of set intersection. In SCN 10, LNCS, pages 418–435. Springer, 2010.

[22] S. Kamara and K. Lauter. Cryptographic cloud storage. In Financial Cryptography Workshops, pages 136–149, 2010.

[23] S. Kamara, C. Papamanthou, and T. Roeder. Dynamic searchable symmetric encryp- tion. In Proc. of CCS'2012, 2012.

[24] K. Kurosawa and Y. Ohtaki. UC-secure searchable symmetric encryption. In Financial Cryptography, page 285, 2012.

[25] M. Naor and O. Reingold. Number-theoretic constructions of efficient pseudo-random functions. In 38th FOCS, pages 458–467. IEEE Computer Society Press, Oct. 1997.

[26] [26]V. Pappas, B. Vo, F. Krell, S. G. Choi, V. Kolesnikov, A. Keromytis, and T. Malkin.

[27] Blind Seer: A Scalable Private DBMS. Manuscript, 2013.

[28] E. Shi, J. Bethencourt, H. T.-H. Chan, D. X. Song, and A. Perrig. Multi-dimensional range query over encrypted data. In 2007 IEEE Symposium on Security and Privacy, pages 350–364. IEEE Computer Society Press, May 2007.

[29] D. X. Song, D. Wagner, and A. Perrig. Practical techniques for searches on encrypted data. In 2000 IEEE Symposium on Security and Privacy, pages 44–55. IEEE Computer Society Press, May 2000.

[30] P. van Liesdonk, S. Sedhi, J. Doumen, P. H. Hartel, and W. Jonker. Computation- ally efficient searchable symmetric encryption. In Proc. Workshop on Secure Data Management (SDM), pages 87–100, 2010.

[31] B. R. Waters, D. Balfanz, G. Durfee, and D. K. Smetters. Building an encrypted and searchable audit log. In NDSS 2004. The Internet Society, Feb. 2004.

[32] WSJ. U.S. Terrorism Agency to Tap a Vast Database of Citizens. Wall Street Journal 12/13/12. http://alturl.com/ot72x

[33] Carlos Aguilar Melchor, Philippe Gaborit "A Fast Private Information Retrieval Protocol" in ISIT 2008, Toronto, Canada, July 6 - 11, 2008.

[34] Andrew Clarke, Eric Pardede "Outsourced XMLDatabase:Query Assurance Optimization," 24th IEEE International Conference on Advanced Information Networking and Applications, ICEGOV2010,

[35] Carlos Aguilar Melchor, Philippe Gaborit, "A Fast Private Information Retrieval Protocol," ISIT 2, pp.1848-1852, July 6 - 11, 2008.

[36] D. Abril, G. Navarro-Arribas, V. Torra, "Towards Privacy Preserving Information Retrieval Through Semantic Microaggregation,"IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, 2010.

[37] Ian Goldberg "Improving the Robustness of Private Information Retrieval" in IEEE Symposium on Security and Privacy(SP'07) 2007.

[38] Tang H, Cui Y, Guan C, Wu J, Weng J, Ren K (2016) Enabling ciphertext deduplication for secure cloud storage and access control. In: Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security, AsiaCCS'16, pp 59–70

[39] Turner V, Gantz J, Reinsel D, Minton S (2014) The digital universe of opportunities: rich data and the increasing value of the internet of things. IDC White Paper, April 2014

[40] Wang J, Chen X, Huang X, You I, Xiang Y (2015) Verifiable auditing for outsourced database in cloud computing. IEEE Trans Comput 64(11):3293–3303.