Volume 5, Issue 4, 2015 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

A Survey on Homomorphic and Searchable Security Algorithms for Cloud Computing

Prasanna B T*
Department of ISE, EPCET,
Bengaluru, India

Dr. C B Akki
Department of ISE, SJBIT,
Bengaluru, India

Abstract*— Cloud computing is a new trend in computer technology which addresses the availability, efficiency and scalability of data with less cost. Cloud is a virtualized distributed network that enables the cloud providers to provide cloud services to their clients. Trust need to be established between cloud provider and their clients to make cloud a successful technology. This calls for a refined security schemes that decide the success of cloud technology. Some of the groups have been formed like Cloud Security Alliance, Open Cloud Consortium to concentrate on research in cloud in general and cloud security in particular. In this paper cloud security related challenges were surveyed and emphasis is given to survey various homomorphic and searchable encryption schemes along with their pros and cons. This research study would help to identify suitable security algorithms for cloud computing.

Keywords*— Cloud computing, challenges, homomorphic encryption, searchable encryption.

I. INTRODUCTION

Globalization resulted in increased requirements for the availability, scalability and efficiency of data at reduced cost. Cloud computing is one such technology which achieves all these factors. Cloud is a distributed network based on virtualization technology that enables the service providers to provide services to the consumers. The risk associated with trust between service providers and service consumers is a major issue in cloud computing. This calls for sophisticated security schemes that decide the success of cloud technology.

II. CLOUD COMPUTING

In 1961, John McCarthy visualized that someday computer technology may evolve to the point where computation power could operate on the utility business model, like electricity or water. The idea became popular, but died down in the early '70s because the hardware infrastructure needed for cloud computing was nowhere in sight yet [1]. Netcentric in 1997 for the first time used the term cloud computing. Later in, 2001, John Markoff used the phrase cloud in one of his articles written for New York Times. In 2006 the phrase cloud computing became popular when Schmidt Google and Amazon used the same. Gartner [2], one of the leading information technology research and advisory companies, says cloud computing will be as influential as e-business.

Mell and Grace from National Institute of Standards and Technology (NIST) defined the cloud computing as a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction [3]. In cloud computing the software and information are provided on demand through internet like electricity to our homes through transmission and distribution system. Customers do not own the hardware instead they get the same on rental basis from third party who maintains the cloud. Hence expenses can be reduced. In cloud computing, customer pays only for the resources that they use and they consume these resources as a service. Depending on the service being provided by the cloud, there are three delivery models: Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS). Depending on the ownership and usage of cloud services there are different deployment models. They are private, public, community and hybrid clouds.

Some of the challenges of cloud that are mentioned in the literature include 1. Data Loss-Customers are responsible for the security of their own data. Here, if data is lost or stolen the customer is in deep trouble. 2. Account hijacking-Since no native APIs are used for login and anyone can easily register as cloud service user. Hence chances of hijacking ones account are very high. 3. Control over the process – In cloud computing the user have very less or no control over the services. 4. Insider attacks by cloud service provider-It may be possible that a fraudulent employee may do the fishing and steal the data. 5. Legal aspects-In case of data loss the user may suffer if there is no Service Level Agreement (SLA), the loss will be of user, because he is not able to put claims against the cloud service provider. 6.Jurisdiction- If the cloud user and cloud providers are from different countries having different IT security laws and if they have some cloud security related disputes then it might cost more to get legal services, than the advantage of cost effectiveness of cloud during any cloud security related disputes between them[5],[6].

A research conducted by the IDC Enterprise Panel (NIST, 2009) [4], concluded that the primary concern which Information Technology personal at various levels expressed are Security, Availability, Integration and Cost. Since

© 2015, IJARCSSE All Rights Reserved

cloud is based on trust model between cloud user and cloud provider, security is one of the major challenges that cloud computing research community needs to consider and explore.

III. SECURITY CHALLENGES IN CLOUD

Cloud security is the major research area for a number of reasons. Firstly, traditional cryptographic methods used for the purpose of data security cannot be used directly, because of user's loss of control of data under cloud. Therefore the challenge of verifying correctness of data storage in the cloud is a critical factor. Secondly, anyone with access permission can manipulate cloud data. Therefore we need to check for integrity of the data. Lastly, the deployment of cloud computing is virtual means it is powered by data centres running in a simultaneous cooperated and distributed manner. The Individual user's data is stored redundantly in multiple physical locations. Therefore, distributed protocols for storage correctness assurance will be of most importance in achieving a robust and secure cloud data storage system in the real world [7].

The Cloud Security Alliance (CSA) [36] in 2010 did a research on the threats facing cloud computing and it identified the major seven threats they are Abuse and Nefarious Use of Cloud Computing, Insecure Application Programming Interfaces, Malicious Insiders, Shared Technology Vulnerabilities, Data Loss/Leakage, Account, Service & Traffic Hijacking and Unknown Risk Profile. The same organization CSA in 2013 also identified the following security threats they are Data theft, Loss of data, Service traffic hijacking, Insecure interfaces and API, Denial of service, Malicious insiders, Use of cloud resources by hackers, Lack of foresight and Adjacent vulnerability.

Each deployment model has its own security issues. Since private cloud is in control within an organization the security attacks can be controlled, whereas public cloud is public in nature, hence is more vulnerable to security attacks. Several researchers studied and discussed security issues in cloud. Jon Marler [10], discussed security issues in public cloud like security against hackers, security against resource contention and strategies for private cloud to address these issues by allowing business to control access to private cloud and by avoiding shared infrastructure to protect from Denial of Service (DoS) attack respectively. Kui Ren et al. [7],[54] surveyed some of the security challenges in public cloud like data service outsourcing security, computation outsourcing security, access control, trustworthy service metering, multi tenancy security and privacy etc., and also discussed the cryptographic scheme of homomorphic token with distributed verification of erasure coded data to protect from malicious data modification and server colluding attacks. S Subashini et al [11] discussed the security threats based on service delivery models for SaaS, PaaS and IaaS. The authors mainly concentrated on security issues in SaaS which include the security elements like data security, network security, data locality, data integrity, data segregation, data access, authentication and authorization, data confidentiality, web application security, data breaches, virtualization vulnerability, availability, backup, identity management and sign on process etc. Viega [8], foresees that data and code residing in cloud computing environments will become more tempting targets to hackers. Armbrust et al. [9], raised concerns about availability and confidentiality of data in cloud.

Rohit Bhadauria, et al [12], discussed security issues in different deployment models and different levels of layers. They have also discussed different security models and the schemes used, listed in the Table 1 below:

Table 1. Security models and schemes

Models	Schemes
Data storage security	Homomorphic Algorithms
Identity of user's safety	Active bundle schemes
Trusted Model in cross cloud	Third party trust agent
Virtualized trust management	DHT based Overlay networks
Secure Virtualization	Advanced Cloud Protection System
Border gate way protocol	Pretty Good Border Gateway Protocol

S.Sudha et.al [56], discussed the security issues in different levels like network, host, application and data of cloud user. They also discussed different security models, the proposed approach, their strength and limitation along with security issues in different delivery models. The following Table 2 lists the delivery models and corresponding security issues. The need of security and privacy of data increased after the introduction of public key cryptography in 1976 by Diffie and Hellman [55].

Table 2. Security issues in delivery models

Delivery Models	Security Issues
Infrastructure as a Service(IaaS)	Virtualization vulnerabilities
Platform as a Service(PaaS)	Access, Authorization, Distributed system, Storage and data security issues
Software as a Service(SaaS)	Access control, Identities for accessing the enterprise applications, Integrity

© 2015, IJARCSSE All Rights Reserved

IV. HOMOMORPHIC ENCRYPTION SCHEMES

To secure and maintain the confidentiality of data from insiders attack in an organization one needs to secure the encryption algorithm itself. Shannon [38] for the first time discussed about the security of encryption algorithm. Rivest et al. [37] in 1978 introduced the concept of homomorphism that allows certain algebraic operations on two plaintexts to be carried out on their corresponding cipher texts, without any intermediate decryptions.

Given two encryptions c1 = Epk(m1) and c2 = Epk(m2) of messages m1 and m2 with public key pk, a Homomorphic Encryption scheme allows to compute an encryption Epk (m1 op m2) without decrypting either c1 or c2. Here op denotes some arbitrary operation.

For example RSA public-key encryption scheme produces ciphertexts of the form $m^e \bmod N$, where m is the message, e is the public key and N is the product of two primes. Given two encryptions $c1 = (m1)^e \bmod N$ and $c2 = (m2)^e \bmod N$, encryption of m1 and m2 can be computed by $c1 \times c2 = (m1)^e \bmod N \times (m2)^e \bmod N = (m1 \times m2)^e \bmod N$.

Let us take a numerical example by considering two primes p and q i.e. p= 7, q= 17, then product of 2 primes is $N=p \times q= 119$. If totient value $\phi(N)$ is 96(i.e. $(p-1)\times(q-1)$) and public key $e= 5(1< e < \phi(N))$, then $\gcd(\phi(119), 5)= 1$ and private key $d \equiv e^{-1} \bmod \phi(N)= 77$(By Extended Euclid's algorithm). Let the input message be m1= 22 and m2= 19, then the encryption of m1 is given as $c1 \equiv 225 \bmod 119= 99$ and the encryption of m2 is given as: $c2 \equiv 195 \bmod 119= 66$.Then $c3=c1 \times c2 \equiv 99 \times 66 \bmod 119= 108$. Applying the decryption algorithm over c3 results in 61, which is equivalent to the multiplication of the two plaintexts i.e. $m3=m1 \times m2 \equiv 22 \times 19 \bmod 119= 61$. Hence RSA supports homomorphic operation of multiplication modulo 119 (i.e. multiplication modulo N).

Variants of homomorphic algorithms were designed by researchers to do computation on encrypted data. All these can be broadly classified into Partial Homomorphic Encryption (PHE) scheme and Fully Homomorphic Encryption (FHE) scheme [63].

A. Partial Homomorphic Encryption (PHE)

Partial homomorphic scheme properly perform only a limited number of operations for example either addition or multiplication on encrypted data due to an inability to properly decrypt after a certain threshold of noise introduced by the operations. Some of the known partial homomorphic cryptosystems are Unpadded RSA, ElGamal, Goldwasser-Micali, Benaloh, Paillier, Okamato-Uchiyama, Naccache-Stern, Damgard-Jurik, and Boneh-Goh-Nissim.

Goldwasser-Micali (GM) scheme [39], in this scheme a Quadratic Residuosity problem which allows bit wise exclusive or of homomorphic evaluation is used to provide security. This scheme uses computations modulo n = p X q, a product of two large primes. Product and a square are used in encryption, whereas decryption requires exponentiation. Some drawbacks of this scheme are its input which consists of a single bit. Encrypting k bits leads to a high cost. Other problem is expanding, where a single bit of plaintext is encrypted in an integer modulo n, which in turn increases the size of the ciphertext unboundedly.

Benaloh's scheme [40], this scheme is derived from GM scheme, where encryption is similar as in GM scheme. But decryption is more complex. Consider the input and output sizes as l(k) and l(n) bits respectively. The expansion value i.e., l(n)/l(k) in this approach is less than that achieved in GM. By taking the small value of k we can increase the efficiency of this scheme.

Naccache-Stern scheme [41], this scheme is an improved version of Benaloh's scheme. It uses value of k (input) greater than that used in the Benaloh's scheme. The encryption step is same as in Benaloh's scheme. But the cost of decryption is less. The value of expansion is 4, which is same as that in Benaloh's scheme.

Okamoto-Uchiyama scheme [42], by taking, n = p2 X q (p and q are large prime), and group G, this scheme achieves k=p. One of the advantages is that this scheme reports the security equivalent to factorization of n. For this scheme expansion value is 3. Even this scheme is hacked in JQY attacks and chosen valid/invalid attacks as reported in [62].

Paillier scheme [43], this is one of the popular PHE methods. It is an improvement over the earlier schemes in the sense that it is able to decrease the value of expansion from 3 to 2. This method uses n=p X q with GCD (n, Φ (n)) =1. Encryption cost is not high, but decryption is costlier since it needs one exponential modulo n2 to the power λ (n) and a multiplication modulo n. Chinese remainder theorem published by a mathematician Sun Tzu explains the cost effective efficient use of decryption method. In 2002, Cramer and Shoup proposed an advanced efficient scheme based on Paillier scheme to protect cryptosystems against chosen ciphertext attacks. This method is an extension of the ElGamal cryptosystem.

Damgard-Jurik [44], proposed a generalization of Paillier scheme. Here expansion value sometimes reaches value 1. But it is computationally more intensive compared to Paillier scheme. If we want to encrypt or decrypt n blocks l(k) bits, executing Paillier's scheme n times is less expensive than executing Damgard-Jurik's scheme.

Galbraith scheme [45], it is an adaptation of elliptic curves cryptosystem (ECC) method for homomorphic encryption based on one way trapdoor function. This scheme uses the algebraic structure of elliptic curves over finite fields. The advantage of ECC is a smaller key size, reducing storage and transmission requirements. The computational cost is very high both in key generation and decryption process. Koyama et al. proposes an elliptic curve RSA based scheme, which found to be not semantically secure.

Boneh, Goh and Nissim scheme [31], this scheme is based on bilinear pairings on elliptic curves and performs many addition operations and a single multiplication operation on plaintexts. This scheme requires more message space to perform discrete logarithms during encryption.

© 2015, IJARCSSE All Rights Reserved

Castagno's scheme [46], this scheme uses quadratic field's quotations to improve the performance of PHE's. The scheme achieves an expansion value of 3.

Yin Hu [57] did a comparative study of different PHE schemes as listed in the following Table 3.

Table 3. Comparison of PHE Schemes

Scheme	Homomorphism	Computation	Benefits	Limitations
RSA	Multiplicative	Mod. Exp. in Zpq	The basic simple homomorphic scheme	Multiplicative variant is not semantically secure
ElGamal	Multiplicative	Mod. Exp. in GF (p)	Simple, Natural and Efficient	Relay on exponentiation of operations which affect parallel operation
Goldwasser Micali	XOR	Mod. Exp. in Zpq	Uses Quadratic Residuosity using XOR for encryption which provides high security	Decryption process requires exponentiation. High expansion value
Benaloh	Additive	Mod. Exp. in Zpq	Expansion value is less compared to known homomorphic algorithms	Decryption process is complex
Paillier	Additive	Mod. Exp. in Z (pq)2	Encryption cost is less	Decryption cost is more
Paillier ECC variations	Additive	Scalar-point mult. in elliptic curves	Smaller key size, reduced storage and transmission requirements	High computational cost both in key generation and decryption process
Naccache-Stern	Additive	Mod. Exp. in Zpq	Cost of decryption is less	
Kawachi-Tanaka-Xagawa	Additive	Lattice Algebra	Ciphertext having the size equal to plaintext	Computation cost is high
Okamoto-Uchiyama	Additive	Mod. Exp. in Zp2q	Security equivalent to factorization of n	Expansion value is high. Ciphertext attacks are reported
Boneh-Goh-Nissim	2-DNF formulas	Mod. Exp. in Z (p. q)2, Bilinear Map	Semantically secured	Computationally expensive
Melchor-Gaborit-Herranz	d-op. mult	Lattice algebra	d degree polynomial method, and uses shortest vector problem for encryption	Ciphertext grows exponentially

B. Fully Homomorphic Encryption (FHE)

FHE is a cryptosystem that performs both addition and subtraction operations on encrypted data without affecting the ring structure of plaintexts. Many researchers have done extensive work towards making FHE practical [14], [64], [47], [65], [50].

In 2009, Gentry [64], in his PhD thesis discussed about FHE. The FHE schemes are used to perform arbitrary complex operations on encrypted data. Later, Gentry's scheme became the blueprint for the researchers working on improving the efficiency of FHE. Gentry's scheme mainly consists of three modules: Ideal lattices based SomeWhat Homomorphic Encryption (SWHE), a bootstrapping module and a module that transfers SWHE to FHE using bootstrapping. Gentry's scheme is based on hardness approximating problems within a sub exponential factor. It uses mathematical objects like ideals in various rings and sparse subset sum assumptions to simplify the complex decryption

© 2015, IJARCSSE All Rights Reserved

circuit. It uses the parameter per gate evaluation time (the ratio of time for homomorphic evaluation of the circuit to the time for evaluating circuit on plaintext) to define the efficiency.

Smart and Vercauteren [47], presented a FHE scheme based on elementary theory of algebraic number fields (a finite extension of the field of rational numbers Q). To improve the efficiency, this scheme uses smaller key and ciphertext. One of the drawbacks of this scheme is that it takes longer time to generate the keys.

Marten van Dijk, Gentry, Halevi and Vaikuntanathan [65] (DGHV scheme), discusses simpler SWHE by using the elementary modular arithmetic methods like the addition and multiplication over the integers and approximate GCD. The efficiency of this scheme is found to be low due to the difficulty in preserving the hardness of the approximate GCD problem.

Gentry and Haveli [15], continued the work done by Smart et.al. [47], and implemented efficient key generation module (without the requirement of determinant of lattice to be prime) in bootstrapping process. Reduced time of key generation and simplified decryption procedure helped in reducing the per-gate-evaluation time.

Ogura et al. [52], discussed a method of FHE to control the bound of the circuit depth. The relation between circuit depth and Eigen values of a basis of a lattice is used to reduce the per-gate-evaluation time. However the increased length of ciphertext makes this scheme inefficient.

Stehle and Steinfield [66], improved Gentry's FHE based on ideal lattices [13] and proposed Faster FHE. They implemented probabilistic decryption algorithm with an algebraic circuit (Algebraic method for analysis and synthesis of logic circuits) of low multiplicative degree. Hardness assumption of the security is stronger compared to Gentry's method.

Chunsheng et.al [53], proposed an improved FHE by removing the assumption of sparse subset problem from the FHE proposed by Smart and Vercauteren [47]. The self-loop bootstrappable technique allows the security of the scheme to depend only on the hardness of the polynomial coset problem. The self-loop bootstrappable technique uses three mathematical structures: (i) Hardness of factoring integer problem

(Decomposing composite numbers into smaller non trivial divisors), (ii) Solving Diophantine equation problem (Polynomial equation in two or more unknowns such that unknowns take integer values and are searched), and (iii) Finding approximate greatest common divisor problem (Largest positive integer that divides a number without remainder).

Zvika Brakerski, Vinod Vaikuntanathan [51], proposed a SWHE scheme based on LWE (Learn With Errors). To improve the efficiency of the scheme a re-linearization technique and a new dimension modulus technique are used to reduce ciphertext size and to simplify complex decryption circuit respectively.

Vadim Lyubashevsky et.al [67], [48], proposed a FHE scheme based on algebraic variant of LWE called Learning with Errors over Rings (RLWE). To improve the efficiency, RLWE is applied to solve a short vector problem on ideal lattices. Reduced size of key and ciphertext helps in lowering per-gate-evaluation time.

Zvika Brakerski, Craig Gentry, Vinod Vaikuntanathan [49], have proposed a scheme where per-gate-evaluation time is reduced by evaluating L-level (depth of the circuit being evaluated) arithmetic circuits without using bootstrapping. The security hardness depends on RLWE and Level of the circuit (L).

Jean-Sebastien Coron et al. [68], presented a new scheme called Batch FHE over the Integers based on DGHV scheme [65]. To reduce the per-gate-evaluation time, a vector of encrypted plaintext bits are processed homomorphically as a single ciphertext.

Kurt Rohloff et.al [69], proposed a scalable FHE scheme that is based on NTRU (Nth degree Truncated polynomial Ring [70]). The scheme uses bootstrapping technique with simplified power-of-2 rings and double-CRT representations of ciphertext, which improves the efficiency of parallel computations on encrypted data.

Based on versatility, speed and ciphertext size the following Table 4 compares PHE and FHE schemes.

Table 4: PHE vs FHE

Homomorphic Encryption	**Versatility**	**Speed**	**Ciphertext Size**
PHE	Low	Fast	Small
FHE	High	Slow	Large

Variants of above discussed Homomorphic Cryptosystems could be used to build searchable encryption schemes with perfect privacy of client data in cloud.

V. SEARCHABLE ENCRYPTION

In cloud, the primary concern is of maintaining both confidentiality and privacy of owner's data from untrusted users. HE schemes are surveyed so far to address the issue of confidentiality. SE technology can be used to deal privacy of data which are addressed now.

Numerous SE algorithms have been proposed by many researchers. Song et al. [22], first introduced the concept of SE. The scheme uses two layered symmetric encryption method to encrypt each word independently in the file. Goh et al. [23], proposed an efficient search technique based on bloom filters, which uses separate index file for each document. One of the drawbacks of the scheme is space inefficiency. Chang et al. [24], proposed a scheme based on Searchable

© 2015, IJARCSSE All Rights Reserved

Symmetric Encryption (SSE) method. But adaptive queries from the adversary may lead to compromise of privacy. This inefficiency was addressed by Curtmola et al. [21], where a hash based indexing is used for entire file. The scheme suffers from hiding the access patterns of any particular user. Boneh et al. [31], presented the first public-key based searchable encryption scheme. In their construction, anyone with the public key can write data to the cloud but only authorized users with the private key can search. All these schemes mentioned above search only one keyword, whereas the conjunctive keyword search scheme proposed by P. Golle et al. [29], and Y. H. Hwang et al. [16], is used to search multiple keyword which is suitable for cloud type of technology. Li. et al. [17], proposed fuzzy keyword search over encrypted cloud data. The scheme tries to match exact keyword in a file. If not found then a close match of the keyword is done and corresponding file is returned to the user. Jianhua et al. [58], proposed the conjunctive fuzzy keyword search, which is an improved method over conjunctive and fuzzy keyword search methods.

Song et al. [22], [21], proposed the first SSE scheme that works on symmetric key encrypted data. The scheme doesn't use indexing for keyword search. Searching will be done in linear time. The drawback is that the scheme is vulnerable to statistical attacks and the keyword search reveals the exact match in the document to the cloud provider.

Cao et al. [18], proposed an efficient SSE scheme based on ranked keyword search technique. Multi-keyword queries are searched linearly in the database and the top-k most relevant documents are returned by the server. The advantage of scheme lies in searching of documents at low cost. On the other hand symmetric encryption has an inherent key management problem.

Goh [23], proposed a method where indexes are secured by using bloom filters and pseudo-random functions as hash functions. Here each document has its own index file. In order to differentiate a word's hash values in different files, every word in the document is first hashed with the master key and the output is hashed again with the document id to get code word. This code word and random uniform distribution of one's (1's) are inserted into the bloom filter. The user sends the code word to the server and checks the code word bits to get the required data based on keyword.

Chang and Mitzenmacher [24], proposed a method where a binary index array for each document is created using all possible predefined dictionary words in the database. Output of pseudorandom function is used by the user to mask bits in the index array of each file. Short seeds are used by the user to recover selective parts of the index and by using bit masked string, server retrieves the corresponding data.

Curtmola et al. [21], proposed a method where an inverted index (implemented using linked list) having document identifiers is maintained for each keyword. Every node in the list stores information about the position and the decryption key of the next node. The nodes from all inverted indexes are encrypted with random keys and are randomly inserted into an array. With this, by knowing position and decryption key of the first node of an inverted index, it is possible to find all documents which include the corresponding keyword. To improve the efficiency of the above scheme, top-k single keyword retrieval schemes are proposed in the literature [20].

Wang et al. [17], proposed a ranked key word search which builds an inverted index for every keyword in the database. The actual scores are encrypted with a modified Order-Preserving Symmetric Encryption (OPSE) [25] scheme to have security, where the numeric ordering of the plaintexts is preserved in the ciphertexts. During query processing, the trapdoor sent by the user allows the server to decrypt the corresponding entry in the inverted index. The server uses encrypted score comparisons to retrieve the top k results.

P. Naresh, et al [20], have discussed Ranked Searchable Symmetric Encryption (RSSE) scheme where rank search is built over the SSE cryptographic primitive. This technique generates public/private key pair and index file containing keywords from files that need to be searched. The encrypted file, the index file and frequency based relevance score are put in to the cloud. Upon the request by the user a trapdoor is generated and search will be carried on the encrypted files based on ids and relevance scores.

Cao et al. [18], proposed the multi-keyword rank based search. The scheme queries by using inner product similarity as the scoring function. Similar to the work of Chang and Mitzenmacher [24], they use a predefined dictionary of all possible words in the database, and construct an encrypted binary array for each document. To compute the similarity scores from the encrypted indexes, they employ the secure k Nearest Neighbour (kNN) computation method proposed by W.K.Wong et al. [26].

Public-key encryption with keyword search (PEKS) method searches public key encrypted data in cloud. PEKS and other methods based on PEKS [43], [44], are very restrictive in the types of queries that can be performed and none of them uses ranked keyword search method.

Boneh et al. [27], introduced a method called Public Key Encryption with Keyword Search (PEKS). In this approach, the public key encryption method is used to encrypt the files as well as keywords selected by the sender. The encrypted files and keywords are subsequently sent to the server for storage. The cloud provider creates a trapdoor for the keyword and sends it to the server. After testing the trapdoor with each encrypted keyword the server returns the keyword matched file(s) to the user or provider.

Baek et al. [28], enhanced the performance of PEKS framework by addressing its major limitations like (i) preventing the server from reusing the trapdoors, (ii) eliminating the need for a secure authenticated channel between the owner and the server, and (iii) adding multi-keyword search capabilities. But the enhanced scheme also fails in supporting ranked keyword search.

Boneh et al. [30], proposed a searchable encryption scheme that provides perfect privacy. Keyword information is stored in encrypted bloom filters and uses homomorphic encryption scheme of Boneh et al. [31], to allow the senders to modify the index in a secure manner. Although the scheme is computationally expensive, it maintains privacy by hiding user data from the server.

© 2015, IJARCSSE All Rights Reserved

Ning Caoy et al. [18], proposed a Boolean search based single keyword searchable encryption method. The scheme builds encrypted searchable index to search for a keyword in a database. In this scheme only user with public key can access and write data to the cloud, but only authorized users with private key can search. One of the drawbacks of this approach is post processing of every file to get exact match and unnecessary network traffic. Conjunctive keyword search techniques over encrypted data have been proposed for this problem. Conjunctive keyword search returns "all-or-nothing", which means it only returns those documents in which all the keywords specified by the search query appear.

P. Golle et al. [29], proposed a conjunction keyword search method. Here conjunction of keywords is used for searching. The scheme uses keyword rank to retrieve most efficient data. Wildcard based method and gram based method are used for constructing fuzzy keyword sets. To reduce the storage overhead, fuzzy keywords are stored in multi way tree data structure by using symbol based trie-traverse method.

Cong Wang et al [17], proposed a scheme that finds the most relevant data based on rank. Since keyword search is done based on rank, only relevant data is accessed and network traffic will be reduced. Keyword relevant information is not leaked ensuring privacy of data stored.

Kiruthigapriya Sengoden et al. [34], proposed a concept based search that allows users to selectively retrieve files from server. This searchable scheme uses Boolean keyword search, which will combine words and phrases using the words AND, OR, NOT operators. The concept related keywords are used for search. So the combination of both keyword search and concept search produces the relevant search result which greatly improves the efficiency of search. The increased network traffic and inefficient data retrieval are some of the drawbacks that lead to inefficiency of the scheme.

Nigduo peng et al [35], proposed a content based search. The scheme divides the document into smaller snippets and uses query biased snippet having queried keywords and index to search within each snippet. The index stores the information about the keyword frequency in each snippet, which enables the server to dynamically calculate the best snippet for the user when queried by multiple keywords.

Jin Li et al. [19], proposed fuzzy keyword search based on edit distance that is used to measure keywords. A set of fuzzy keyword distances is created. For keyword measure, edit distance uses straight forward or wild card based approaches. In straight forward approach indexing is constructed by calculating edit distance of all the forms of keywords. Trapdoors are shared between user and the owner. While retrieving, file user computes the trapdoor. Based on the request, server matches with index table and returns all potential identifiers or keywords.

Multi keyword Ranked Search over Encrypted Cloud data (MRSE) Scheme uses homomorphic encryption and vector space to support multi keyword top-k retrieval. In MRSE 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.

S.Buyrukbilen et al [32], introduce the first method that provides ranked results from multi-keyword searches on public-key encrypted data. By avoiding a linear scan of the documents and by parallelizing the computations to the possible extent, this method reduces the computational complexity of public key cryptosystem. The scheme encrypts keyword information of each document in a bloom filter [33], and hierarchically aggregate (using homomorphic encryption) the individual indexes into a tree structure. Client will do the query processing, and traverse the tree in best-first manner. The query is hidden from the server or cloud provider by using an efficient private information retrieval (PIR) protocol [34]. In this method the indexes are split into multiple chunks, and use several CPUs in parallel to execute the user queries efficiently.

Wenhai Sun et al. [59], [60] proposed a MRSE scheme based on similarity based ranking. Here search index is created on the basis of term frequency and vector space. Search index is used for multi keyword search and ranking the search result. Search efficiency is improved by applying tree structure on index.

Zhihua Xia, Li Chen, Xingming Sun, and Jin Wang [61], proposed a scheme based on MRSE. The scheme uses latent semantic analysis to reveal relationship between terms and documents. K - Nearest Neighbors method is used to achieve secure search function.

There are several groups interested in developing standards for cloud security. The Cloud Security Alliance (CSA) an active group in cloud security [36] is gathering information from solution providers, non-profit organizations and individuals to enter into discussion about the current and future best practices for information assurance in the cloud. Cloud Standards, a wiki page, is used to document the activities of the various standards under development by other groups (SDOs). The Open Web Application Security Project (OWASP), focused on improving the security of application software, lists top ten vulnerabilities that are used by attacker to cause harm to the stakeholders of an application. Open Grid Forum (OGF) is an open community committed to driving the rapid evolution and adoption of applied distributed computing. It publishes documents containing security issues and solutions in the area of distributed computing. The Open Cloud Consortium (OCC) is a group of researchers from universities and IT companies. The main objective of OCC is to look into the cloud computing challenges and related security issues. This is a relatively new group formed in the mid-2008. The OCC is working on developing framework, standards, benchmarks, reference implementations and managing a test bed i.e., open cloud test bed and also sponsoring workshops and other events related to cloud computing.

VI. CONCLUSION

Cloud computing is a collection of communicating nodes distributed geographically across physical locations. Here the need for protecting or securing the data from unauthorized users is a key issue i.e., the security is a major challenge in cloud computing. Security deficiencies need to be identified and suitable efficient methods are required for the success of cloud technology. Some of the research organizations like Cloud Security Alliance, Open Cloud Consortium were

© 2015, IJARCSSE All Rights Reserved

concentrating their research on cloud security in particular and cloud computing in general. The paper apart from addressing the available methods to mitigate the security challenges also focuses on variants of homomorphic and searchable encryption methods available in literature and studying their performance.

REFERENCES

[1] Cary Whiaker. Cloud Computing – Storm Clouds or is it Smooth Flying? , East Carolina University, 2011.

[2] Bodkin J, Seven Cloud Computing Security Risks, Availableat http://www.gartner.com/DisplayDocument?id=685308. And http://www.gartner.com/it/page.jsp?id=707508.

[3] Mel, Peter and Tim Grace. Draft NIST Working Definition of Cloud Computing, Available at http://csrc.nist.gov/groups/SNS/clouddcomputing/cloud-def-v15.doc, on August 28, 2009.

[4] Making Virtual Machines Cloud-Ready, a trend micro white paper, August 2009. Available at http://www.securecloud.com/cloud-content/us/pdfs/business/white-papers/wp_cloudsecurity-unlock-opportunities.pdf.

[5] Bunya, R., Broberg, J.I, Brandic, Venugopal, S., and Yeo, C.S. Cloud Computing and Emerging IT Platforms: Vision, Hype, and Reality for delivering Computing as the 5th Utility, Future Generation Computer Systems 25, 599–616, 2009.

[6] Cloud Security Alliance. Top Ten Threats, Available at www.cloudsecurityalliance.org/topthreats/csathreats.v1.0.pdf.

[7] Cong Wang, Kui Ren, and Qian Wang. Ensuring Data Storage Security in Cloud Computing, Worcester Polytechnic Institute, Massachusetts, April 19, 2009.

[8] Viega, J. Cloud Computing and the Common Man, IEEE Computer Society 42(8): 106-108.

[9] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. H. Katz, A. Konwinski, G. Lee, D. A. Patterson, A.Rabkin, I. Stoica, and M. Zaharia. Above the clouds: A Berkeley view of cloud computing, University of California, Berkeley, Tech. Rep. USB- EECS-2009- 28, Feb 2009.

[10] Jon Marler. Securing the Cloud: Addressing Cloud Computing Security Concerns with Private Cloud, Rackspace Knowledge Centre, March 27, 2011, Article Id: 1638.http://www.rackspace.com/knowledge_center/private-cloud/securing-the-cloud-addressing-cloud computingsecurity-concerns-with- private- cloud.

[11] Jon Marler. Securing the Cloud: Addressing Cloud Computing Security Concerns with Private Cloud, Rackspace Knowledge Centre, March 27, 2011, Article Id: 1638. Available http://www.rackspace.com/knowledge_center/private-cloud/securing- the-cloud-addressing-cloud-computingsecurity-concerns-with- private- cloud.

[12] Rohit Bhadauria, and Sugata Sanyal. A survey on Security Issues in Cloud Computing, International Journal of Computer Applications. Volume 47- Number 18, 2012.

[13] Craig Gentry, Fully homomorphic encryption using ideal lattices, In Michael Mitzenmacher, editor, STOC, pages 169-178, ACM, 2009.

[14] Craig Gentry. Toward basing fully homomorphic encryption on worst-case hardness, In Tal Rabin, editor, CRYPTO, Volume 6223 of Lecture Notes in Computer Science, pages 116-137. Springer, 2010.

[15] Craig Gentry and Shai Halevi. Implementing gentry's fully-homomorphic encryption scheme, In EUROCRYPT, 2011.

[16] Y. H. Hwang and P. J. Lee. Public Key Encryption with Conjunctive Keyword Search and Its Extension to a Multi-user system, in Proc. of Pairing'07, 2007, pp. 31–45.

[17] C. Wang, N. Cao, J. Li, K. Ren, and W. Lou. Secure ranked keyword search over encrypted cloud data, in IEEE ICDCS, 2010, pp. 253–262.

[18] Ning Caoy, Cong Wangz, Ming Liy, Kui Renz, and Wenjing Louy. Privacy-Preserving Multi-keyword Ranked Search over Encrypted Cloud Data", INFOCOM, 2011 Proceedings IEEE.

[19] Jin Li, Con Wang, Kui Ren, Ning Cao, Qian Wang, and Wenjing Lou .Fuzzy Keyword Search over Encrypted Data in Cloud Computing, INFOCOM, 2010 Proceedings IEEE.

[20] P. Naresh, K. Pavan kumar, and D. K. Shareef. Implementation of Secure Ranked Keyword Search by Using RSSE, International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181 Vol. 2 Issue 3, March – 2013.

[21] R. Curtmola, J. A. Garay, S. Kamara, and R. Ostrovsky. Searchable symmetric encryption: improved definitions and efficient constructions, in ACM CCS, 2006, pp. 79–88.

[22] D. X. Song, D.Wagner, and A. Perrig. Practical techniques for searches on encrypted data, in IEEE S&P, 2000, pp. 44–55.

[23] E. J. Goh. Secure indexes. Technical Report 216, IACR, 2003.

[24] Y. Chang and M. Mitzenmacher. Privacy preserving keyword searches on remote encrypted data. In J. Ioannidis, A. D. Keromytis, and M. Yung, editors, Proceedings of the Third international conference on Applied Cryptography and Network Security, volume 3531 of LNCS, pages 442–455. Springer, 2005.

[25] A. Boldyreva, N. Chenette, Y. Lee, and A. O'Neill. Order-preserving symmetric encryption, in EUROCRYPT, 2009, pp. 224– 241.

[26] W. K. Wong, D. W.L. Cheung, B. Kao, and N. Mamoulis. Secure kNN computation on encrypted databases, in ACM SIGMOD, 2009, pp. 139–

© 2015, IJARCSSE All Rights Reserved

[27] D. Boneh, G. D. Crescenzo, R. Ostrovsky, and G. Persiano. Public key encryption with keyword search, in EUROCRYPT, 2004, pp. 506–522.

[28] J. Baek, R. Safavi-Naini, and W. Susilo. Public key encryption with keyword search revisited, in ICCSA, 2008, pp. 1249– 1259.

[29] P. Golle, J. Staddon, and B. R. Waters. Secure conjunctive keyword search over encrypted data, in ACNS, 2004, pp. 31–45.

[30] D. Boneh, E. Kushilevitz, R. Ostrovsky, and W. E. Skeith III. Public key encryption that allows PIR queries. In A. Menezes, editor, Advances in cryptology — CRYPTO 2007, volume 4622 of LNCS, pages 50–67. Springer, 2007.

[31] D. Boneh, E.-J. Goh, and K. Nissim. Evaluating 2-DNF formulas on ciphertexts, in Theory of Cryptography - TCC'05, ser. Lecture Notes in Computer Science, vol. 3378. Springer, 2005, pp. 325–341.

[32] S.Buyrukbilen and S.Bairas. Privacy preserving ranked search on public key encrypted data, in Proc. IEEE International Conference on High Performance Computing and Communications (HPCC), November 2013.

[33] B. H. Bloom. Space/time trade-offs in hash coding with allowable errors, Communications of the ACM, vol. 13, no. 7, pp. 422–426, 1970.

[34] C. Gentry and Z. Ramzan. Single-database private information retrieval with constant communication rate, in ICALP, 2005, pp. 803–815.

[35] Ningduo peng et al. Query biased preview over outsourced and encrypted data, scientific world journal, 2013; 2013: 860621.

[36] Cloud Security Alliance. Top Ten Threats, and Available at www.cloudsecurityalliance.org/topthreats/csathreats.v1.0.pdf.

[37] Rivest, R., Adleman, L., and Dertouzos, M. On Data Banks and Privacy Homomorphisms, Foundations of Secure Communication, Academic Press, pp. 169-177.

[38] Shannon, C. Communication Theory of Secrecy Systems, Bell System Technical Journal, Vol 28, Issue 4, October 1949, pp. 656- 715.

[39] Goldwasser, S., and Micali, S. Probabilistic Encryption. Journal of Computer and System Sciences, Vol 28, Issue 2, April 1984, pp. 270-299.

[40] Benaloh, J. Verifiable Secret-Ballot Elections. Doctoral Dissertation, Department of Computer Science, Yale University, New Haven, Connecticut, 1988, USA.

[41] Naccache, D. and Stern, J. A New Public Key Cryptosystem Based on Higher Residues, In Proceedings of the 5th ACM Conference on Computer and Communications Security (CCS'98), pp. 59-66, ACM Press, 1988, New York, NY, USA.

[42] Okamoto, T., and Uchiyama, S. A New Public-Key Cryptosystem as Secure as Factoring, In Advances in Cryptology- Proceedings of EUROCRYPT'98, Lecture Notes in Computer Science (LNCS), Vol 1403, Springer-Verlag,1998, pp. 308-318.

[43] Paillier, P. Public-Key Cryptosystems Based on Composite Degree Residuosity Classes, In Advances in Cryptology – Proceedings of EUROCRYPT'99, Lecture Notes in Computer Science (LNCS), Vol 1592, Springer-Verlag, 1999, pp. 223-238.

[44] Damgard, I., and Jurik, M. A Generalisation, a Simplification and Some Applications of Paillier's Probabilistic Public-Key System, In Proceedings of the 4th International Workshop on Practice and Theory in Public Key Cryptography (PKC'01), Lecture Notes in Computer Science (LNCS), Vol 1992, Springer-Verlag,2001, pp. 119-136.

[45] Galbraith, S. D. Elliptic Curve Paillier Schemes. Journal of Cryptology, Vol 15, No 2, August 2002, pp. 129-138.

[46] Castagnos, G. An Efficient Probabilistic Public-Key Cryptosystem over Quadratic Fields Quotients. Finite Fields and Their Applications, Vol 13, No 3, July 2007, pp. 563-576.

[47] Smart, N. P., and Vercauteren, F. Fully Homomorphic Encryption with Relatively Small Key and Ciphertext Sizes, In Public Key Cryptography - Proceedings of the 13th International Conference on Practice and Theory in Public Key Cryptography (PKC'10), Lecture Notes in Computer Science (LNCS), Vol 6056, Springer-Verlag, 2010, pp. 420-443.

[48] Lyubashevsky, V., and Micciancio, D. Asymptotically Efficient Lattice-Based Digital Signatures, In Proceedings of the 5th International Conference on Theory of Cryptography (TCC'08), Lecture Notes in Computer Science (LNCS), Vol 4948, Springer-Verlag, 2008, pp. 37-54.

[49] Brakerski, Z., Gentry, C., and Vaikuntanathan, V. Fully Homomorphic Encryption without Bootstrapping, In Proceedings of the 3rd Innovations in Theoretical Computer Science Conference (ITCS'12), pp. 309-325, ACM Press, New York, NY, 2011 USA.

[50] Brakerski, Z., and Vaikuntanathan, V. Fully Homomorphic Encryption from Ring-LWE and Security for Key Dependent Messages, In Advances in Cryptology- Proceedings of CRYPTO'11, Lecture Notes in Computer Science (LNCS), Vol 6841, Springer-Verlag,2011, pp. 505-524.

[51] Brakerski, Z., and Vaikuntanathan, V. Efficient Fully Homomorphic Encryption from (Standard) LWE, In Proceedings of the IEEE 52nd Annual Symposium on Foundations of Computer Science (FOCS'2011), ACM Press, New York, USA, pp. 97-106.

© 2015, IJARCSSE All Rights Reserved

[52] Kobayashi, T., Ogura, N., Uchiyama, S., and Yamamoto, G. An Improvement of Key Generation Algorithm for Gentry's Homomorphic Encryption Scheme, In Advances in Information and Computer Security- Proceedings of the 5th International Conference on Advances in Information and Computer Security (IWSEC'2010), Lecture Notes in Computer Science (LNCS), Vol 6434, Springer-Verlag, pp. 70-83.

[53] Chunsheng, G. More Practical Fully Homomorphic Encryption. International Journal of Cloud Computing and Services Science, Vol 1, Issue 4, 2012, pp. 199-201.

[54] Cong Wang, Kui Ren, and Qian Wang. Security challenges for the public cloud, available at http://www.cs.cityu.edu.hk/~congwang/research.html.

[55] W. Diffie and M. Hellman. New directions in cryptography. IEEE Transactions on Information Theory, 1976, pp. 22(6):644{654).

[56] V. Madhu Viswanatham, and S.Sudha. Addressing security and privacy issues in cloud computing, Journal of Theoretical and Applied Information Technology, ISSN: 1992-8645, 20th February 2013. Vol. 48 No.2.

[57] Yin Hu. Improving the efficiency of homomorphic encryption schemes, Thesis for Ph.D., Worcester polytechnic institute, May 2013. Available in https://www.wpi.edu/Pubs/ETD/Available/etd-042513-154859/unrestricted/YHu.pdf.

[58] Jianhua Yu, Jin Li, Xueli Wang, and Wei gao. Conjunctive Fuzzy Keyword Search over Encrypted Data in Cloud Computing, TELKOMNIKA Indonesian Journal of Electrical Engineering, Vol.12, No.3, March 2014, pp. 2104 ~ 2109.

[59] Y.T.Hou, H.Li, W.Lou, and W.Sun. Privacy-preserving keyword search over encrypted data in cloud computing, Insecure Cloud computing, edited by S.Jajodia et al., Springer, 2014.

[60] Sun, W., Wang, B., Cao, N., Li, M., Lou, W., Hou, Y.T., Li, H. Privacy-preserving multikeyword text search in the cloud supporting similarity-based ranking, In Proceedings of the 8th ACMSIGSAC symposium on Information, computer and communications security,2013, ACM, pp. 71–82.

[61] Zhihua Xia, Li Chen, Xingming Sun, and Jin Wang. An efficient privacy preserving semantic multikeyword ranked search over encrypted data, Advanced Science and Technology Letters, Vol.31, 2013, pp.284-2892013.

[62] Mark Joye, " On the Power of Misbehaving Adversaries and Security Analysis of the original EPOC", published in D. Naccache , Ed., Topics in cryptology- CT- RSA 2001, Vol 2020 of Lecture notes computer science, Springer-Verlag 2001. pp-208-222

[63] Sigrun Golush. The development of homomorphic cryptography, Thesis, Vienna University of technology", Available in http://dmg.tuwien.ac.at/drmota/DA_Sigrun%20Goluch_FINAL.pdf.

[64] C. Gentry. A fully homomorphic encryption scheme, Ph.D. dissertation, Stanford University, 2009, Available at http://crypto.stanford.edu/craig.

[65] Marten van Dijk, Gentry, Halevi and Vaikuntanathan. Fully homomorphic encryption over the integers, In EUROCRYPT, 2010, pp. 24–43, Available at http:// eprint.iacr.org/2009/616.pdf.

[66] D.Stehle and R. Steinfeld. Faster Fully Homomorphic Encryption, Cryptology ePrint Archive: Report 2010/299. Available at http://eprint.iacr.org/2010/299.

[67] V. Lyubashevsky, C. Peikert, and O. Regev. On Ideal Lattices and Learning with Errors over Rings, to appear in the proceedings of EUROCRYPT 2010.

[68] Jean-Sebastien Coron, Tancrede Lepoint and Mehdi Tibouchi. Batch Fully Homomorphic Encryption over the Integers, Eurocrypt 2013. Available at http://eprint.iacr.org/2013/036.pdf.

[69] Kurt Rohloff, David Bruce Cousins. A Scalable Implementation of Fully Homomorphic Encryption Built on NTRU, February 2014. Available at http://www.dcsec.unihannover.de/fileadmin/ful/mitarbeiter/brenner/wahc14_RC.pdf.

[70] J. Hoffstein, J. Pipper, and J. H. Silverman. NTRU: A Ring Based Public Key Cryptosystem, in Proc. of Algorithmic Number Theory, Third International Symposium (ANTS 3) (J. P. Buhler, ed.), vol. LNCS 1423, Springer-Verlag, June 21-25 1998, pp. 267-288.

© 2015, IJARCSSE All Rights Reserved