INTERNATIONAL JOURNAL OF ADVANCE RESEARCH AND DEVELOPMENT
IJARnD
(Volume3, Issue3)
Available online at: www.ijarnd.com

Dual Cloud Secure Keyword Search Scheme with Public Key Encryption

Baranidharan. U[1] , Naveen Kumar.A[2], Nagaraj. G[3], Paulinkumar.M[4], Rajkumar. S. D[5]

[1,3,4,5]*Student, The Kavery Engineering College, M. Kalipatti, Tamil Nadu*
[2]*Assistant Professor, The Kavery Engineering College, M. Kalipatti, Tamil Nadu*

ABSTRACT

Dual cloud database to realize convenient and low-cost applications and services. In order of services to provide sufficient functionality for SQL queries, many secure database schemes have been proposed. However, such schemes are vulnerable to privacy leakage to a cloud server. The main reason is that database is hosted and processed the cloud server, which is beyond the control of data owners. For the numerical range query (">", "<", etc.), those schemes cannot provide sufficient privacy protection against practical challenges, e.g., privacy leakage of statistical properties, access pattern. Increased the number of queries will inevitably leak more information to the cloud server. dual cloud architecture for protect database, with a series of intersection protocols are provide security preservation to various numeric related range queries. Secure analysis shows privacy of numerical information can strongly protected against dual cloud providers in our projects. Applying the proxy signature technique to manage the group leader can effectively grant the permission of group management to one or more chosen group members can analysis easy located from the different resource. Proxy server encryption techniques most computational operation delegated to Cloud Servers without disclosing the private information. Extensive security and performance analysis are shows that our proposed scheme is high efficient and satisfies the could later retrieve. Security requirements for public cloud secure sharing the data. Efficient and resilient against loss malicious data modification attack and even server attacks.

Keywords: *Database, Range Query, Privacy Preserving, Cloud Computing.*

1. INTRODUCTION

The growing industry of cloud has provide a service paradigm of storage/computation outsourcing helps to reduce users' burden of IT infrastructure maintenance, and reduce the cost for both the enterprises and individual users However, due to the privacy concerns that the cloud service provider is assumed semi-trust (honest-but-curious.), it becomes a critical issue to put sensitive service into the cloud, so encryption or obfuscation are needed before out saucing sensitive data such as database system - to cloud. The typical scenario for outsourced database is described in Fig as that in Crypt DB.A cloud client, such as an IT enterprise, wants to outsource its database to the cloud, which contains valuable and sensitive information (e.g. transaction records, account information, disease information), and then access to the database (e.g. SELECT, UPDATE, etc.) Due to the assumption that cloud provider is honest-but-curious the cloud might try his/her best to obtain private information for his/her own benefits. Even worse, the cloud could forward such sensitive information to the business competitors for profit, which is an unacceptable operating risk.

The privacy challenge of outsourced database is two-hold. Sensitive data is stored in cloud, the corresponding private. The data records often contain some sensitive information that should not be exposed to the cloud server. Therefore, the client should encrypt the database and store the encrypted version on the server. In some scenarios, the data (plaintext) of client undergoes frequent while small modifications. For example, one anti-virus company outsources its virus database to a cloud server. Also, the company must add the new-discovered viruses to the database every day. Generally, the daily new-discovered viruses are a very tiny part of whole database and almost all parts of database remain unchanged. In this case, the client must *re-compute* and *update* the encrypted version (cipher text) on the server at all times. For every large data, it is extremely expensive for the resources-constrained client to re-compute and update the cipher text from scratch each time. Therefore, it is meaningful to propose efficient constructions for VDB with incremental updates (Inc.-VDB, for short). Loosely speaking, Inc.-VDB means that re-computing and updating the cipher text in VDB are both incremental algorithms, i.e., the client can efficiently perform both operations with previous values, rather than from scratch.

Trivially, we can construct efficient VDB schemes based on message authentication codes or digital sig-natures for a static database. However, it is another thing if the client

(frequently) performs updates on the database. As noted in the main technical difficulty in this case is that the client must have a mechanism to revoke the signatures given to the server for the previous values. Otherwise, the malicious server can utilize the previous (while valid) database records and corresponding signatures to respond the current query of the client. This is called the Backward Substitution updates (BSU) attack on VDB. In order to solve this issue, the client should keep track of every change locally. However, this totally contradicts the goal of outsourcing, i.e., the client should use much less resources than those needed to store the database locally.

2. CLOUD COMPUTING

Cloud computing is an (IT) paradigm that enables ubiquitous access to shared pools of configurable system resource and higher-level services that can be rapidly provided with minimal management effort, often over the internet. Cloud computing relies on sharing of resources to achieve coherence and economic of support, similar to a public utility.[11].

3. RELATED WORK

Plenty of researchers have devoted considerable attention to the problem of how to securely outsource different kinds of expensive computations. First project also proved the impossibility of secure outsourcing an exponential computation while locally doing only polynomial time work.it is meaningful only to consider outsourcing database connections expensive polynomial time computations. Access though the pattern format in cress the number.

The computer science community presented a framework to secure outsourcing the scientific computations such as matrix, multiplications and quadrature equitation. However, the solution used the disguise technique and thus led to the leakage of private information. Later, there are plenty of research work that also investigated this problem. Investigated the problem of computing the edit distance between two sequences and presented an efficient protocol to securely outsource sequence cloud index comparisons to two servers. One are more efficient scheme for secure outsource sequence comparisons. [1], [3], [6], [53],

Prevent the cryptographic community, Chum and Petersen firstly introduced the notion of wallets with observers, a piece of secure hardware are installed on the client's computer to perform some expensive computations to resolve the issues.

3.1. Load Balancing Needs

Cloud load balancing is a type of load balancing that is performed in cloud computing.[12] Cloud load balancing is the process of distributing workloads across multiple computing resources. Cloud load balancing reduces costs associated with document management systems and maximizes availability of resources. It is a type of load balancing and not to be confused with Domain Name System (DNS) load balancing. While DNS load balancing uses software or hardware to perform the function,[12] cloud load balancing uses services offered by various computer network companies.

4. EXISTING SOLUTIONS

- Due to the privacy concerns that the cloud service provider is assumed semi-trust. It becomes a critical issue to put sensitive service into the cloud, so encryption is needed before outsourcing sensitive data.
- The cloud could forward such sensitive information to the business competitors for profit, which is an unacceptable operating risk.

Client's frequent queries will inevitably and gradually reveal some private information on data statistic properties.

4.1. Disadvantage

- In existing system only use the access control techniques to block friend in list.
- Don't text based preference are support and therefore it is not probable to avoid un-desired content is does not matter user who propose them Provided that this service is does not a matter of expending beforehand defined web message mining techniques for a unlike usage, moderately it require to developed ad-hoc classic strategies.

4.2. Algorithm

- Pillar cryptographic Algorithm
- Numeric related SQL queries
- Data Encryption Standard (DES) Algorithm

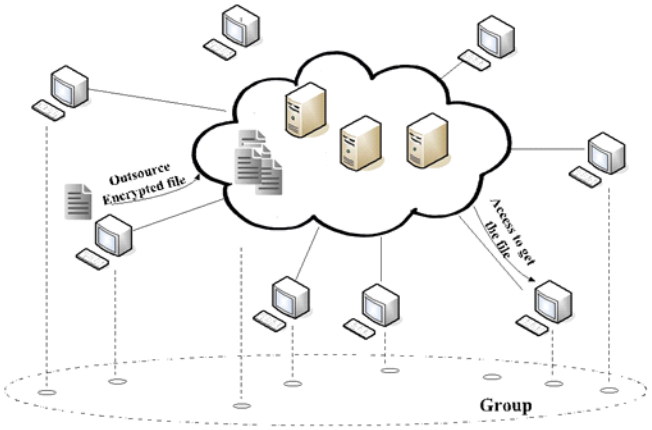


Fig. 1 Cloud Based Group Sharing

5. PROPOSED SYSTEM

Fig. 1 Fuzzy query over encrypted data is becoming a popular topic, since in practical scenarios, some query requests usually want to retrieve data with similar, rather than exactly same indexes. Fuzzy searchable encryption has been introduced for cloud computing in many literatures, such as These schemes deal with the issue that search keywords allows small-scaled distinction in character/numeric level. Specifically for numerical keywords, the query predicate can get numerical records within a range. Some schemes targeted at spatial query, especially which focus on the distance between the query vector and the data. They usually inquire about certain spatial objects (or several numerical attributes) related to the others within a certain distance. Range query has been proposed for that purpose. However, such existing range query schemes are not suitable for practical secure

database due to high storage overhead to maintain the corresponding cipher text.

5.3. Algorithm-New:

i. Order preserving encryption (OPE)

Has been introduced to provide numeric-related range query in structured database, such as Crypt DBOPE preserves the order of values in encryption field, while hiding the actual values. Until now, OPE has been developed to increase both efficiency and security Po pa et al. proposed an ideal-security OPE scheme, in which, an adversary even having the access privilege to a set of Cipher texts still cannot learn the knowledge of data with non-negligible advantage. Although in Bold Yreka definitions, such property has achieved the security boundary of OPE (IND-OCPA), that ideal-secure OPE still cannot satisfy the privacy requirement of secure database. OPE inherently exposes the order of data. that can be utilized to reveal an amount of critical knowledge, although it is always expected to be private.

Bohlen proposed a multi cloud architecture, which can protect the private information of many outsourced services, including database. The main contribution is the introduction of four knowledge partition patterns among multiple cloud service providers: Replication of applications, Partition of application system into tiers, Partition of application logic into fragments, and Partition of application data into fragments. The knowledge is partitioned into two fragments, respectively stored in one cloud, who is assumed to be non-colluding to another cloud.

a. System Architecture

Our proposed secure database system includes a database administrator, and two non-colluding clouds. In this model, the database administrator can be implemented on a client's side from the perspective of cloud service. The two clouds (refer to Cloud A and Cloud B), as the server's side, provide the storage and the computation service. Fig. 2 briefly depicts the architecture of our outsourced secure database system in our scheme.

The two clouds work together to respond each query request from the client/authorized users (availability). For privacy concerns, these two clouds are assumed to be non-colluding with each other, and they will follow the intersection protocols to preserve privacy of data and queries (privacy).

In our scheme, the knowledge of stored database and queries is partitioned into two parts, respectively stored in one cloud. The mechanism guarantees that knowing either of these two parts cannot obtain any useful privacy information. As shown in Fig. 2(a), to conduct a secure database, data are encrypted and outsourced to be stored in one cloud (Cloud A), and the private keys are stored in the other one (Cloud B). (b).For each query, the corresponding knowledge includes the data contents and the relative processing logic. We utilize a prototype of knowledge partition, dividing application logic into two parts, which is firstly proposed by the application logic, as a secret knowledge, is partitioned into [6].

b. Security Assumption

Following the general assumption of many related works in public cloud, we assume the clouds to be honest-but-curious:

On one hand, both of the two clouds will respond with correct information in the interactions of our proposed scheme (honest); on the other hand, the clouds try their best to obtain private information from the data that they process (curious). From the perspective of privacy assurance, here the data not only include permanently stored information.[6].

c. Potential Threats and Privacy Requirements

This section describes the potential threats and the privacy requirements when the database is outsourced to public cloud. The stored data contents and the query processes. Although there are many data encryption schemes, some fail to pro-vide sufficient privacy preservation after statistical analysis: Repeated and large-amount query processes not only leak the access patterns, but also disclose the stored encrypted data progressively.

The privacy issues we consider in this paper mainly include data contents, statistical properties, and query pattern as follows database), but also each temporary query request (i.e., queries).[4],[5].

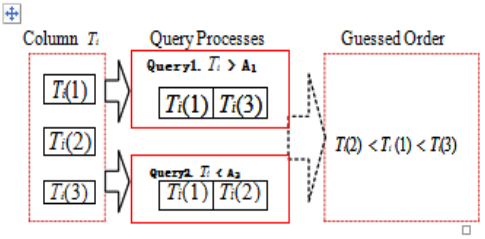
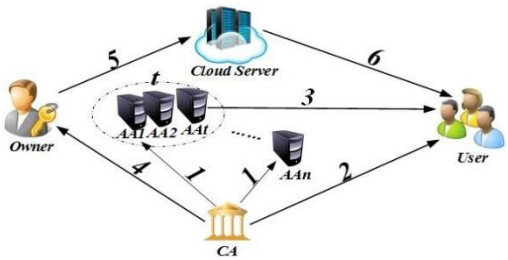


Fig. 3. Repeated Query Discloses Statistical Properties

Fig, 3.Additionally and importantly, as the assumption in some existing works. We assume that the two clouds A and B are non-colluding: Cloud A follows the protocol to add required obfuscation to protect privacy against cloud B, so that cloud B cat obtains additional private information in the interactions with Cloud A. No private information is delivered beyond the scopes of protocols.

Data Contents. The privacy of data contents includes
(1) The definition and description of each column (column name) in the table of the stored database.
(2) The values of each record in the table. Some related works have mainly focused on this issue, in which the column names are blinded (such as Crypt DB and meanwhile the values are encrypted with some other encryption techniques (such as Order Preserving Encryption) and some deterministic encryption schemes so that the adversaries cannot easily and directly guess the meaning of the column, or the values of the data. However, in an outsourced database, utilizing encryption alone, without other mechanisms, is far from being enough to preserve the privacy of the data contents. With the development of data analysis, by extracting features from data and queries, classification technique can help understand the definition of columns, and then breach of confidentiality of data contents.[4],[5].



Fig,4. Could and User Connection

(1) *AA* registers to *CA* to gain (*aid; aid:cert*);

(2) User registers to *CA* to gain (*uid; uid:cert*);
(3) User gains his/her *SK* from any *t* out of *n AA*s;
(4) Owners gain *PK* from *CA*;
(5) Owners upload (*CT*) to the cloud server;
(6) Users download (*CT*) from the cloud server.

Statistical Properties: Besides the static properties can dis-close the private information of data contents, such properties themselves are already sensitive and private for the client. Order Preserving Encryption (OPE), which is widely used in constructing the secure database, with support of range queries, directly exposes the statistical information in the encryption field. Furthermore, the cloud can learn the statistical properties (like order) by repeated query requests. As an example, describes such an attack: After two simple queries over one same column, the order relationship of some data in certain column can be determined. There are also some other direct and indirect scenarios to leak statistical properties. In this way, even though the order property is not exposed to the semi-trusted cloud at the beginning, the cloud can gradually find out the order information after many query requests. Query pattern. The query pattern also contains privacy information, as they can reveal the client's purpose of the query. Even worse, such pattern can leak some statistical properties, as discussed above.

5.4. Module

a. Content Extraction and Management Module

Apart from access context, users may also get back to the previous viewed pages through some content keywords. Instead of extracting content terms from the full web page, we only consider the page segments shown on the screen. There are many term weighting schemes in the information retrieval field. The most generic one is to calculate term frequency-inverse document frequency (tied) for personalized web re-visitation, merely counting the occurrence of a term in the presented page segment is not enough. Also, user's web page browsing behaviors (e.g. visitation time length and highlighting or not), as well as page's subject headings, are counted as user's impression and potential interest indicators for later recall. In a similar manner as access context, we bind an impression score to each extracted content term *d*, showing how likely the user will refer to it for recall based on the four normalized features **Definition 2:** Let d be a content term extracted from the web page segment, shown on the screen of the access program w[was; we]. The impression score of d with w is defined as: dies (w; d)= Len(w; d)+ 2 Highlight(w; d)+3 Head(w; d) + 4 T diff (w; d); where

1) Len (w; d) is the ratio of the time length when the page segment containing d was displayed on the screen versus the maximal display time length of all the viewed page segments; 2)Fig. 4 shows a few content terms extracted from the accessed web page w, where extracted term d's total focus time duration Durer is more than threshold d = 30 seconds. We organize all the extracted content terms, together with their initial impression scores into a Tire tree based on the longest common prefix. For each term at the leaf node of the Tire tree, an inverted index recording the IDs of all the accessed web pages containing the term is built to facilitate content-based re-search. Like probabilistic context tree in the episodic memory, terms' impression scores in the semantic memory.

b. Context Acquisition and Management Module

1) When the Catalan and Fire presented an elegant construction for building a general vb framework from vector command. The first overview their vb general framework and then present a security weakness of the construct the database.

2) Derived a vector command scheme with hiding property can be constructed through composing a standard command scheme with any vector command scheme that does not satisfy the hiding to analysis.

5.5. The General Framework

Catalan-Fire's vb general construction from vac-tor commitment is given as follows.

- Setup (1 k, DB): Let the database be DB = (i, vi) for $1 \le i \le q$. Run the key generation algorithm of vector commitment to obtain the public parameters PP ← V C Keen(1 k, q). Run the committing algorithm to compute the com-moment and auxiliary information (C, aux) ←
- V C Comp (v 1, · · ·, sq.). Define PK = (PP, C) as the public key of vb scheme, S = (PP, aux, DB) as the database encoding, and SK =⊥ as the secret key of the client.
- Verify (PK, x, τ): Parse the proofs τ = (vex, πx). If V C Leap(C, x, vex, πx) = 1, then return vex. Other-wise, return an error ⊥.Update (SK, x, v′): To update the record of index x, the client firstly retrieves the current record vex from the server. That is, the client obtains τ ← Query (PK, S, x) from the server and checks that Verify (PK, x, τ) = vex =6⊥. Also, the client
- Computes (C′, U) ← V C Update(C, vex, x, vex′) and outputs PK′ = (PP, C′) and tax = (PK′, vex′, U).
- Then, the server uses vex′ to update the database record of index x, PK′ to update the public key, and U to update the auxiliary information.

5.7. Formal Definition

1) We consider the database DB as a set of tuples (x, m_x) in some appropriate domain, where x is an index and m_x is the corresponding value. Informally, a VDB scheme allows a resource-constrained client to outsource the storage of a very large database to a server in such a way that the client can later retrieve and update the database records from the server. Inherently, any attempts to tamper with the data by the dishonest server will be detected with an overwhelming probability when the client queries the database. In order to ac this paper is organized as follows. In Section

2) We present the formal definition and security requirements of V DB. Some preliminaries are presented in Section in Section. [9],[10]

3) We overview Catalano-Fiore's VDB Framework from vector commitment and present some security flaws of the construction. We propose a new efficient V DB framework and a con-create VDB scheme in Section

4) The security analysis of the proposed VDB scheme and comparison with existing schemes are given in Section 6. Finally, con-including remarks will be madein Section. Have the confidentiality of the data record m_x, the client can use a master secret key to encrypt each m_x using a symmetric encryption scheme such as AES. Trivially, given the cipher text v_{ex}, only the client can compute the record m_x. Therefore, we only need to consider the case of encrypted database (x,

v_{ex}). This is implicitly assumed in the existing academic research. The formal definition for verifiable databases with updates is given as follows.

Security Requirements

a. In the following, we introduce some security requirements for V DB. The first requirement is the security of V DB scheme. Intuitively, a V DB scheme is secure if a malicious server cannot convince a versifier to accept an invalid output, i.e., v 6= vex where vex is the value of database record in the index x. Note that vex can be either the initial value given by the client in the setup stage or the latest value assigned by the client in the update procedure. Beanbag, Gunner and Valhalla's presented.

Definition 2: (**Security**) A VDB scheme is secure if for any database DB ∈ [q]×{0, 1}*, where q = poly(k), and for any probabilistic polynomial time (PPT) ad-versify A.[10].

6. WEB RESERVATION BY CONTEXT KEYWORD

Now each user's accessed web page w is bounded with a probabilistic context tree (denoted as w# tree) and a probabilistic term list (denoted as w# list). Let W be the set of user's previously accessed web pages. A revisit query posted by the user at time t is expressed as Wm = Q(W; Qc; Qi; t), where Qc is a set of framework keywords, Qi is a set of satisfied keywords, and response We is a ranked list of matched web pages from W .

In response to a user's web re-visitation request, consisting of a set of context keywords QC and a set of content keywords Q, issued at time t, all the context trees and term lists of user's accessed pages W will be examined, with pages that match Q being extracted as the candidate matched page set Wk. Then the pages with higher matching score will be returned as query result. We call probabilistic context tree w# tree contains QC, if and only if for each context keyword QC ∈ QC, there exists a node c in w# tree such that QC ∈ c:title, denoted as QC ⊆c w# tree. Similarly, we call probabilistic term list w# list contains Q, if and only if for each content keyword q ∈ Q, there exists a term d in w# list such that q = d, denoted as Q ⊆d w# list.

The detailed procedure is illustrated in Algorithm 1. Through scanning the inverted index, the candidate matched page set *Wk.* can be determined based on matched context trees and matched term lists against a revisit query *Q* (line 2-4).To compute context ranking**.**

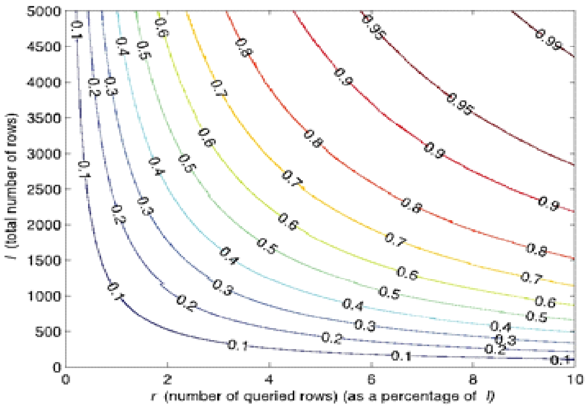
A web page ranking example for a revisit query containing QC and Q.The detailed procedure is illustrated in Algorithm a. Through scanning the inverted index, the candidate matched page set Wk. can be determined based on matched context trees and matched term lists against a revisit query Q (line 2-4). To compute context ranking, it firstly splits the matched context tree into multiple satisfied sub-trees, then traverses the matched nodes to merge ancestor nodes with child nodes along the same hierarchical path. After the match score, we can determine each sub tree's ranking score crank (we sub| QC; t) and add them up (line 5-15).

7. SECURITY ANALYSIS AND PERFORMANCE EVALUATION

In this section, we analyze our proposed scheme in terms of correctness, security, and efficiency. Our security analysis focuses on the adversary model defined in Section 2. We also evaluate the efficiency of our scheme via implementation of both file distribution preparation and verification token precipitation.

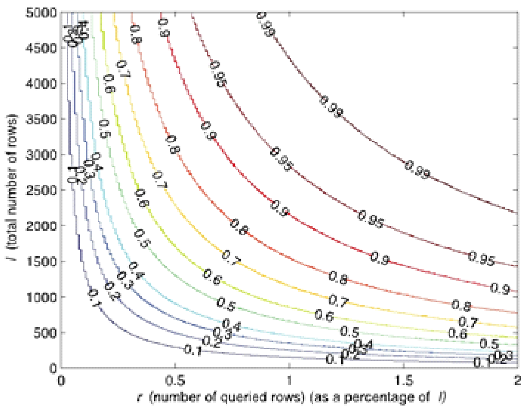
a. Correctness Analysis

First, we analyze the correctness of the verification procedure. Upon obtaining all the response $R^{i}_{s}{}^{es}$ from servers and taking away the random blind values from and, hence, the left-hand side (LHS) of the equation expands as



P d ¼	P r mð	1	P r f Þ	. Fig. 3 plots P d	for Different k ¼ 5.	3	values of From the
l; r; z	while we set p ¼ 16, N_C ¼ 10, and						
proposition 2 of our	previous work in [33],	the false					
	R	ð1þ2 ^{pinch} 1					
negative probability is	Pfk¼P r_1 pp. r_2, where P r_1 ¼		2^nc 1				
and P r_2 ¼ ð1	Pr_1Þð2 ^{p}p. .						

Based on above discussion, it follows that the probability



a. Security Strength

In our scheme, servers are required to operate only on specified rows in each challenge-response protocol

execution. We will show that this "sampling" strategy on selected rows instead of all can greatly reduce the computational overhead on the server, while maintaining high detection probability for data corruption.

7.1. Detection Probability against Data Modification

The cloud NC servers are misbehaving to the possible compromise or Byzantine failure. In the analysis, we don't limit the value of NC. Thus, all the analysis results hold even if all the servers are compromised. We leave the explain a collusion resistance of our scheme against this worst case scenario in a later section. Assume the adversary modifies the data blocks in z rows out of the rows in the encoded file matrix. Consider r be the number of different rows for which the user asks for checking in a challenge. Consider X be a discrete random variable that is defined to be the number of rows chosen by the user that matches the rows modified by the adversary. You will first analyze the matching probability that at least one of the rows picked by the user matches one of the rows modified by the adversary: PR ¼ 1 P fox ¼ 0 ¼ 1 Fri¼01ð1 manful z i ; 1gÞ

If none of the specified r rows in the itch verification process are deleted or modified, the adversary avoids the detection.

Next, we study the probability of a false negative result that there exists at least one invalid response calculated from those specified r rows, but the checking equation still holds. Consider the responses Rði1Þ; . . . ; Rain returned from the data storage servers for the itch challenge, each response value RI Jim, calculated within FF ð2pÞ, is based on r blocks on server j. The number of responses Rðmþ1Þ; . . . ; Run from parity servers is k ¼ n m.

ALGORITHM:

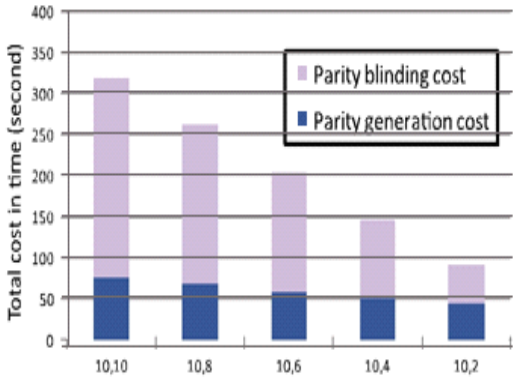
Thus, according to of data modification detection across all storage servers is figure we can see that if more than a fraction of the data file is corrupted, then it suffices to challenge for a small constant number of rows in order to achieve detection with high probability. For example, if z ¼ 1% of l, every token only needs to cover 460 indices in order to achieve the detection probability of at least 99 percent.

probability is P_R	¼ 1		$_{i}{}^{r}{}_{¼0}{}^{1}$ð1 mind			z^ l i	; 1gÞ, where z^ z.		
$_{n}$ext, $_{j}$ we consider			the Q	False	negative probability that				
ð Þ	ð Þ	B							
R_I	¼ v_i	when	at	least	One	of z^ blocks is modified.			
According	to [33,		Proposition		1], tokens calculated in				

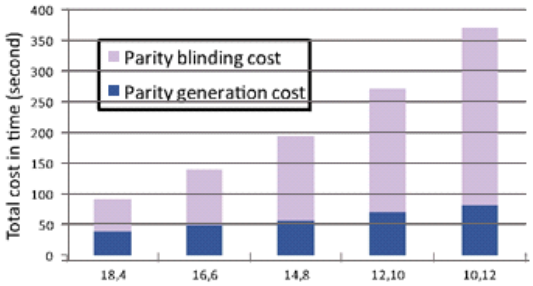
7.1. Identification Probability for Misbehaving Servers:

We have shown that, if the adversary modifies the data blocks among any of the data storage servers, our sampling checking scheme can successfully detect the attack with high probability. As long as the data modification is caught, the user will further determine which server is malfunctioning. This can be achieved by comparing the identifying misbehaving server(s) can be computed in a similar way. It is the product of the matching probability for sampling check and the probability of complementary event for the false negative result. Obviously, the matching

FF ð2pÞ for two different data vectors collide with probability PR ¼ 2 p. Thus, the identification probability for misbehavior server(s) is Pd ¼p r 1 rB m f. Along with the analysis in detection probability, if z 1% of l and each b b¼b.



(A)



(B)

7.2. EVALUATION:

In this section, we provide a thorough experimental evaluation of the proposed Inc.
VDB scheme. Our experiments are simulated with the pairing-based cryptography (PB) library and Opens' open-source library on a LINUX machine with Intel Core TM I7-4600U processors running at 2.70 GHz and 8 GB memory. Throughout this experiment, in order to preciselyevaluate the computation complexity at both client and server sides, we simulate both entities on this LINUX machine.

Since the pairing functions are actually shown to be insecure (or very inefficient after some fixing), we do the experiment using the asymmetric pairings (i.e., either Type 2 or Type 3 pairings). The elliptic curve we used is a MINT d224-curve, where the base field size is 224-bit and the embedding degree of the curve is . Also, we adopt the famous Crypt DB database system in the experiments and the data set is the encrypted file with the length of 8 MB.[8].

The providing a time costs simulation for schemes and another scheme in Fig.3.A& B

- Also it can be provide the performance analysis of our schemes.

- The argue groups G 1 and G 2 in Beanbags Geneon Valhalla's scheme are different from those in our scheme since their scheme uses bi linear groups of Composite order. Actual the operations in the Composite order groups require

much more expensive computational overload though we use the same notions of all schemes.

The regular means the output of operation should be computed from scratch.

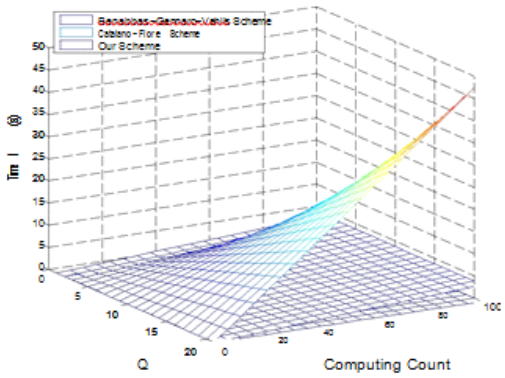
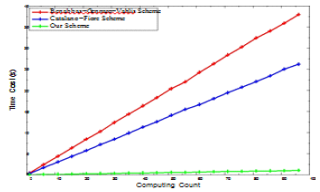


Fig.1. Query Comparison

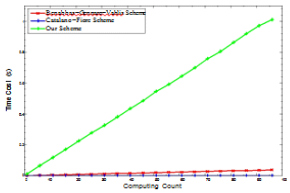
Fig.1 Query Comparison incremental encryption in Fig.4 that time cost of query, verify and update algorithms for all three schemes are shown in Fig. 1, Fig. 2, Fig.3(a) and Fig. 3(b), respectively. Fig.1 shows that the query time cost of our scheme is always 0, and the query time cost of scheme is relatively small compared with scheme. As shown in Fig. 2 the verification time cost of the three schemes are all linear with the com mutating count, and our verification algorithm is the most efficient one. The main reason is that we use the incremental hash algorithm in our scheme [the input for the incremental hash algorithm in our simulation is 8 MB].[7]
In Fig. 3(a), we provide the efficiency comparison for data update of the client side. The simulation results show that the growth rate of our scheme is much smaller than that of schemes Fig.
3(b) shows the efficiency comparison for data update of server side. The scheme does not need any computation cost in this phase, and thus the computation time is always 0. Since we introduce the pairing computation in the server data update phase, the computation cost of our server side update is relatively higher than that of the scheme.

However, we argue that the computational overhead of query algorithm is only performed by the cloud server rather than the resource-constrained client. Therefore, it is reasonable for cloud outsourcing environment. On the other hand, the simulation results in Fig. 4 indicate that our incremental encryption scheme is much more efficient than the normal encryption scheme when the number of updated blocks is sufficiently large.

In both verification and update algorithms which are performed by client, the simulation results indicate that our scheme is more efficient than scheme besides, since the scheme suffers from the FAUN attack and the scheme only provides private venerability, our scheme is most suitable for real-world applications.



(a) Client Update Computation



(b) Server Update Computation

8. CONCLUSION

In this paper, we investigate the problem of data security in cloud data storage, which is essentially a distributed storage system.

To achieve the assurances of cloud data integrity and availability and enforce the quality of dependable cloud storage service for users, we propose an effective and flexible distributed scheme with explicit dynamic data support, including block update, delete, and append. We rely on erasure-correcting code in the file distribution preparation to provide redundancy parity vectors and guarantee the data dependability.

By utilizing the homo-orphic token with distributed verification of erasure- coded data, our scheme achieves the integration of storage correctness insurance and data error localization, i.e., whenever data corruption has been detected during the storage correctness verification across the distributed servers, we can almost guarantee the simultaneous identification of the misbehaving server(s).

Considering the time, computation resources, and even the related online burden of users, we also provide the extension of the proposed main scheme to support third-party auditing, where users can safely delegate the integrity checking tasks to third- party auditors and be worry-free to use the cloud storage services. Through detailed security and extensive experiment results, we show that our scheme is highly efficient and resilient to Byzantine failure, malicious data modification attack, and even server colluding attacks.

In this paper, we investigate the problem of data security in cloud data storage, which is essentially a distributed storage system.

To achieve the assurances of cloud data integrity and availability and enforce the quality of dependable cloud storage service for users, we propose an effective and flexible distributed scheme with explicit dynamic data support, including block update, delete, and append. We rely on erasure-correcting code in the file distribution preparation to provide redundancy parity vectors and guarantee the data dependability. By utilizing the homo- orphic token with distributed verification of erasure- coded data, our scheme achieves the integration of storage correctness insurance and data error localization, i.e., whenever data corruption has been detected during the storage correctness verification across the distributed servers, we can almost guarantee the simultaneous identification of the misbehaving server(s). Considering the time, computation resources, and even the

related online burden of users, we also provide the extension of the proposed main scheme to support third-party auditing, where users can safely delegate the integrity checking tasks to third- party auditors and be worry-free to use the cloud storage services. Through detailed security and extensive experiment results, we show that our scheme is highly efficient and resilient to Byzantine failure, malicious data modification attack, and even server colluding attacks.

9. REFERENCES

[1]M.J. Atallah, and K.B. Frikken, *Securely outsourcing linear al-gebra computations*, Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security (Asi-aCCS), pp.48-59, 2010.

[2] M.J. Atallah, K. N. Pantazopoulos, J. R. Rice, and E. H. Spafford, *Secure outsourcing of scientific computations*, Advances in Computers, vol.54, pp.216-272, 2001.

[3] C. Wang, K. Ren, J. Wang, and Q. Wang, *Harnessing the Cloud for Securely Outsourcing Large-scale Systems of Linear Equations*, IEEE Trans. Parallel Distribution Systems, 24(6), pp.1172-1181, 2013. A preliminary version of this paper is presented at ICDCS, pp.820-828, 2011.

[4] X. Wu, X. Zhu, G.-Q. Wu, and W. Ding, "Data mining with big data,"IEEE Transactions on Knowledge and Data Engineering, vol. 26, no. 1,97–107, 2014.

[5]S.R. Ganta, S. P. Kasiviswanathan, and A.Smith, "Composition attacks and auxiliary information in data privacy,"in Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2008, pp. 265–273.

[6]GRuschka,M.jenson,L.L.Lancono,andtheir applications,PKC 2013,LNCS 7778, SpringerVerlag,pp.55-72,2013.

[7]D. Catalano and D. Fiore, Vector commitments and their applica-tions, PKC 2013, LNCS 7778, Springer-Verlag, pp.55-72, 2013.

[8] J. Camenisch, S. Hohenberger, and M. Pedersen, Batch Ver-ification of Short Signatures, EUROCRYPT 2007, LNCS 4515, Springer, pp. 246-263, 2007.

[9] S.Benabbas, R.Gennaro, and Y.Vahlis, Verifiable delegation of computation over large datasets, CRYPTO 2011, LNCS 6841, Springer, pp.111-131, 2011.

[10] J. Camenisch, M. Kohlweiss, and C. Soriente, An accumulator based on bilinear maps and efficient revocation for anonymous credentials, PKC 2009, LNCS 5443, Springer, pp.481-500, 2009.

[11] https://www.wikipedia.org./wiki/cloud-computing.

[12]hhtp://www.wikipedia.org./wiki/loadbalancing/cloud computing.

