International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169
Volume: 3 Issue: 3 1577 - 1581

Secured Data Outsourcing in Cloud Computing

Nivedita B. Patil
P.G. Student, Department of Computer Science,
Astral Institute of Technology & Research, Indore, India.
nivedita.patil04@gmail.com

Prof. Abhay Pawar
Assistant Professor, Department of Computer Science,
Astral Institute of Technology & Research, Indore, India.
abhay655@gmail.com

Rohit P. Vibhandik
P.G. Student, Department of Computer Science,
B. M. College of Technology, Indore, India.
vibhandik.rohit@gmail.com

Abstract— Cloud computing is a popular technology in the IT world. After internet, it is the biggest thing for IT world. Cloud computing uses the Internet for performing the task on the computer and it is the next- generation architecture of IT Industry. It is related to different technologies and the convergence of various technologies has emerged to be called as cloud computing. It places the application software and databases to the huge data centers, where the supervision of the data and services may not be fully trusted. This unique attribute poses many new security challenges which have not been well understood. In this paper, we develop system which allows customer to use cloud server with various profits and strong securities. So when customer stores his sensitive data on cloud server he should not worry about securities, we also protect customer's account from malicious behaviors by verifying the result. This result verification mechanism is highly efficient for both cloud server and cloud customer. Covering security analysis and experiment results shows the immediate practicability of our mechanism design.

Keyword-*Cloud computing, security,saas, multitenant.*

*****

I. INTRODUCTION

Cloud computing is the use of computational resources such as hardware and software that are delivered as a service over an internet. The name cloud computing comes from the use of a cloud-shaped symbol to show the complex infrastructure it contains in system diagrams. Cloud computing trusts remote services with an user's data, software and computation. Cloud computing is a common term for anything that involves delivering hosted services over an Internet. A cloud service has three distinct characteristics that differentiate it from traditional hosting. It provides on demand access, typically by the minute or the hour; it is expandable - a user can have as much or as little of a service as he wants; and the service is fully managed by the service provider. A cloud may be private or public. A public cloud allows services to everyone on the Internet. At Present, Amazon Web Services is the largest public cloud service provider [6].

Quality of service is an important part from the point of data security. In cloud computing there are challenging security threats for various reasons. Firstly, we can't apply old cryptographic technique for data security protection because the user may lose control of data under cloud computing [2].

Each customer stores various kind of data in the cloud and customer wants longtime assurance of data security but the problem of verifying correctness of data stored in the cloud is more challenging. Another security threat is the customer frequently changed data which is stored in the cloud like inserting, deleting, modifying, appending, re-ordering, etc [1]. Lastly, the deployment of Cloud Computing is powered by data centers running in a synchronized, cooperated and distributed approach. Each user's data is redundantly stored in numerous physical locations to reduce the data integrity intimidation. Therefore, distributed protocols for the purpose of storage, correctness and assurance will be most important in achieving a robust and secure cloud data storage system in the existent world. However, such important region remains to be fully opened up in this literature. The cloud computing posses various service models which are given below.

In this paper, we only focus on the Software as a service. In this model, cloud service providers install and operate application software in the cloud and cloud users uses the software from cloud clients. The cloud users need not manage the cloud infrastructure and platform on which the application is running. This thing eliminates the need to install and run the applications on the cloud user's computers, simplifying maintenance and support. Elasticity is the main feature which makes cloud computing different from other applications which can be achieved by cloning tasks onto multiple virtual machines at run-time to meet the changing work demand of the fastest growing IT world. Load balancers are those who distribute the work over the set of virtual machines. This process is not noticeable to the cloud user who sees only a solitary access point. To hold a large number of cloud users, cloud applications can be multitenant-any machine serves many cloud user organization. Common naming conventions to refer to special types of cloud based application software are: communication as a service, business process as a service, test environment as a service, desktop as a service.

Numerous trends are opening up in the era of Cloud Computing is an Internet dependent development and use of computer technology. The low cost and more dominant processors in combination with the software as a service (SaaS) computing architecture, are transforming data centers in the pools of computing service on a large degree [4]. The growing bandwidth of network and reliable, flexible network connections make it possible that users can now subscribe high quality services from data and software that resides solely on remote data centers.

The remarkable benefits, outsourcing computation to the commercial public cloud is also grudging customer's direct control over the systems that consume and produce their data during the computation, which unavoidably brings in new security issues and challenges towards this promising computing model. On the one hand, the outsourced computation on the cloud server often contain sensitive information and data, such as the business related financial records, computational models, proprietary research data,

property related useful information or personally identifiable health related information etc[1].

To take action against unauthorized information outflow, this sensitive data must be encrypted before outsourcing. So to provide end-to-end data confidentiality assurance on the cloud and beyond. However, usual data encryption techniques prevent cloud from performing any significant operation of the fundamental plaintext data information, making the computations over encrypted data information is a very hard problem [1]. On the other hand, the operational details inside the cloud are not transparent to customers. As a result, there exists various motivations for cloud server to behave deceitfully and to return inaccurate results, i.e., they may behave away from the classical semi honest model. For example, to carry out the computations that require a large amount of computing assets, there are huge financial incentives for the cloud to be "lazy" if the customers cannot tell the correctness of the output [1]. In addition, possible software bugs, hardware failures or even attacks from outsiders might also create an effect the quality of the computed results. Thus, we can say that, according to the customer's point of view the cloud is not secure.

The remaining paper is organized as follows. Section II contains the Literature Survey. Then we have discussed the problem in Section III. Section IV provides the Proposed Work and V demonstrates the result of the system. Finally, concluding remark of the whole paper.

II. LITERATURE SURVEY

According to "Non-interactive verifiable computing: Outsourcing computation to entrusted workers" the author R. Gennaro [4] said that, the work is based on the critical and a bit surprising inspection that Yao's Garbled Circuit Construction, in addition to providing safe two-party computation, also provides a "one-time" verifiable computation. In other words, we can get used to Yao's construction to allow a client to outsource the computation of a task on a single input. Specifically, in the preprocessing stage the client garbles the circuit C according to Yao's construction. Then in the "input preparation" stage, the client reveals the random labels associated with the input bits of x in the garbling. This allows the worker to compute the random labels associated with the output bits, and with the output bits client will reconstruct F(x). If the output bit labels are sufficiently long and random, the worker would not be able to guess the labels for an incorrect output, and therefore the client is assured that F(x) is the correct output.

According to "Secure outsourcing of sequence comparisons", the author M. J. Atallah [9] said that, we more precisely stated the edit distance problem, in which the cost of an insertion or deletion or substitution is a symbol-dependent non-negative weight, and the edited distance then the least-cost set of insertions, deletions, and substitutions necessary to transform one string into the other. More officially, if we let λ be a string of length n, $\lambda = \lambda_1 \ldots \lambda_n$ and μ be a string of length m, $\mu = \mu_1 \ldots \mu_m$, both over some alphabet $\sum$. There are three types of allowed edit operations to be done on λ: insertion of a symbol, deletion of a symbol, and substitution of one symbol by another [5]. Each operation has a cost linked with it, namely I(a) denotes the cost of inserting the symbol a, D(a) denotes the cost of deleting a, and S(a, b) denotes the cost of substituting a with b [7][8]. Each sequence of operations that transforms λ into μ has a cost associated with it and the least-cost of such sequence is the edit-distance. The edit path is the actual sequence of operations that corresponds to the edit distance. According to "Secure outsourcing of scientific computation", the authors M. J. Atallah et al. [9] said that they created the first analysis of safe outsourcing of numerical and scientific computation. A set of problem dependent disguising methods are proposed for different scientific applications like linear algebra, sorting, string pattern matching, etc. However, these disguise techniques explicitly allow information disclosure to certain degree. Atallah et al. discuss in [10] and [11], produced two protocol designs for both secured sequence comparison outsourcing and secured algebraic computation outsourcing. However, both protocols used heavy cryptographic primitive such as homomorphic encryptions and/or oblivious transfer and do not scale well for large problem set.

In addition, both designs are built upon the hypothesis of two non-colluding servers and thus vulnerable to colluding attacks. Based on the same guess, Hohenberger et al. [3] give protocols for safe outsourcing of modular exponentiation, which is considered as prohibitively expensive in most public-key cryptography operations. Very recently, Atallah et al. [5] given a provably safe protocol for safe outsourcing matrix multiplications based on secret allocation. While this work outperforms their previous work [11] in the sense of single server hypothesis and computation effectiveness, the main drawback is the large communication overhead. Namely, due to secret sharing method, all scalar operations in original matrix multiplication are expanded to polynomials, introducing important amount of overhead. Considering the case of the result authentication, the communication overhead must be further doubled, due to the introducing of additional pre-computed "random noise" matrices [12][13].

Another large existing list of work that relates to (but is also significantly different from) Secure Multi-party Computation (SMC), first introduced by Yao [11] and later extended by Goldreich et al. [1] and many others. SMC allows two or more parties to together compute some general function while hiding their inputs to each other. As general SMC can be very ineffective, Du and Atallah et al. [5] have proposed a series of customized solutions under the SMC context to a spectrum of special computation harms, such as privacy-preserving cooperative statistical analysis, scientific computation, geometric computations, sequence comparisons, etc. [14]. However, unswervingly applying these approaches to the cloud computing model for secure computation outsourcing would still be problematic. With the major reason is that they did not address the asymmetry among the computational powers possessed by cloud and the customers, i.e., all these schemes in the context of SMC impose each involved parties comparable computation burdens, which we exclusively avoid in the mechanism design by shifting as much as possible computation burden to cloud only.

Lately, Li and Atallah [16] had presented a study for secure and mutual computation of linear programming under the SMC framework. The solution of this problem is based on the additive split of the constraint matrix between two involved parties, followed by a series of interactive (and arguably heavy) cryptographic protocols collaboratively

executed in each iteration step of the Simple Algorithm. This solution has the computation irregularity problem mentioned previously. Besides, they only consider honest-but-curious representation and thus do not guarantee that the final solution is optimal.

Some recent general result can be found in Goldwasser et al. In distributed computing as well as targeting the specific computation assignment of one-way task inversion, Golle et al. [15] planned to insert some pre-computed results (images of "ringers") along with the computation workload to defeat entrusted (or lazy) workers. In Du. et al. [14] planned a system of cheating detection for general computation outsourcing in grid computing. The server is required to provide a assurance via a Merkle tree based on the results it computed. The customer can then use the dedication combined with a sampling approach to carry out the result verification (without re-doing much of the outsourced work.) However, all above schemes allocate server actually see the data and result it is computing with, which is strictly forbidden in the cloud computing model for data privacy. Thus, the problem of result authentication essentially becomes more complex, when both the input/output isolation is demanded. So the duality hypothesis of LP problem and efficiently bundles the result authentication within the method designs, with little extra overhead on both customer and cloud server [17].

III. PROBLEM DEFINATION

Here, we considered a data outsourcing architecture with two different things. Firstly the cloud customer has large amount of data outsourced problems to the cloud server. Secondly, the cloud servers, which have different data resources and provide different utility services, like hosting the public cloud in a pay-par-use manner. In this system we focused on the data outsourcing on the cloud. In the cloud, cloud customer can upload his/her data on the cloud server in original format. So the data is not secure because the cloud server is not fully honest model. As cloud server is a semi-honest model it misbehaves with cloud customer and losses or corrupt cloud customers important data. So we have to analyze this problem of data security and design some goal to achieve the data security problem on cloud.

A. *Design Goals*

To enable secured data outsourcing on the cloud, our system design should achieve the following security goal and performance of system.

- Correctness - Any cloud server that faithfully follows the mechanism must produce an output that can be decrypted and verified successfully by the customer.
- Soundness - No cloud server can generate an incorrect output that can be decrypted and verified successfully by the customer with non-negligible probability.
- Input/output privacy - No sensitive information from the customer’s private data can be derived by the cloud server during the data outsourced.
- Communication Latency - We also check the communication latency between the customers and the cloud for this application since the data outsourced with the running time.

IV. PROPOSE WORK

While developing this system, we present data outsourcing scheme which provides input/output privacy protection and also provides efficient result verification. After studying the `Literature Survey' we implement the solution in our developed system. Here, an overview of securing data outsource methods and discuss different techniques to secure the data outsource while performing transformation on cloud.

A. *Design Mechanism*

Here, we have developed a system `Secure data outsourcing in cloud computing' in which secure data using encryption and decryption technique. In this system, result verification step is run on the cloud server side. Secrete Key, Input Encryption and Output Decryption steps on the cloud customer side. These four steps are:

- *Secrete Key* - Secrete key is key which is generated when the data is encrypted using Rijndael algorithm. This key is a private key with size of 128-bit only. This key is generated only on the cloud customer side and it is saving automatically on the cloud customer's window. This secrete key is also use to decrypt data. Without this key no one can decrypt data.
- *Input Encryption* - To provide strong security to cloud customer’s data we use encryption and decryption technique. Here, combine RSA algorithm with MD5 algorithm to Encrypt data on cloud customer's window. When data is encrypted the secrete key is automatically generated. After encrypting the cloud customers data, that encrypted data is stored on the cloud servers so the cloud server do not loss or corrupt data.
- *Output Decryption* - Here also we combine RSA algorithm with MD5 algorithm to decrypt data on cloud customer's window. When the data is accessed by cloud customer that time cloud server takes cloud customers data from the cloud server. The data is stored on the server is in encrypted format so when cloud customer access this data he need to provide the secrete key to decrypt data. Decryption is done when the data is transforming from cloud server to cloud customer.
- *Result Verification* – Here, we check the input given to the server and output retrieved from the server. So we can compare the input and obtained output and verify both the input and output.

B. *Security Mechanism*

In the security mechanism, we give the best security to cloud customer for storing cloud customers data on the cloud server and prevent cloud customers data from losses and corrupt. So for providing better security to cloud, we check user authentication. The authentication is done through the cloud customer's login. So for user's login we apply combination of the MD5 and RSA algorithm. This algorithm encrypts and decrypts the user password to provide better security and authenticating the valid user. The security of the users authentication is better and no one can access the cloud customers account without user name and password because the password is converted in 128-bit. After authenticating user,

the cloud server checks that user is authorize or not, because authorization occurs after successful authentication.

The Figure 1 shows the architecture of secure data outsourcing on cloud. In the figure 1, we send data from cloud customer to cloud server, that data is storing on the server without encrypted form means original data. We study about the nature of the cloud server i.e. cloud server is a semi-honest model. So the cloud server misbehave with the cloud customers data and loss or corrupt the cloud customers important data and cloud server also send cloud customer data to another cloud customer. So there is a chance to misuse of the cloud customers data. The cloud is not secure from the cloud customer's point of view. So we implement this system on the cloud customer side.

Here, we use an encryption and decryption technique to secure data which is uploaded on the server. Cloud customer upload data on the cloud server, that data is encrypted on the cloud customer. After encrypting data, that encrypted data is stored on the server. When customer wants his data from cloud, then he just get his data from the cloud server and the encrypted data is going to decrypt by providing the secrete key which is generated at the time of encryption then this data is accessed by the cloud customer.

Here, we use encryption and decryption technique on the cloud customer side and the secrete key will be generated at the cloud customer's window. The secrete key is automatically save at the cloud customers window. For the encryption and decryption of the data we use the combination of the RSA and MD5 algorithms. Using the combination of these algorithms the encryption and decryption techniques are strong. With this encryption and decryption technique we provide better security to the cloud.

In this system, we also implemented another technique to protect misuse of cloud customer's data. We verify the data which is retrieved by cloud customer from the cloud server is correct data or not. This verification will be done at cloud server side. In this technique, the cloud server compares the input data and output data (which were uploaded by cloud customer). If the input data and output data is equal then the cloud server send the data to cloud customer after decrypted.

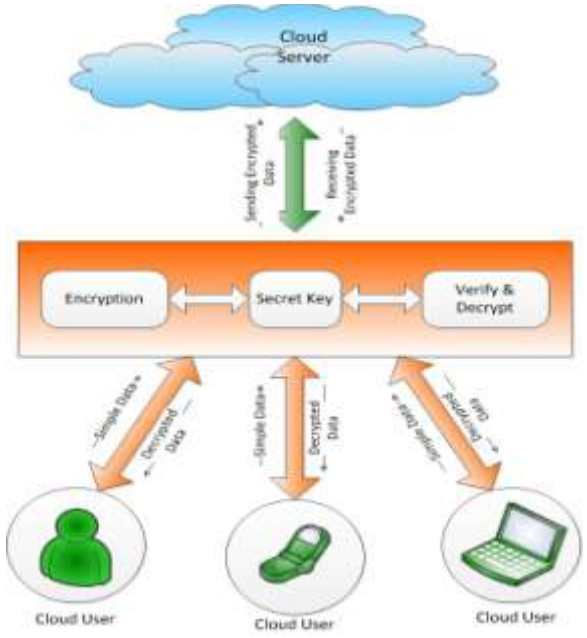

Figure 1. Architecture of secure data outsourcing in cloud computing.

V. RESULT

After finishing the implementation details we have analyzed the system performance and it's behaviors by giving input in different sizes.

A. System Requirements

We now assess the practical efficiency of the implemented system with experimental results. We have implemented "Secured data outsourcing in cloud computing" on configurations. This includes both the customer and the cloud side processes in Visual Studio 2010 and SQL Server R2 for storing database. Both customer and cloud server computations in our experiment are conducted on the same workstation with an Intel Core i3-370M processor running at 2.40 GHz with 4 GB RAM. Here, we also use the Cloud Environment. After purchasing the cloud environment. We have install cloud resources and its environment on our system to use relative cloud environment. Using this cloud environment we get the proper result also. We also have calculated communication latency between the cloud server and cloud customers.

B. Exprimental Results

In this mechanism, we have implemented both the cloud server and cloud customer using four algorithm steps. In the algorithms steps run at both cloud server side and cloud customer side. In the given algorithm steps Secrete Key, Input Encryption and Output Decryption run at cloud customer side and Result Verification step is run at cloud server side.

In the result phase, we checked the system performance with the text data as input value and we observe the system behavior by giving this text data as input in different sizes. We recorded time (sec) for this variable size data. As shown in figure 3, we calculated the cloud efficiency for the variable input data and figure 3 shows the communication latency of cloud customer with cloud server.

The efficiency of cloud increases with increase in data sizes up to certain limit and then it will decreases. We calculated the communication latency which is also increases

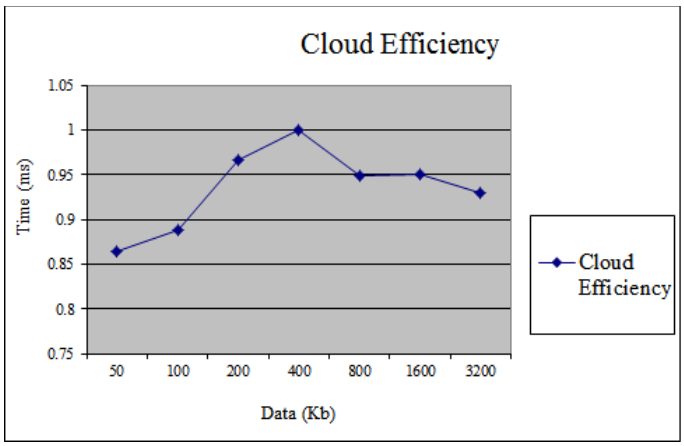

Figure 2. Graph of Cloud Efficiency.

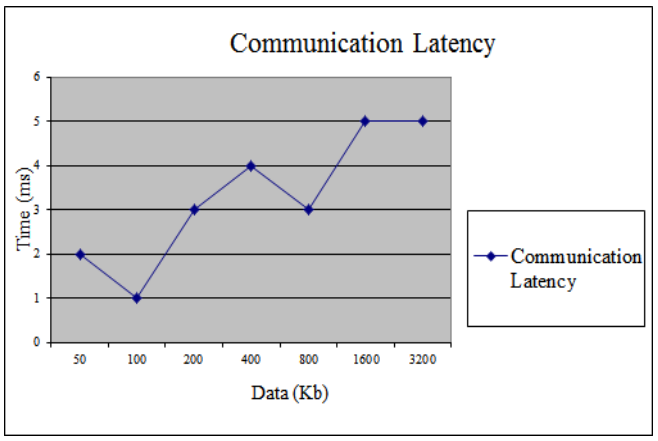

Figure 3. Graph of Communication Latency.

CONCLUSION

As the cloud computing use the internet for sharing the resources so the security of the data is the main issue in the cloud. So in this paper, we formalize the problem of securely outsourcing data in cloud computing. Here, we provide the input/output privacy and correctness/soundness guarantee. So the customer's data will be secure on the cloud server and the personal data may not be corrupt by the cloud server. Here, we also check the communication latency of the cloud. So in this system we improve the performance of the system.

ACKNOWLEDGMENT

At the time of making a survey on this area many people were helped me. I specially thankful to Prof. Atul Thakkar, Principal, Astral Institute of Technology & Research for valuable guidance on this area. Last but not the least we also thank to our Faculty members, staff and friends for being instrumental towards the completion of this paper.

REFERENCES

[1] C. Wang, K. Ren and J. Wang, "Secure and practical outsourcing of linear programming in Cloud computing", IEEE Transition on cloud computing, pp.820-828, April 2011

[2] C. Wang, Q. Wang, K. Ren and W. Lou, "Ensuring data storage security in Cloud Computing", in Proc. Of IWQoS'09, July 2009

[3] S. Hohenberger and A. Lysyanskaya, "How to securely outsource cryptographic computations", in Proc. of TCC, 2005, pp 264-282.

[4] R. Gennaro, C. Gentry and B. Parno, "Non-interactive verifiable computing: Outsourcing computation to entrusted workers", in Proc. of CRYTO'10, Aug 2010.

[5] M. Atallah and K. Frikken, "Securely outsourcing linear algebra computations", in Proc of ASIACCS, 2010, pp. 48-59.

[6] N. Gohring, "Amazon's S3 down for several hours", Online at http://www.pcworld.com/businesscenter/artic le/142549/amazons_s3_down_for_several_hours.html, 2008.

[7] Amazon.com, "Amazon Web Services (AWS)", Online at http://aws.amazon.com, 2008.

[8] Sun Microsystems, Inc., "Building customer trust in cloud computing with transparent security," 2009, online at https://www.sun.com/offers/details/sun_trans parency.xml.

[9] 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

[10] M. J. Atallah and J. Li, "Secure outsourcing of sequence comparisons," Int. J. Inf. Sec., vol. 4, no. 4, pp. 277–287, 2005.

[11] D. Benjamin and M. J. Atallah, "Private and cheating-free outsourcing of algebraic computations," in Proc. of 6th Conf. on Privacy, Security, and Trust (PST), 2008, pp. 240–245.

[12] A. C.-C. Yao, "Protocols for secure computations (extended abstract)," in Proc. of FOCS'82, 1982, pp. 160–164..

[13] O. Goldreich, S. Micali, and A. Wigderson, "How to play any mental game or a completeness theorem for protocols with honest majority," in Proc. of STOC'87, 1987, pp. 218–229.

[14] W. Du and M. J. Atallah, "Secure multi-party computation problems and their applications: a review and open problems," in Proc. of New Security Paradigms Workshop (NSPW), 2001, pp. 13–22.

[15] P. Golle and I. Mironov, "Uncheatable distibuted computations", in Proc. of CT-RSA, 2001, pp. 425-440.

[16] J. Li and M. J. Atallah, "Secure and private collaborative linear programming," in Proc. of CollaborateCom, Nov. 2006.

[17] W. Du, J. Jia, M. Mangal and M. Murugesan, "Uncheatable grid computing", in Proc. Of ICDCS, 2004, pp. 4-11.