Fully Homomorphic Computation- a Scheme to Provide End-to-End Data Confidentiality in the Cloud

Priyanka C[#1], Priya Darshini C[#2], Vinayak G Shavi[#3], Sarvar Begum[*4]
[#] BE 8th SEM, Department of Computer Science and Engineering
Visveswaraya Technological University, RYMEC, Cantonment, Bellary, 583104, Karnataka, INDIA.
[*]Assistant Professor, Department of Computer Science and Engineering
Visveswaraya Technological University, RYMEC, Cantonment, Bellary, 583104, Karnataka, INDIA.

Abstract *— Cloud computing is a general term for the delivery of hosted services over the internet. Cloud Computing enable the customers with limited computational resources to outsource their data on to the Cloud where massive computational power can be easily utilized in a pay per-use manner. However, security is the major concern that prevents the wide adoption of computation outsourcing in the cloud, especially when end-user's confidential data are processed and produced during the computation. The outsourced computation workloads often contain sensitive information, such as the business financial records, proprietary research data, or personally identifiable health information etc. To fight against unauthorized information leakage, sensitive data have to be encrypted before outsourcing so as to provide end to- end data confidentiality assurance in the cloud and beyond.*

Keywords — *Cloud computing, security, outsourcing, encrypted.*

I. INTRODUCTION

Cloud Computing is the delivery of computing services such as servers, storage, databases, networking, software, analytics and more over the Internet ("the cloud").Cloud Computing provides convenient on-demand network access to a shared pool of configurable computing resources that can be rapidly deployed with great efficiency and minimal management overhead.one fundamental advantage of the cloud paradigm is computation outsourcing, where the computational power of customers cloud is no longer limited by their resource-constraint devices. By outsourcing the workloads into the cloud, customers could enjoy the literally unlimited computing resources in a pay-per-use manner without committing any large capital outlays in the purchase of hardware and software and/or the operational overhead. Despite having numerous benefits, users cannot outsource data on to the cloud securely, which inevitably brings in new security concerns and challenges towards this promising computing model. The outsourced computation workloads often contain sensitive information. To combat against unauthorized information leakage, sensitive data have to be encrypted before outsourcing so as to provide end-to-end data confidentiality assurance in the cloud and beyond.

A. *Types of cloud deployments*

1) *Public cloud*

Public clouds are owned and operated by a third party cloud service provider, which deliver their computing resources like servers and storage over the Internet. Microsoft Azure is an example of a public cloud. With a public cloud, all hardware, software and other supporting infrastructure is owned and managed by the cloud provider. You access these services and manage your account using a web browser.

2) *Private cloud*

A private cloud refers to cloud computing resources used exclusively by a single business or organisation. A private cloud can be physically located on the company's on-site data centre. Some companies also pay third-party service providers to host their private cloud. A private cloud is one in which the services and infrastructure are maintained on a private network.

3) *Hybrid cloud*

Hybrid clouds combine public and private clouds, bound together by technology that allows data and applications to be shared between them. By allowing data and applications to move between private and public clouds, hybrid cloud gives businesses greater flexibility and more deployment options.

B. *Types of cloud services*

1) *Software as a Service* (SAAS)

SAAS offers customers towards develop software applications on demand over the Internet which running on cloud infrastructure and accessible to client machine in the course of lines such as web browser. Services such as software for operating system, databases, servers, network access, power, and data centre space, etc. are contracted by the Cloud service provider.

2) *Platform as a Service* (PAAS)

PAAS provides a layer of application build up or an environment to develop software which is encapsulated and offered as a service. The user has the freedom to build his applications and it runs on the provider's high level infrastructure. Various features of PAAS are auto scaling, supports multiple hosting, extensibility, etc.

3) *Infrastructures as a Service* (IAAS)

IAAS specifies to the distribution of computing resources, basic storage and computing capabilities for executing services using virtualization technology. IAAS provides fundamental computing resources such as servers, storage system, network, etc.

II. PROPOSED MODEL

Ordinary data encryption techniques in essence prevent cloud from performing any meaningful operation of the underlying plaintext data, making the computation over encrypted data a very hard problem. On the other hand, the operational details inside the cloud are not transparent enough to customers. As a result, there exist various motivations for cloud server to behave unfaithfully and to return incorrect results, i.e., they may behave beyond the classical semi honest model. So we use Fully Homomorphism Encryption (FHE) scheme where the computation is represented by an encrypted combinational Boolean circuit that allows to be evaluated with encrypted private inputs.

A. *Design Goals*

To enable secure and practical outsourcing, the design mechanism should achieve the following security and performance guarantees.

1) ***Correctness***:

Any cloud server that faithfully follows the mechanism must produce an output that can be decrypted and verified successfully by the customer.

2) ***Soundness***:

No cloud server can generate an incorrect output that can be decrypted and verified successfully by the customer with non-negligible probability.

3) ***Input/output privacy***:

No sensitive information from the customer's private data can be derived by the cloud server during performing the LP computation.

4) ***Efficiency***: The local computations done by customer should be substantially less than solving the original LP on his own. The computation burden on the cloud server should be within the comparable.

B. *Module Description*

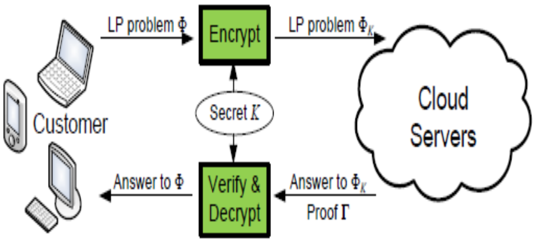

Fig 1: Frame work of fully homomorphic encryption

Fig 1 represents the framework of fully homomorphic encryption scheme. In this framework, the process on cloud server can be represented by algorithm Proof Gen and the process on customer can be organized into three algorithms (KeyGen, ProbEnc, and ResultDec). These four algorithms are explained below:

KeyGen (1k) → {K}. This is a randomized key generation algorithm which takes a system security parameter k, and returns a secret key K that is used later by customer to encrypt the target linear program problem.

ProbEnc (K, ϕ) → {ϕK}. This algorithm encrypts the input tuple ϕ into ϕK with the secret key K. According to problem transformation, the encrypted input ϕK has the same form as ϕ and thus defines the problem to be solved in the cloud.

ProofGen (ϕK) → {(y,Γ)}. This algorithm augments a generic solver that solves the problem ϕK to produce both the output y and a proof Γ. The output y later decrypts to x, and Γ is used later by the customer to verify the correctness of y or x.

ResultDec (K,ϕ, y, Γ) → {x,⊥}. This algorithm may choose to verify either y or x via the proof Γ. In any case, a correct output x is produced by decrypting y using the secret K. The algorithm outputs ⊥ when the validation fails, indicating the cloud server was not performing the computation faithfully.

1) Key Generation

This is a randomized key generation algorithm which takes a system security parameter k, and returns a secret key K that is used later by customer to encrypt the target linear programming problem.

2) Problem Encryption

This algorithm encrypts the input tuple Φ into Φk with the secret key K. According to problem transformation, the encrypted input Φk has the same form as Φ, and thus defines the problem to be solved in the cloud.

3) Proof Generation

This algorithm augments a generic solver that solves the problem Φ K to produce both the output y and a proof ┌. The output y later decrypts to x, and ┌ is used later by the customer to verify the correctness of y or x.

4) Key Description

The mechanism must produce an output that can be decrypted and verified successfully by the customer.

III. CONCLUSION

The problem of securely outsourcing data in cloud computing is formalised, and provides such a practical mechanism design which full-fills input/output privacy, cheating resilience, and efficiency. The Fully Homomorphism Encryption (FHE) scheme helps in the encryption and decryption of the data and keeping the data secure during the computation without any leakage or insecurities to the data hence keeping the user's information confidential and helping in secure outsourcing of data onto the cloud.

REFERENCES

[1] Naveen M, G Hemanth Kumar, "Secure and Practical Outsourcing of Linear Programming in Cloud Computing," IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661, p-ISSN: 2278-8727, Volume 19, Issue 5, Ver. I (Sep. - Oct. 2017), PP 22-26, DOI: 10.9790/0661-1905012226

[2] Lochan .B "Practical Outsourcing of Linear Programming in Secured Cloud Computing" Lochan .B / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 6 (1), 2015, 71-73

[3] Cong Wang, Kui Ren, and Jia Wan "Secure and Practical Outsourcing of Linear Programming in Cloud Computing", 978-1-4244-9921-/11/$26.00 ©2011 IEEE

[4] Naseer Amara, Huang Zhiqui, Awais Ali" Cloud Computing Security Threats and Attacks with their Mitigation Techniques", 978-1-5386-2209-4/17 $31.00 © 2017 IEEE DOI 10.1109/CyberC.2017.37

[5] G. Shanmugasundaram, V. Ashwini, G. Suganya, "A COMPREHENSIVE REVIEW ON CLOUD COMPUTING SECURITY", 978-1-5090-5/17 $31.00 © 2017 IEEE 2017 International Conference on Innovations in information Embedded and Communication Systems (ICIIECS)