**Secure Optimization Computation Outsourcing in Cloud Computing: A Case Study of Linear Programming**

**ABSTRACT:**

Cloud computing enables an economically promising paradigm of computation outsourcing. However, how to protect customers confidential data processed and generated during the computation is becoming the major security concern. Focusing on engineering computing and optimization tasks, this paper investigates secure outsourcing of widely applicable linear programming (LP) computations. Our mechanism design explicitly decomposes LP computation outsourcing into public LP solvers running on the cloud and private LP parameters owned by the customer. The resulting flexibility allows us to explore appropriate security/efficiency tradeoff via higher-level abstraction of LP computation than the general circuit representation. Specifically, by formulating private LP problem as a set of matrices/vectors, we develop efficient privacy-preserving problem transformation techniques, which allow customers to transform the original LP into some random one while protecting sensitive input/output information. To validate the computation result, we further explore the fundamental duality theorem of LP and derive the necessary and sufficient conditions that correct results must satisfy. Such result verification mechanism is very efficient and incurs close-to-zero additional cost on both cloud server and customers. Extensive security analysis and experiment results show the immediate practicability of our mechanism design.

**EXISTING SYSTEM:**

* Recent researches in both the cryptography and the theoretical computer science communities have made steady advances in “secure outsourcing expensive computations”.
* Based on Yao’s garbled circuits and Gentry’s breakthrough work on fully homomorphic encryption (FHE) scheme, a general result of secure computation outsourcing has been shown viable in theory, where the computation is represented by an encrypted combinational boolean circuit that allows to be evaluated with encrypted private inputs.
* Frikken give a provably secure protocol for secure outsourcing matrix multiplications based on secret sharing. While this work outperforms their previous work in the sense of single server assumption and computation efficiency (no expensive cryptographic primitives), the drawback is the large communication overhead. Namely, due to secret sharing technique, all scalar operations in original matrix multiplication are expanded to polynomials, introducing significant amount of overhead.

**DISADVANTAGES OF EXISTING SYSTEM:**

* Applying the existing mechanism to our daily computations would be far from practical, due to the extremely high complexity of FHE operation as well as the pessimistic circuit sizes that cannot be handled in practice when constructing original and encrypted circuits.
* In existing approaches, either heavy cloud-side cryptographic computations or multi-round interactive protocol executions, or huge communication complexities, are involved.
* In short, practically efficient mechanisms with immediate practices for secure computation outsourcing in cloud are still missing.

**PROPOSED SYSTEM:**

* In this paper, we study practically efficient mechanisms for secure outsourcing of linear programming (LP) computations. Linear programming is an algorithmic and computational tool which captures the first order effects of various system parameters that should be optimized, and is essential to engineering optimization.
* We propose to explicitly decompose the LP computation outsourcing into public LP solvers running on the cloud and private LP parameters owned by the customer.
* Specifically, we first formulate private data owned by the customer for LP problem as a set of matrices and vectors. This higher level representation allows us to apply a set of efficient privacy-preserving problem transformation techniques, including matrix multiplication and affine mapping, to transform the original LP problem into some random one while protecting the sensitive input/output information.

**ADVANTAGES OF PROPOSED SYSTEM:**

* It has been widely used in various engineering disciplines that analyze and optimize real-world systems/models, such as packet routing, flow control, power management of data centers, etc.
* The flexibility of such decomposition allows us to explore higher level abstraction of LP computations than the general circuit representation for the practical efficiency.
* For the first time, we formalize the problem of securely outsourcing LP computations, and provide such a secure and practical mechanism design which fulfills input/output privacy, cheating resilience, and efficiency.
* Our mechanism brings cloud customer great computation savings from secure LP outsourcing as it only incurs overhead on the customer, while solving a normal LP problem usually requires more time.
* The computations done by the cloud server shares the same time complexity of currently practical algorithms for solving the linear programming problems, which ensures that the use of cloud is economically viable.
* The experiment demonstrates the immediate practicality: our mechanism can always help customers achieve more than 50% savings when the sizes of the original LP problems (with feasible solutions) are not too small, while introducing no substantial overhead on the cloud.

**SYSTEM ARCHITECTURE:**



**MODULES**

* Customer
* Cloud
* Linear Programming Methodology
* Analysis on Input and Output Privacy

**MODULES DESCRIPTION**

**Customer:**

In this module, we develop the Customer features functionalities. Customer first register his/her details and login. Customer can outsource sensitive and valuable data to cloud using linear programming methodology with the matrix-matrix multiplications in problem encryption algorithm ProbEnc and file secret key automatically generate to his/her mail ID. They are can view the uploaded file details. If customer wants to download his/her file from a cloud by using the secret key of the file. If the is not match to the file means customer cannot able to download that file.

**Cloud:**

In this module, we design the Cloud functionalities. The Cloud entity can view all customer details, file upload details and customer file download details. In this module, we use the DriveHQ Cloud Service API for the Cloud Integration and develop the project.

**Linear Programming Methodology:**

Secure LP outsourcing in cloud can be represented by decomposing LP computation into public LP solvers running on the cloud and private data owned by the customer. Because different decompositions of LP usually lead to different trade-offs among efficiency and security guarantees, how to choose the right one that is most suitable for our design goal is thus of critical importance. To systematically study the difference, we organize the different decompositions into a hierarchy which ensembles the usual way that a computation is specified: a computation at a higher abstraction level is made up from the computations at lower abstraction levels. At higher abstraction levels, more information about the computations becomes public so that security guarantees become weaker. But more structures become available, and the mechanisms become more efficient. At lower abstraction levels, the structures become generic, but less information is available to the cloud so that stronger security guarantees could be achieved at the cost of efficiency.

Because we aim to design practically efficient mechanisms of secure LP outsourcing, we focus on the top level of the hierarchy. We will study problem transformation techniques that enable customers to secretly transform the original LP into some random one to achieve the secure LP outsourcing design.

**Analysis on Input and Output Privacy:**

We now analyze the input/output privacy guarantee under the aforementioned ciphertext only attack model. Specifically, the only information the cloud server obtains and the obvious fact that A and B of original LP problem are general full rank matrices. Note that in our model no secret transformation key shall be used twice. Offline guessing on problem input/output does not bring cloud server any advantage, since there is no way to justify the validity of the guess. We assume our system uses finite precision floating numbers, and each entry xi of the original solution x should be in range where L with k as our security parameter and poly as a polynomial function.

**SYSTEM REQUIREMENTS:**

**HARDWARE REQUIREMENTS:**

* System : Pentium Dual Core.
* Hard Disk : 120 GB.
* Monitor : 15’’ LED
* Input Devices : Keyboard, Mouse
* Ram : 1GB.

**SOFTWARE REQUIREMENTS:**

* Operating system : Windows 7.
* Coding Language : JAVA/J2EE
* Tool : Netbeans 7.2.1
* Database : MYSQL

**REFERENCE:**

Cong Wang, Member, IEEE, Kui Ren, Senior Member, IEEE, and Jia Wang, Member, IEEE, “Secure Optimization Computation Outsourcing in Cloud Computing: A Case Study of Linear Programming”, **IEEE TRANSACTIONS ON COMPUTERS, VOL. 65, NO. 1, JANUARY 2016.**