Available online at www.elixirpublishers.com (Elixir International Journal)

Computer Science and Engineering

Elixir Comp. Sci. & Engg. 73 (2014) 26546-26549

ISSN: 2229-712X

# Architecture for Secure Cloud Computing using Garbled Circuits

Sonali Lunawa* and Abhijit Patankar

Department of Computer Engineering, Pune University.

**ARTICLE INFO**

**Article history:**

Received: 22 June 2013;

Received in revised form:

18 August 2014;

Accepted: 27 August 2014;

**Keywords**

Cloud Computing,

Homomorphic Encryption,

Private Cloud,

Public Cloud.

**ABSTRACT**

Cloud computing security challenges for secure outsourcing of data and computations are increasing. Many hardware or fully Homomorphic Encryption solutions exists for Cloud computing security. The hardware based solution is not able to scale and fully Homomorphic Encryption is yet not practically used due to its low efficiency. Client should have trust service provider for confidentiality and integrity of their data and computation. In this paper, we describe architecture for secure outsourcing data and computation by providing confidentiality and integrity. In this architecture, Private Cloud which performs encryption and decryption of data and computations using Garbled Circuit for DNA matching using Levenshtein distance. Public Cloud does processing of operations. Comparison of our architecture with others techniques can be carried for proving its efficiency for secure outsourcing of data and computation. Efficient design can be used for different application where large mathematical computation is applied.



## Introduction

Cloud Computing is virtualized pool of resources in terms of hardware and software.

1. Resources are provided on metered basis hence users are charged only for resources which they use.
2. Provide rapid elasticity when demand changes for resources.
3. Maintenance and Security are ensured by Service Providers.
4. Provides Real-time monitoring of resources used.

## Different Deployment Models

1. Public cloud: In public cloud, multiple users can access the computing resources provided by a single service provider and pay only for operating resources.
2. Private cloud: In the private cloud, computing resources are used and controlled by a private organization. The main advantage of this it is high security, compliance and QoS are under the control of organization.
3. Hybrid cloud: A hybrid cloud is combines a public and private cloud. If organization has their own private cloud at peak time requirements increases so can afford for public cloud, then return if no longer needed.
4. Community cloud: Many of the organizations jointly share the same cloud infrastructure for some social purpose.

## Need of Security

The concept of cloud computing is to reduce the processing burden on the client by improving the ability of cloud to handle them, by using client as a simple in- put and output device, and bask of computations on the cloud. As per survey data security risk is as shows in Figure 1. Hence, data security has become the primary concern for people to shift to cloud computing.

Cloud computing is very cost effective and more flexible for the clients to work. As it has number of benefits still introduces security risks for organizations for which isolate their data from other cloud clients and to maintain confidentiality and integrity demands of their users.

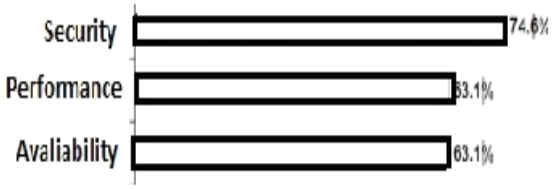


**Figure 1: Rate and Challenges graph for Cloud On-Demand**

Secure computation of arbitrary functions on encrypted data can be achieved based on fully homomorphic encryption. However, due to low efficiency this encryption technique is not yet usable in practice. Confidentiality and integrity of the outsourced data has to be protected as well as secure computations on it need to be performed in the Public cloud without interaction with the client. Many security challenges inspired researchers to find solution for maintaining user confidentiality and integrity for outsourcing data.

## Garbled Circuits

Garbled Circuits were introduced by Yao as a generic mechanism for secure computation. A standard garbled circuit protocol involves two parties which act as a semi-honest wish to compute the output of a function that depends on private data of both parties without revealing their private data. One party act as generator, who generates a garbled circuit for computing the function and the other party is the evaluator, evaluates that circuit to produce encrypted result, which can then be work semi honestly. Garble Circuits is more flexible and cheap programming, due to its low cost of manipulation of boolean values. Garbled Circuits are designed by converting operands and operators to gates as AND, OR, NOT, etc

## Eucalyptus Fast Start

Eucalyptus launched an updated version as Fast Start installer. Fast Start is a self-contained installer for Eucalyptus. This installation includes two machines, one “Front-End” machine and one or more node controllers. “Cloud-in-a-Box” is can be installed on a single machine.

Tele:

E-mail addresses: sonali.lunawat@gmail.com

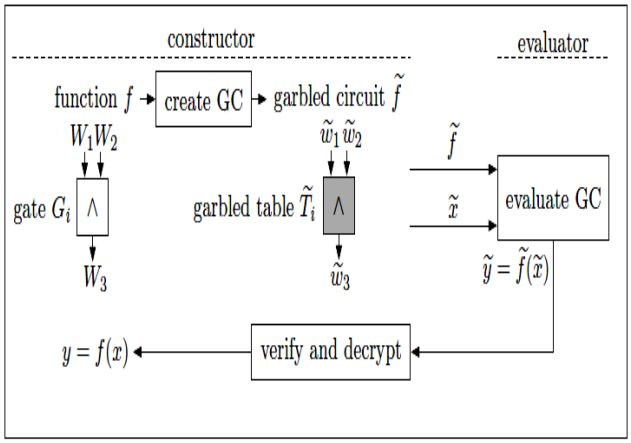


**Figure 2: Overview of Garbled Circuit [7]**

The Eucalyptus Cloud can dynamically scale up or down depending workloads, highly efficient scalability, and increased trust and control for IT. The Eucalyptus cloud computing is open source has five components: Cloud Controller (CLC), Cluster Controller (CC), Walrus, Storage Controller (SC) and Node Controller (NC).

*Cloud Controller (CLC)* is responsible for managing the virtualized resources via different APIs.

*Walrus* implements scalable bucket storage. The implementation of Walrus is same as Amazons S3 (Simple Storage Service), providing persistent storage images and data.

*Cluster Controller (CC)* controls the execution of virtual machines VM running on the nodes and manages its networking between external users.

*Storage Controller (SC)* provides network storage that can be dynamically attached by VM.

*Node Controller (NC)* acts as hypervisor which controls VM activities as execution, inspection, and termination of VM instances.

**Dna Matching**

Dna Matching is a buzzword in computational molecular biology. The DNA matching can be done using a set of strings having a sequence of symbols of four alphabets A, C, G and T as composed of nucleotides where Adenine (A), Cytosine (C), Guanine (G), and Thymine (T). DNA matching plays a vital role in various applications for data analysis related to protein and gene. In this paper, we focus on architecture for secure Cloud Computing for outsourcing data and computations. This system finds DNA match using Levenshtein distance which is used to find minimum number insertion, deletion between two strings to get exact match. It is implemented using below expression

$$Lev_{a,b} \mid a \mid\mid b \mid \text{ where}$$

$$Lev_{a,b} = \begin{cases} 0 & i = j = 0 \\ i & i > 0 \\ j & j > 0 \\ mid & \begin{cases} Lev_{a,b}(i-1,j)+1, \\ Lev_{a,b}(i,j-1)+1, else \\ Lev_{a,b}(i-1,j-1)+[a_i \neq b_j] \end{cases} \end{cases}$$

**Related Work**

Security Modules (HSM) or Smartcards design by ARM provide a secure execution environment to execute clients programs. However, cryptographic co-processors are very expensive and do not scale well. Homomorphic Encryption allows computing on encrypted data without using additional helper information and they are restricted for specific operations like addition or multiplication. Also, they are not suitable for branching of encrypted values. Recently, proposed fully Homomorphic Encryption is not yet been used for practical operation as low efficiency. So, Encryption alone is not sufficient for outsourcing data which is shared among cloud. Token based Cloud Computing [8] approach combines a tamper-proof hardware token T used in the setup phase only with efficient computations performed in parallel in the computation cloud. The basic idea is that T generates a garbled circuit during the setup phase and in the time-critical online phase the garbled circuit is evaluated in parallel by the computation cloud. In this approach it requires trust in hardware which can face many challenges of attestation. To improve the efficiency of the secure computation, model additionally allows that uses a tamper-proof hardware token T, integrated within infrastructure, for performing computations within a shielded environment, must provide guarantee not to leak any information. As T needs to be built tamper-proof and cost-effective, it will have a restricted amount of memory only. In many cases the available memory within T will not be sufficient to store or intermediate values during evaluation. The token T could be instantiated with a cryptographic coprocessor built by a third-party manufacturer in a way that T does not leak any information. Authors of this paper have done qualitative performance comparison of the proposed architectures and leave a prototype implementation for their quantitative performance comparison as future work. To achieve trustworthy computations in cloud infrastructures is to adapt existing trusted computing solutions to the cloud computing use these solutions as building blocks in new cloud architecture models [2]. A traditional trusted computing architecture can provide some-degree security for the customers. Such system can forbid the owner of a host to interfere all the computation. The customer can also run the remote testing program which can let the customer know if the procedure of the host is secure or not. If the users or customers detect any kind of abnormal behaviour from the host, they can immediately terminate their VMmachines. Unfortunately, such apparent perfect platforms also have some fatal flaws. For example, as we know, the service providers always provide a list of available machines to the customers. Afterwards, the customer will be automatically assigned a machine. However such dynamical assigned machine from the provider backend can incur some kinds of security threat which cannot be solved by such system. The most prominent approach to Trusted Computing technology has been specified by the Trusted Computing Group (TCG). The TCG proposes to extend common computing platforms with trusted components in soft- ware and hardware. The Trusted Computing is based on a smart design which uses Trusted Platform Module (TPM) chip which has a kind of private key (EK). It can also have some problems if the attackers use the playback attack. Twin Cloud architecture [1] for secure outsourcing of data and computations. This architecture can be extended to multiple clients operating on same data. In this approach client provides the data D to outsource as well as the program P (formulated by Hardware Description Language (HDL)) to compute. Divides architecture in Setup and Query phase for ease of communication. In this architecture compilation of garbled circuit is done using circuit compiler as Fairplay.

**Problem Statement**

Our architectural setup will have Private Cloud and Public Cloud.

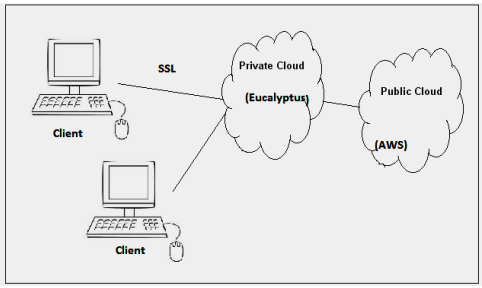


**Figure 3: Architecture for secure Cloud Computing**

Firstly, we will implement an encryption algorithm RSA which is named after Ron Rivest, Adi Shamir and Len Adleman for proving as Homomorphic Encryption. With architectural setup in our system client will send its information i.e. large string data in GBs for DNA matching. The string is converted in Hex code. In this number of edits are found to match string done by using Levenshtein Distance. Then data will be passed through secure channel to Private Cloud act as *generator* will encrypt the data by Garbled Circuit. After verifying it and the encrypted data will be passed to Public Cloud for performing non-critical operations.

Result of match is passed to Private Cloud *evaluator* for decryption and then match string output is given to client. Private Cloud is used as proxy between the client and the Public Cloud. The Private Cloud provides a execution for encryption and decryption of client data using Garbled Circuit due to which fully trusted by clients.

**Objectives**

To provide a solution that proves for high security. To provide the system which will be implemented with minimum cost and prove efficient in time. We are implementing setup using all open source software and Fastgc framework for designing Garbled circuit which will prove efficient in time and cost from the existing approach.

**Data Flow Architecture**

Data Flow Architecture of our system viewed as a extension to system. In this system Private Cloud is design by Eucalyptus and Commodity using Amazon Web Services (AWS).

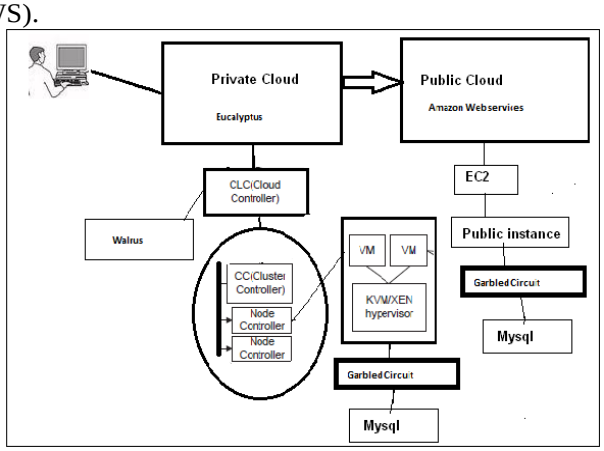


**Figure 4: Data Flow Architecture**

**Experimental Setup**

Private Cloud: We have made setup using Fast Start Eucalyptus having on machine as Cloud–in-box and 2 node controllers by using router for static ip address.

**Public Cloud: Created an account with AWS.**

Input: Client will give input to the Private Cloud as DNA sequence for matching. In Public Cloud we will
stored around 20 sample DNA string on AWS instance using mysql database having size up to 10,000.

E.g. AGGCCAGGCGGTCGACTTTTTCACTGGGT GTCTCACTTG ATCGCCACAG AAATTGACGA GAAATTCAGT GAAGTTGCGT TCAGAGATGT TCAGTATAATTGATATTTCCGATAGTACCACAGGTAAT TATTAAATGTGCAAAATCTGCTATTAATTTAGCACTGC TACATTGGTCTGCTAAATATTCATACCTAACTTGGGTC TGCTAAATATTTACAGCCATCCCCATTGTCTGCTAATC AGTATAATTGATATTTCCGATAGTACCACAGGTAATTA TTAAATGTGCAAAATCTGCTATTAATTTAGCACTGCTA CATTGGTCTGCTAAATATTCA

Output: If match found then give result of match found and the time required to processing Garbled Circuit.

**Results and Discussion**


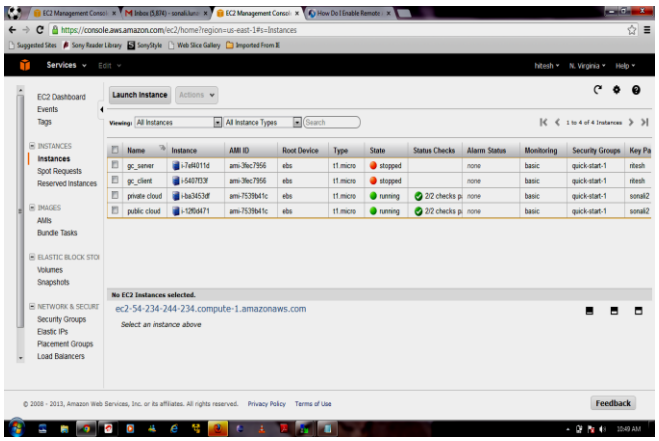

**Figure 5: Snapshot of Experimental setup**

Result of RSA using this system for Homomorphic Encryption is proved by deploying a server and client code on two different running instances.

**Expected results: <u>Private Cloud</u>**

Time (ms) Starting program: 1371533788841 (0.0, 0.0)
Enter the String: AGCT
Inserted data successfully
Hex Conversion of given string::4147c3d4
Waiting for client to connect
Client has connected 1
Elapsed time (ms) on circuit preparation: 15703 (0.009765625, 0.015625)
Elapsed time (ms) on OT preparation: 1455 (13.1025390625, 8.9716796875)
Elapsed time (ms) on sending labels for self's inputs: 10 (7.8515625, 0.0)
Elapsed time (ms) on sending labels for peers inputs: 84 (15.703125, 22.197265625)
Elapsed time (ms) on circuit garbling: 36672 (12594.375, 0.0)
output (pp): 0
Elapsed time (ms) on output labels received and interpreted: 1 (0.0, 0.3310546875)

output (verify): 0
Match Foun

**Public Cloud**

Time (ms) Starting program: 1371533803843 (0.0, 0.0)
Matched String = AGCT
Hex Conversion of given string::4147c3d4
1
Elapsed time (ms) on circuit preparation: 704 (0.0, 0.0)
Time (ms) right before NPOT public key generation: 1371533804565 (0.0, 0.0)
Elapsed time (ms) on NPOT public key generation: 63 (0.0, 0.0)
Elapsed time (ms) on OT preparation: 1367 (0.0, 0.0)
Elapsed time (ms) on receiving labels for peer's inputs: 24 (0.0, 0.0)
Elapsed time (ms) on receiving labels for self's inputs: 85 (0.0, 0.0)
Match Found

By using time constraint for similar large character sequence we will prove it efficiency on Hybrid Cloud as secure method for large computational application

**Conclusion**

The security of our architecture is maintained as Private Cloud has control over Public Cloud. The DNA matching system is highly secured can be used in many application of DNA matching. The Public Cloud is neither able to successfully modify nor to learn the outsourced data as these are verified and in encrypted. Client is transparent to the system.

Our architecture can be extended by combining benefits of Garbled Circuits and fully Homomorphic Encryption. Application handling large mathematical computation can be developed to make use of Garbled Circuits and above architecture for better security.

**References**

[1] Sven Bugiel, Stefan Nurnberger, Ahmad Reza Sadeghi and Thomas Schneider, "Twin Clouds: Architecture for Secure Cloud Computing", *Center for Advanced Security Research Darmstadt, Technische University Darmstadt, Germany*, 2010.

[2] Fei Hu, Meikang Qiu, "Review on Cloud Computing: Design Challenges in Architecture and Security", *Journal of Computing and Information Technology - CIT'19*, 2011.

[3] Jianfeng Yang Zhibin, "Cloud Computing Research and Security Issues", IEEE, 2009

[4] R. M. Savola, A. Juhola, I. Uusitalo, "Towards wider cloud service applicability by security, privacy and trust measurements", *International Conference on Application of Information and Communication Technologies (AICT)*, Oct 2010.

[5] Minqi Zhou, Rong Zhang, WeiXie, Weining Qian, Aoying Zhou, "Security and Privacy in cloud Computing: A Survey Quality", *Proc .IEEE Int. Conf. on Educational and Information Technology*, 2010.

[6] K.Jarvinen, V.Kolesnikov, A R Sadeghi and T. Schneider, "Garbled circuits for Leakage-resilience: Hardware implementation and evaluation of one-time programs", *In Cryptographic Hardware and Embedded Systems Springer*, 2010.

[7] A.R Sadeghi and T. Schneider, " Token based Cloud Computing secure out- sourcing of data and arbitrary computations with lower latency", *In Trust and Trust worthy Computing (TRUST'10)workshop on Trust in the Cloud, Springer*, 2010

[8] Y.Huang, D. Evans, J. Katz, and L. Malka, "Faster secure two-party computation using Garbled circuits", *In USENIX Security Symposium*, 2011.

[9] N. Santos, K. P. Gummadi, and R. Rodrigues "Towards trusted cloud computing", *In Hot Topics in Cloud Computing ( HotCloud'09).USENIX Associa- tion*, 2009