Computation on Encrypted Data via Cloud based Outsourcing

B.Tech. Minor Project Report

BY

HARSH PATHAK (14115036)

VIPUL BAJAJ (14115904)

N. POOJA (14115062)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY

RAIPUR, C.G. (INDIA)

DECEMBER, 2017

COMPUTATION ON ENCRYPTED DATA VIA CLOUD BASED OUTSOURCING

Minor Project Report

"Submitted in the partial fulfillment of the Requirements for the award of degree

Of

Bachelor of Technology

In

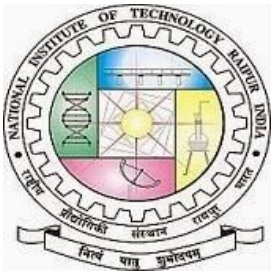
Computer Science and Engineering"

By

HARSH PATHAK (14115036)

VIPUL BAJAJ (14115904)

N. POOJA (14115062)



DEPARTMENT OF COMPUTER SC. & ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY
RAIPUR , C.G. (INDIA)

DECEMBER, 2017

DECLARATION

I hereby declare that the project "Computation on Encrypted Data via Cloud based outsourcing" submitted in the partial fulfillment of the award of the degree of B.Tech in Computer Science and Engineering Department, NIT Raipur is original work and the project has not formed the basis for the award of any other degree associateship, fellowship or any other similar titles.

Signature
HARSH PATHAK

Signature
VIPUL BAJAJ

Signature
N. POOJA

Place : Raipur
Date : 11th Dec 2017

CERTIFICATE

I hereby certify that the work which is being presented in the B.Tech. Minor Project Report entitled *"**Computation on Encrypted Data via Cloud based outsourcing**"* in partial fulfillment of the requirements for the award of the **Bachelor of Technology in Computer Sc. & Engineering** and submitted to the Department of Computer Sc. & Engineering of National Institute of Technology Raipur is an authentic record of my own work carried out during a period from June 2017 to December 2017 under the supervision of **DR. MANU VARDHAN ,CSE Department**.

The matter presented in this thesis has not been submitted by me for the award of any other degree elsewhere.

Signature of Candidate

HARSH PATHAK

Roll no 14115036

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Signature of Supervisor

Date: 11th DEC 2017

Dr. MANU VARDHAN
Assistant Professor (CSE)

Dr. DILIP SINGH SISODIA
HEAD OF COMPUTER SCIENCE & ENGINEERING DEPARTMENT
NATIONAL INSTITUTE OF TECHNOLOGY RAIPUR CG.

CERTIFICATE

I hereby certify that the work which is being presented in the B.Tech. Minor Project Report entitled *"**Computation on Encrypted Data via Cloud based outsourcing**"* in partial fulfillment of the requirements for the award of the **Bachelor of Technology in Computer Sc. & Engineering** and submitted to the Department of Computer Sc. & Engineering of National Institute of Technology Raipur is an authentic record of my own work carried out during a period from June 2017 to December 2017 under the supervision of **DR. MANU VARDHAN ,CSE Department**.

The matter presented in this thesis has not been submitted by me for the award of any other degree elsewhere.

Signature of Candidate

VIPUL BAJAJ

Roll no 14115904

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Signature of Supervisor

Date: 11th DEC 2017

Dr. MANU VARDHAN
Assistant Professor (CSE)

Dr. DILIP SINGH SISODIA
HEAD OF COMPUTER SCIENCE & ENGINEERING DEPARTMENT
NATIONAL INSTITUTE OF TECHNOLOGY RAIPUR CG.

CERTIFICATE

I hereby certify that the work which is being presented in the B.Tech. Minor Project Report entitled *"**Computation on Encrypted Data via Cloud based outsourcing**"* in partial fulfillment of the requirements for the award of the **Bachelor of Technology in Computer Sc. & Engineering** and submitted to the Department of Computer Sc. & Engineering of National Institute of Technology Raipur is an authentic record of my own work carried out during a period from June 2017 to December 2017 under the supervision of **DR. MANU VARDHAN ,CSE Department**.

The matter presented in this thesis has not been submitted by me for the award of any other degree elsewhere.

Signature of Candidate

N. POOJA

Roll no 14115062

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Signature of Supervisor

Date: 11th DEC 2017 ***Dr. MANU VARDHAN***

Assistant Professor (CSE)

Dr. DILIP SINGH SISODIA
HEAD OF COMPUTER SCIENCE & ENGINEERING DEPARTMENT
NATIONAL INSTITUTE OF TECHNOLOGY RAIPUR CG

ABSTRACT

Computation outsourcing to the cloud has become a popular application in the age of cloud computing. This computing paradigm brings in some new security concerns and challenges, such as input/output privacy and result verifiability. Matrix multiplication computation (MMC) is a common scientific and engineering computational task. But such computation involves enormous computing resources for large matrices, which is burdensome for the resource-limited clients.So we are motivated to design a protocol to enable secure, robust cheating resistant, and efficient outsourcing of MMC to a malicious cloud.

The main idea to protect the privacy is employing some transformations on the original MMC problem to get an encrypted MMC problem which is sent to the cloud; and then transforming the result returned from the cloud to get the correct result to the original MMC problem.The scheme allows to securely compute multiplication of a secret matrix with a known public matrix. And thus this solution provides an efficient and secure way of outsourcing the matrices for computation.

ACKNOWLEDGEMENT

We would like to thank our project advisor, **DR MANU VARDHAN** , Assistant Professor, Dept of CS&E, NIT Raipur, for introducing this project and for his inspiring guidance, constructive criticism, valuable suggestions and support. We also thank him for his constant feedback and expertise that paved a way forward for us.

We would also like to thank our Head of Dept. **Dr Dilip Singh Sisodia** for providing us with the necessary facilities and constant guidance and motivation.

We would also like to thank **Dr A M Rawani** , Director , NIT Raipur for providing us with all the necessary facilities and infrastructure .

HARSH PATHAK (14115036)
VIPUL BAJAJ (14115904)
N. POOJA (14115062)

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

INTRODUCTION

Cloud computing is the practice of using a network of remote servers hosted on the Internet to store, manage, and process data, rather than a local server or a personal computer. Clouds are becoming really popular and with Big giant service providers like google , microsoft , alibaba and many others providing us platforms for cloud computing at an substantial cheaper rate. There is a revolutionary change being brought up not only on Academic but also on the way business are done. Knowledge process outsourcing (KPO) has been given an enormous boost because of the cloud services available. Now there is no need to set up you own infrastructure rather you can rely on these service providers.

In mathematics, a matrix (plural: matrices) is a rectangular array of numbers, symbols, or expressions, arranged in rows and columns. Matrices have a wide variety of applications Matrix mathematics applies to several branches of science, as well as different mathematical disciplines. Matrix multiplication is used to solve system of linear equations , used in linear regression and even encryption.

1. Challenges

All of this sounds really promising but then the questions arises how much can we trust such systems ? Are they completely secure ? Can processing of confidential data be done on such platforms ? So in short the challenge is how to maintain Data confidentiality (Input/Output).

The next challenge that we can face is the verification of the output that we will receive. Also the computations must be quick and efficient as possible.

2. Solutions

One way to ensure that our data is confidential is by using Cryptography. We can encrypt our data and send it to cloud. Although Encrypted data loses its meaning and

is gibberish, and mathematical computations are useless on such ciphertext , we must find such an encryption scheme such that somehow we are able to do meaningful computations. One proposed solution is the use of matrix multiplication for encryption

3. Preliminaries

System Model

Clouds can be broken theoretically into 3 categories

1. **Secure and Trustworthy cloud** - Suppose you own your own cloud service , then you can be pretty much sure that the data will be safe and you won't need any encryption as such.

   But most of us don't have a cloud service of our own and rely on 3rd party service providers .
2. **Semi Secure cloud -** Such clouds can neither be trusted or distrusted. They lie somewhere in between the grey zone. They may or may not collect read or process your data
3. **Untrusted Cloud -** Such type of cloud services may process , store or manipulate your data and hence data must be encrypted properly to secure the systems from such systems.

Working of the system model

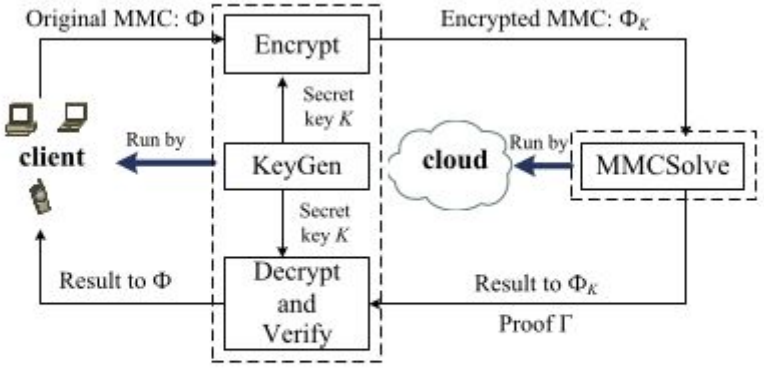


Fig 1. Secure MMC Outsourcing Model

Client side will generate keys(here matrices) and use those keys to encrypt the data matrix. It will then outsource the encrypted matrix to the cloud system.Cloud side will perform the computation and give back the result to the client side.When the cloud will sent back the result, Client side will decrypt and verify the result.

CHAPTER 2
LITERATURE REVIEW

LITERATURE REVIEW

With the rapid development in availability of cloud services, the techniques for securely outsourcing the prohibitively expensive computations to untrusted servers are getting more and more attentions in the scientific community.

For the first time, we utilize the sparse matrix to propose a new secure outsourcing for Matrix Multiplication on Encrypted Data via Cloud based outsourcing algorithm of large-scale linear equations in the fully malicious model.Furthermore, the client can detect the misbehaviour of cloud server.

Cloud computing has plenty of benefits for real-world applications. In the outsourcing computation paradigm, the users with resource-constraint devices can outsource heavy computation workloads into the cloud server and enjoy the unlimited computing resources in a pay-per use manner.

Despite the tremendous benefits, outsourcing computation also inevitably suffers from some new security challenges

- Firstly, the computation tasks often contain some sensitive information that should not be exposed to the (semi-trusted) cloud servers.Therefore, the first security challenge is the privacy of the outsourcing computation: the curious cloud servers should not learn anything about what it is actually computing (including the secret inputs and outputs).
- The second security challenge is the checkability of the outsourcing computation the outsourcer should have the ability to detect any failures if the cloud servers misbehave.

Next, we describe Procedure for MMC-Encryption,

- The client generates matrices R1,R2,R3,R4 The client computes E1 E2. Later this encrypted matrices will be outsourced to the cloud.
- The cloud performs matrix multiplication on the encrypted matrices outsourced from the client i.e. compute Z= E1*E2 and cloud then sends matrix Z back to the client.
- On receiving the returned matrix Z from the cloud, the client compute the Decrypted Resultant Matrix Res, thus the client can use efficiently (via time $O(N^2)$) to perform matrix multiplication over an unsecure cloud.

Therefore, We propose a new privacy-preserving algorithm for outsourcing matrix multiplication computation (MMC) to the cloud. By delegating the most expensive computation of MMC to the cloud, our algorithm relieves the client of its high computation burden. Moreover, with a series of carefully-designed random matrices, our algorithm can properly protect the privacy of input/output data of outsourced MMC

CHAPTER 3

STEPS INVOLVED AND DESIGN OF THE SYSTEM

1. *Encryption Scheme*

Let us look at the scenario once more. At the Client's end we have the Data matrices D1 and D2 and we want to encrypt those matrices. For this we will be generating few random sparse matrices. We need sparse matrices because multiplication of sparse matrices can be done in order of $O(n^2)$ [2]. Also we have to make sure that the random sparse matrices matrices should be invertible and their inverse must also be a sparse matrix.

Given : Data Matrices D1 and D2

Goal : Encrypted matrices E1 and E2

Steps for reaching our goals

Step 1) Generate 5 Random matrices[1] R1 , R2 , R3 and R4.*

Step 2) Matrix multiplication of both the sides.

$A := R1 * D1 * R2$

$B := R2^{-1} * D2 * R3$

Step 3) We must now add R4 to matrices A and B . [3]

$E1 := A + R4$

$E2 := B + R4$

(We need to do this because - if D1 and D2 are extremely sparse matrices then the encryption based simply on multiplication would not work because anything multiplied with zero will be zero and hence un encrypted.

Hence we add matrices R4 to A and B Respectively)

You can see from the example given below that Matrix E1 is more Encrypted than A.

[1] These matrices should be Sparse and Invertible . Inverse of these matrices must also be a Sparse matrix

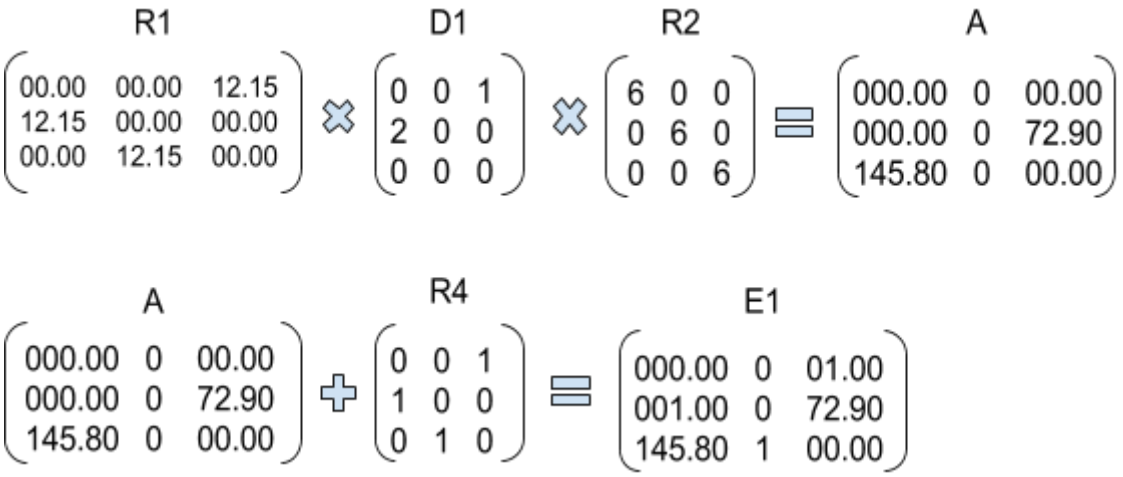


Fig 2 Encryption of Data Matrix D1

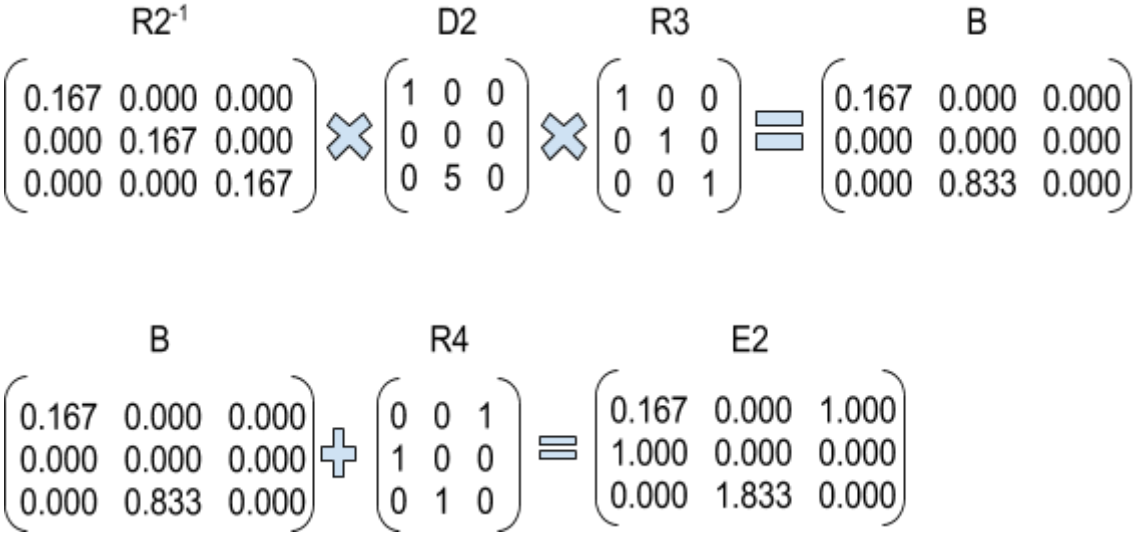


Fig 3 Encryption of Data MAtrix D2

Thus here we get the Data Matrices D1 and D2 encrypted as E1 and E2 respectively. And this matrices i.e. E1 and E2 are outsourced to cloud for computation.

2. *Decryption Scheme*

After getting the resultant matrix (Z = E1*E2) from cloud, the client will perform the decryption of the result. The client will use the generated random matrices (i.e Keys used for encryption) to decrypt and get the result.

Given : Resultant Matrix Z.

Goal : Decrypted Resultant Matrix Res.

Steps for reaching our goals

Step 1) Compute $R1^{-1}$ and $R3^{-1}$ from the pre generated matrices (i.e. matrices generated at encryption time).

Step 2) Compute the intermediate temporary matrix T as follows

$T := R4*R4 + A*R4 + R4*B$.

Step 3) Compute the decrypted resultant matrix Res by multiplying (Z-T) with $R1^{-1}$ and $R3^{-1}$ as follows.

$Res := R1^{-1} * (Z-T) * R3^{-1}$.

The following examples shows how decryption works at client side.

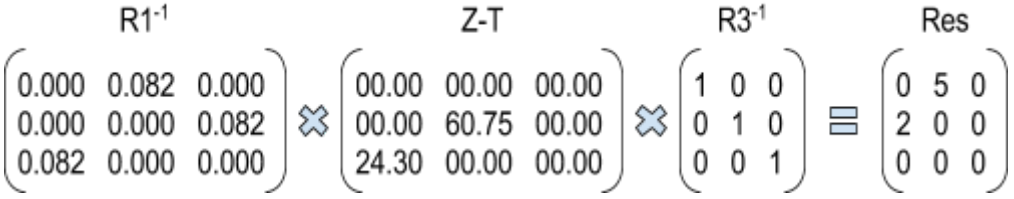


Fig 6 Decryption Example

Here Res is the decrypted result which should be equal to product of original matrices (D1 and D2).

3. *Verification*

After getting the decrypted result matrix (Res) ,at the client will perform the verification of the result. The client will use a new column matrix (X) for the verification which will be of dimension of nx1 where n is the dimension of the resultant matrix.

Given : Decrypted Result (Res) and Original Multiplication Result (D1*D2).

Goal :To verify whether the decrypted result matrix is same as the original multiplication result matrix.

Steps for reaching our goals

Step 1) Multiply the decrypted result matrix and the original multiplication result matrix with generated column matrix as follows.

$Y = Res * X$

$Y' = (D1 * D2) * X$

Step 2) Compare Y and Y' ,if Y and Y' are equal then the multiplication of encrypted matrices is successful else unsuccessful.

The following examples shows how verification works at client side.

$$\underset{Res}{\begin{pmatrix} 0 & 5 & 0 \\ 2 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}} \times \underset{X}{\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}} = \underset{Y}{\begin{pmatrix} 0 \\ 2 \\ 0 \end{pmatrix}}$$

$$\underset{D1*D2}{\begin{pmatrix} 0 & 5 & 0 \\ 2 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}} \times \underset{X}{\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}} = \underset{Y'}{\begin{pmatrix} 0 \\ 2 \\ 0 \end{pmatrix}}$$

Fig 7 Verification Example

SCREENSHOTS

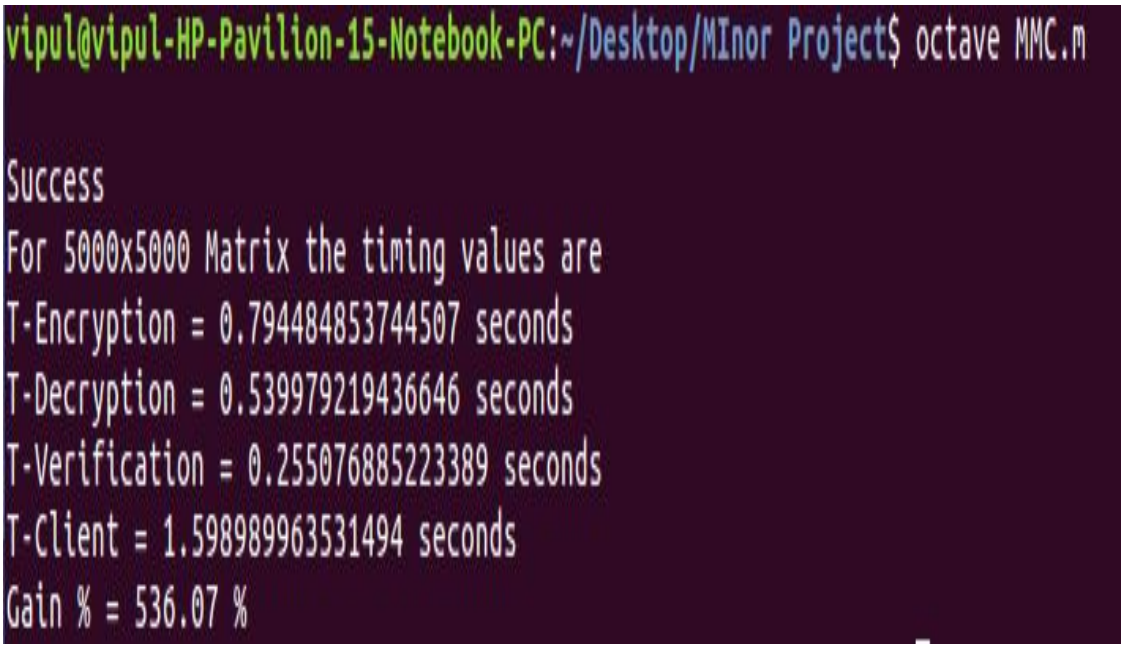


Fig 6 Result at Client Side

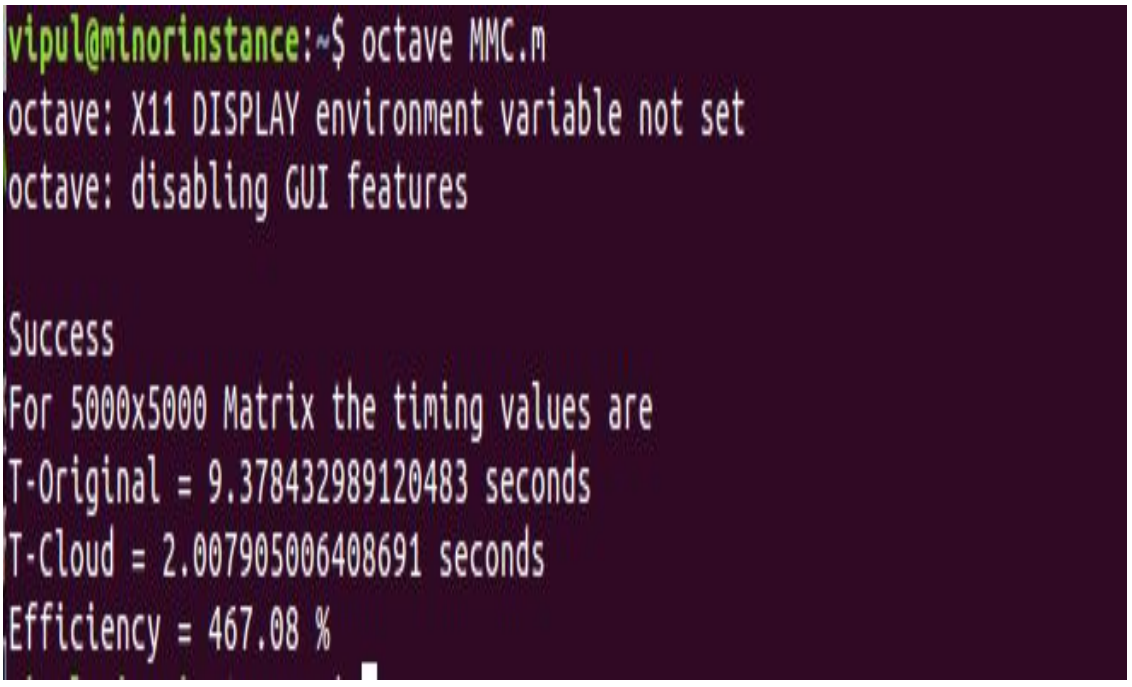


Fig 7 Result at Cloud Side

CHAPTER 4
RESULTS OBTAINED

RESULTS

**** CLIENT SIDE ****

Client Specs:

No. of Processors : 4

RAM : 15GB

Storage : 50GB of HDD

Success

For 500x500 Matrix the timing values are

T-Original = 0.004136085510254 seconds

T-Encryption = 0.012278079986572 seconds

T-Decryption = 0.012042999267578 seconds

T-Verification = 0.001251935958862 seconds

T-Client = 0.025573015213013 seconds

Gain % = 16.17 %

Success

For 1000x1000 Matrix the timing values are

T-Original = 0.022428035736084 seconds

T-Encryption = 0.040571212768555 seconds

T-Decryption = 0.054660081863403 seconds

T-Verification = 0.004900932312012 seconds

T-Client = 0.100132226943970 seconds

Gain % = 22.40 %

Success

For 1500x1500 Matrix the timing values are

T-Original = 0.070749998092651 seconds

T-Encryption = 0.088316202163696 seconds

T-Decryption = 0.134612083435059 seconds

T-Verification = 0.012101173400879 seconds

T-Client = 0.235029458999634 seconds

Gain % = 30.10 %

Success

For 2000x2000 Matrix the timing values are

T-Original = 0.164494037628174 seconds

T-Encryption = 0.172270059585571 seconds

T-Decryption = 0.228874206542969 seconds

T-Verification = 0.020723104476929 seconds

T-Client = 0.421867370605469 seconds

Gain % = 38.99 %

Success

For 2500x2500 Matrix the timing values are

T-Original = 0.573264837265015 seconds

T-Encryption = 0.306514024734497 seconds

T-Decryption = 0.452462196350098 seconds

T-Verification = 0.044807910919189 seconds

T-Client = 0.803784132003784 seconds

Gain % = 71.32 %

Success

For 3000x3000 Matrix the timing values are

T-Original = 0.985081195831299 seconds

T-Encryption = 0.446721076965332 seconds

T-Decryption = 0.600624799728394 seconds

T-Verification = 0.058194160461426 seconds

T-Client = 1.105540037155151 seconds

Gain % = 89.10 %

Success

For 3500x3500 Matrix the timing values are

T-Original = 1.544623851776123 seconds

T-Encryption = 0.561305284500122 seconds

T-Decryption = 0.826658964157104 seconds

T-Verification = 0.088337898254395 seconds

T-Client = 1.476302146911621 seconds

Gain % = 104.63 %

Success

For 4000x4000 Matrix the timing values are

T-Original = 2.278019905090332 seconds

T-Encryption = 0.682734966278076 seconds

T-Decryption = 1.046057939529419 seconds

T-Verification = 0.102581024169922 seconds

T-Client = 1.831373929977417 seconds

Gain % = 124.39 %

Success

For 4500 x 4500 Matrix the timing values are

T-Original = 3.273758172988892 seconds

T-Encryption = 0.922335147857666 seconds

T-Decryption = 1.465464830398560 seconds

T-Verification = 0.142386198043823 seconds

T-Client = 2.530186176300049 seconds

Gain % = 129.39 %

Success

For 5000x5000 Matrix the timing values are

T-Original = 4.405617952346802 seconds

T-Encryption = 1.145731925964355 seconds

T-Decryption = 1.757513999938965 seconds

T-Verification = 0.173759937286377 seconds

T-Client = 3.077005863189697 seconds

Gain % = 143.18 %

Success

For 5500x5500 Matrix the timing values are

T-Original = 5.098109960556030 seconds

T-Encryption = 1.304455995559692 seconds

T-Decryption = 2.065148115158081 seconds

T-Verification = 0.216340780258179 seconds

T-Client = 3.585944890975952 seconds

Gain % = 142.17 %

Success

For 6000x6000 Matrix the timing values are

T-Original = 7.603370904922485 seconds

T-Encryption = 1.508707046508789 seconds

T-Decryption = 2.260138988494873 seconds

T-Verification = 0.237186908721924 seconds

T-Client = 4.006032943725586 seconds

Gain % = 189.80 %

Success

For 6500x6500 Matrix the timing values are

T-Original = 9.724493026733398 seconds

T-Encryption = 1.744881153106689 seconds

T-Decryption = 2.708001852035522 seconds

T-Verification = 0.301161050796509 seconds

T-Client = 4.754044055938721 seconds

Gain % = 204.55 %

Success

For 7000x7000 Matrix the timing values are

T-Original = 12.128478050231934 seconds

T-Encryption = 2.023494005203247 seconds

T-Decryption = 2.993271827697754 seconds

T-Verification = 0.322830200195312 seconds

T-Client = 5.339596033096313 seconds

Gain % = 227.14 %

Success

For 7500x7500 Matrix the timing values are

T-Original = 14.851166963577271 seconds

T-Encryption = 2.310668945312500 seconds

T-Decryption = 3.626542091369629 seconds

T-Verification = 0.366325139999390 seconds

T-Client = 6.303536176681519 seconds

Gain % = 235.60 %

Success

For 8000x8000 Matrix the timing values are

T-Original = 17.908633947372437 seconds

T-Encryption = 2.683721065521240 seconds

T-Decryption = 3.916908979415894 seconds

T-Verification = 0.403802871704102 seconds

T-Client = 7.004432916641235 seconds

Gain % = 255.68 %

Success

For 8500x8500 Matrix the timing values are

T-Original = 21.633709907531738 seconds

T-Encryption = 2.952996253967285 seconds

T-Decryption = 4.414504051208496 seconds

T-Verification = 0.458849906921387 seconds

T-Client = 7.826350212097168 seconds

Gain % = 276.42 %

Success

For 9000x9000 Matrix the timing values are

T-Original = 25.396991014480591 seconds

T-Encryption = 3.608202934265137 seconds

T-Decryption = 5.479771137237549 seconds

T-Verification = 0.616723060607910 seconds

T-Client = 9.704697132110596 seconds

Gain % = 261.70 %

Success

For 9500x9500 Matrix the timing values are

T-Original = 29.452494144439697 seconds

T-Encryption = 4.076701879501343 seconds

T-Decryption = 6.143598794937134 seconds

T-Verification = 0.636761903762817 seconds

T-Client = 10.857062578201294 seconds

Gain % = 271.27 %

Unsuccessful

For 10000x10000 Matrix the timing values are

T-Original = 34.963022947311401 seconds

T-Encryption = 4.547210931777954 seconds

T-Decryption = 6.851265907287598 seconds

T-Verification = 0.687085151672363 seconds

T-Client = 12.085561990737915 seconds

Gain % = 289.30 %

Unsuccessful

For 10500x10500 Matrix the timing values are

T-Original = 39.160212993621826 seconds

T-Encryption = 4.603868961334229 seconds

T-Decryption = 6.850789070129395 seconds

T-Verification = 0.710709095001221 seconds

T-Client = 12.165367126464844 seconds

Gain % = 321.90 %

Unsuccessful

For 11000 x 11000 Matrix the timing values are

T-Original = 46.788379192352295 seconds

T-Encryption = 4.923585891723633 seconds

T-Decryption = 7.443346977233887 seconds

T-Verification = 0.801120042800903 seconds

T-Client = 13.168052911758423 seconds

Gain % = 355.32 %

Unsuccessful

For 11500x11500 Matrix the timing values are

T-Original = 51.789206027984619 seconds

T-Encryption = 5.993155002593994 seconds

T-Decryption = 9.197479009628296 seconds

T-Verification = 1.005900859832764 seconds

T-Client = 16.196534872055054 seconds

Gain % = 319.75 %

Unsuccessful

For 12000 x 12000 Matrix the timing values are

T-Original = 58.937861919403076 seconds

T-Encryption = 5.981251955032349 seconds

T-Decryption = 10.022524833679199 seconds

T-Verification = 1.053715944290161 seconds

T-Client = 17.057492733001709 seconds

Gain % = 345.52 %

Unsuccessful

For 12500 x 12500 Matrix the timing values are

T-Original = 67.835979938507080 seconds

T-Encryption = 6.845319032669067 seconds

T-Decryption = 10.202654838562012 seconds

T-Verification = 1.009556055068970 seconds

T-Client = 18.057529926300049 seconds

Gain % = 375.67 %

Unsuccessful

For 13000 x 13000 Matrix the timing values are

T-Original = 75.363669872283936 seconds

T-Encryption = 7.362871885299683 seconds

T-Decryption = 11.618237018585205 seconds

T-Verification = 1.106352090835571 seconds

T-Client = 20.087460994720459 seconds

Gain % = 375.18 %

Unsuccessful

For 13500x13500 Matrix the timing values are

T-Original = 85.346364974975586 seconds

T-Encryption = 7.514461040496826 seconds

T-Decryption = 12.340278863906860 seconds

T-Verification = 1.139369964599609 seconds

T-Client = 20.994109869003296 seconds

Gain % = 406.53 %

**** CLOUD RESULT ****

%Cloud Specs:

%No. of Processors : 8

%RAM : 52GB

%Storage : 50GB of SSD

%Efficiency(in %) = (T-original / T-cloud) * 100

%16 sept 2017

For 500x500 Matrix the timing values are

T-Original = 0.004136085510254 seconds

T-Cloud = 0.002379894256592 seconds

Efficiency = 173.79 %

For 1000x1000 Matrix the timing values are

T-Original = 0.022428035736084 seconds

T-Cloud = 0.014256954193115 seconds

Efficiency = 157.31 %

For 1500x1500 Matrix the timing values are

T-Original = 0.070749998092651 seconds

T-Cloud = 0.049114942550659 seconds

Efficiency = 144.04 %

For 2000x2000 Matrix the timing values are

T-Original = 0.164494037628174 seconds

T-Cloud = 0.118711948394775 seconds

Efficiency = 138.56 %

For 2500x2500 Matrix the timing values are

T-Original = 0.573264837265015 seconds

T-Cloud = 0.300907135009766 seconds

Efficiency = 190.51 %

For 3000x3000 Matrix the timing values are
T-Original = 0.985081195831299 seconds
T-Cloud = 0.507642984390259 seconds
Efficiency = 194.04 %

For 3500x3500 Matrix the timing values are
T-Original = 1.544623851776123 seconds
T-Cloud = 0.786417007446289 seconds
Efficiency = 196.41 %

For 4000x4000 Matrix the timing values are
T-Original = 2.278019905090332 seconds
T-Cloud = 1.167154788970947 seconds
Efficiency = 195.17 %

For 4500 x 4500 Matrix the timing values are
T-Original = 3.273758172988892 seconds
T-Cloud = 1.670812845230103 seconds
Efficiency = 195.93 %

For 5000x5000 Matrix the timing values are
T-Original = 4.405617952346802 seconds
T-Cloud = 2.284009933471680 seconds
Efficiency = 192.88 %

For 5500 x 5500 Matrix the timing values are
T-Original = 5.098109960556030 seconds
T-Cloud = 3.027452230453491 seconds
Efficiency = 168.39 %

For 6000x6000 Matrix the timing values are

T-Original = 7.603370904922485 seconds

T-Cloud = 3.927580118179321 seconds

Efficiency = 193.58 %

For 6500 x 6500 Matrix the timing values are

T-Original = 9.724493026733398 seconds

T-Cloud = 4.993230104446411 seconds

Efficiency = 194.75 %

For 7000x7000 Matrix the timing values are

T-Original = 12.128478050231934 seconds

T-Cloud = 6.198975086212158 seconds

Efficiency = 195.65 %

For 7500 x 7500 Matrix the timing values are

T-Original = 14.851166963577271 seconds

T-Cloud = 7.508424997329712 seconds

Efficiency = 197.79 %

For 8000x8000 Matrix the timing values are

T-Original = 17.908633947372437 seconds

T-Cloud = 9.159410953521729 seconds

Efficiency = 195.52 %

For 8500 x 8500 Matrix the timing values are

T-Original = 21.633709907531738 seconds

T-Cloud = 11.003872871398926 seconds

Efficiency = 196.60 %

For 9000x9000 Matrix the timing values are

T-Original = 25.396991014480591 seconds

T-Cloud = 13.087163925170898 seconds

Efficiency = 194.06 %

For 9500x9500 Matrix the timing values are

T-Original = 29.452494144439697 seconds

T-Cloud = 15.364851951599121 seconds

Efficiency = 191.68 %

For 10000x10000 Matrix the timing values are

T-Original = 34.963022947311401 seconds

T-Cloud = 17.601392030715942 seconds

Efficiency = 198.63 %

For 10500x10500 Matrix the timing values are

T-Original = 39.160212993621826 seconds

T-Cloud = 20.346372127532959 seconds

Efficiency = 192.46 %

For 11000 x 11000 Matrix the timing values are

T-Original = 46.788379192352295 seconds

T-Cloud = 23.607152938842773 seconds

Efficiency = 198.19 %

For 11500x11500 Matrix the timing values are

T-Original = 51.789206027984619 seconds

T-Cloud = 27.205862045288086 seconds

Efficiency = 190.36 %

For 12000x12000 Matrix the timing values are

T-Original = 58.937861919403076 seconds

T-Cloud = 30.469288825988770 seconds

Efficiency = 193.43 %

For 12500x12500 Matrix the timing values are

T-Original = 67.835979938507080 seconds

T-Cloud = 34.722978115081787 seconds

Efficiency = 195.36 %

For 13000x13000 Matrix the timing values are

T-Original = 75.363669872283936 seconds

T-Cloud = 38.868041992187500 seconds

Efficiency = 193.89 %

For 13500x13500 Matrix the timing values are

T-Original = 85.346364974975586 seconds

T-Cloud = 43.300329208374023 seconds

Efficiency = 197.10 %

For 14000x14000 Matrix the timing values are

T-Original = 46.825609207153320 seconds

T-Cloud = 48.259330034255981 seconds

Efficiency = 97.03 %

For 14500x14500 Matrix the timing values are

T-Original = 52.520282030105591 seconds

T-Cloud = 53.864274024963379 seconds

Efficiency = 97.50 %

For 15000x15000 Matrix the timing values are

T-Original = 58.389204978942871 seconds

T-Cloud = 59.610464096069336 seconds

Efficiency = 97.95 %

For 15500x15500 Matrix the timing values are

T-Original = 64.695460081100464 seconds

T-Cloud = 65.561442136764526 seconds

Efficiency = 98.68 %

For 16000x16000 Matrix the timing values are

T-Original = 71.013462066650391 seconds

T-Cloud = 71.745795965194702 seconds

Efficiency = 98.98 %

For 16500x16500 Matrix the timing values are

T-Original = 78.257222175598145 seconds

T-Cloud = 77.453938007354736 seconds

Efficiency = 101.04 %

For 17000x17000 Matrix the timing values are

T-Original = 85.333053112030029 seconds

T-Cloud = 85.345552206039429 seconds

Efficiency = 99.99 %

For 17500x17500 Matrix the timing values are

T-Original = 93.140377998352051 seconds

T-Cloud = 93.664276123046875 seconds

Efficiency = 99.44 %

For 18000x18000 Matrix the timing values are

T-Original = 101.480674982070923 seconds

T-Cloud = 101.968886137008667 seconds

Efficiency = 99.52 %

For 18500x18500 Matrix the timing values are

T-Original = 111.231128931045532 seconds

T-Cloud = 110.284446954727173 seconds

Efficiency = 100.86 %

For 19000x19000 Matrix the timing values are
T-Original = 117.673303127288818 seconds
T-Cloud = 119.151823997497559 seconds
Efficiency = 98.76 %

For 19500x19500 Matrix the timing values are
T-Original = 129.023737907409668 seconds
T-Cloud = 129.974792957305908 seconds
Efficiency = 99.27 %

For 20000x20000 Matrix the timing values are
T-Original = 138.517518997192383 seconds
T-Cloud = 139.113164901733398 seconds
Efficiency = 99.57 %

For 20500x20500 Matrix the timing values are
T-Original = 151.561541080474854 seconds
T-Cloud = 151.910385131835938 seconds
Efficiency = 99.77 %

For 21000x21000 Matrix the timing values are
T-Original = 163.534067869186401 seconds
T-Cloud = 163.794775962829590 seconds
Efficiency = 99.84 %

The client accepts the unchecked Resultant Matrix Z = E1*E2 from cloud,the client will perform the decryption of the result .

Decrypted resultant matrix Res := $R1^{-1}$ * (Z-T) * $R3^{-1}$.
where T is a temporary matrix T := R4*R4 + A*R4 + R4*B.

The client use the column matrix X to compute Y and Y' as follows
Y = Res*X
Y’ = (D1*D2)*X
such that if Y=Y' the client Accepts Z as the correct result or rejects it.

Correctness guarantee

The proposed protocol is correct if both the client and the cloud follow the protocol honestly, then the result Z returned by a honest cloud server will always be decrypted successfully and the corresponding result Z is always correct.
Observe that A and B are given by
A := R1 * D1 * R2
B := $R2^{-1}$ * D2 * R3

Note that an honest cloud server computes Z=AB = R1*D1*D2*R3 then by $R1^{-1}$ *Z* $R3^{-1}$
we have Z= D1*D2 .This implies the proposed protocol is correct.

Security

- **Input privacy**

  The protocol can protect the privacy of the client’s data. On one hand, given the encrypted E1, E2, the cloud cannot get meaningful knowledge of the client’s original input data which is referred to as input privacy.

- **Output privacy**

  On the other hand, the correct result to the original MMC problem U is also hidden from the cloud, and this is called as output privacy. he proposed

protocol can protect output privacy if given the returned result Z , the cloud cannot recover the correct result to the original MMC problem

Robust cheating resistance

The correct result from a faithful cloud server must be verified successfully by the client. The correctness of the decrypted result Z is checked if the cloud is faithful we will have Z =D1*D2 .

So if Y-Y' = $(0...0)^T$ we show that the result is from a faithful cloud server verified successfully by the client. else if the cheating cloud return a false Z , then this leads to Z != AB.

CHAPTER 5

LANGUAGES AND TECHNOLOGIES USED

1. ***Google Cloud Compute Engine***

Google Compute Engine delivers virtual machines running in Google's innovative data centers and worldwide fiber network. Compute Engine's tooling and workflow support enable scaling from single instances to global, load-balanced cloud computing.

Compute Engine's VMs boot quickly, come with persistent disk storage, and deliver consistent performance. Our virtual servers are available in many configurations including predefined sizes or the option to create Custom Machine Types optimized for your specific needs. Flexible pricing and automatic sustained use discounts make Compute Engine the leader in price/performance.

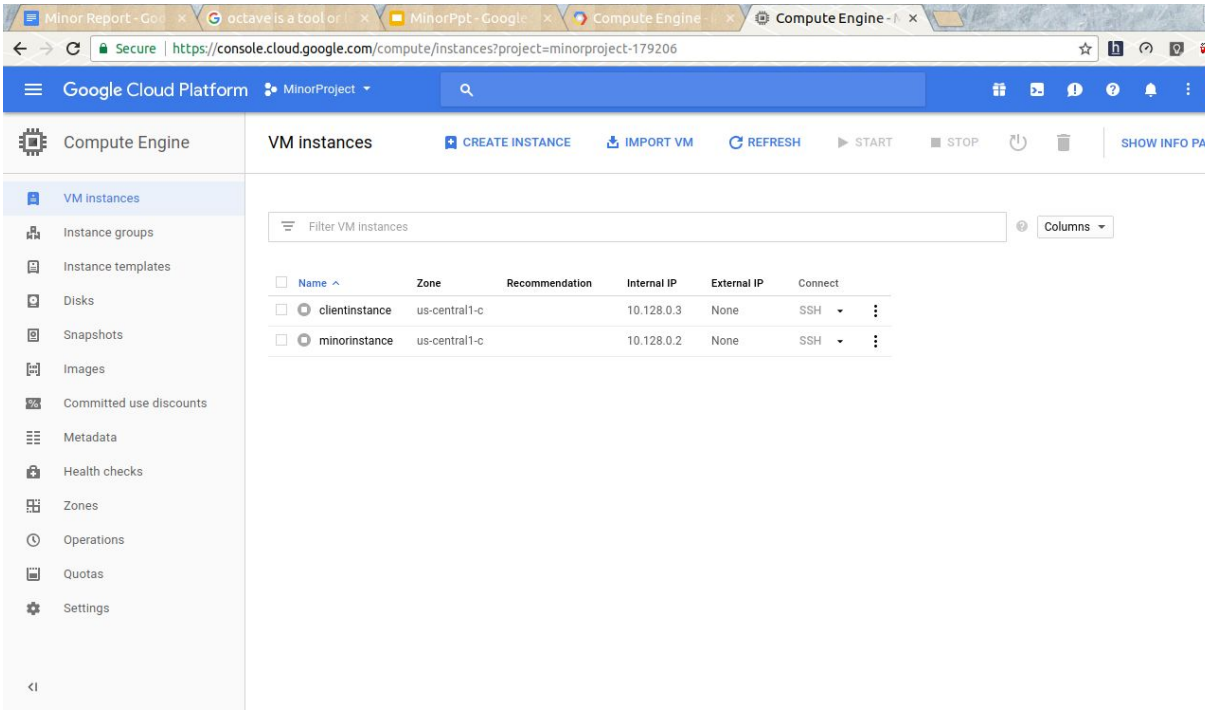


Fig 8 Google Cloud Compute Engine Console

The **configuration of the cloud** used in our project is as follows:

Operating System : Ubuntu 16.04 Lts

No. of CPU Cores : 8 Cores

Random Access Memory (RAM) : 52GB

Storage : 50GB SSD Disk

We can connect to the instances of google cloud compute engine in the following ways:

- *Connecting using Compute Engine tools*
- *Connecting to an instance from your browser*
- *Connecting to an instance through the command line*

2. *Octave*

GNU Octave is software featuring a high-level programming language, primarily intended for numerical computations. Octave helps in solving linear and nonlinear problems numerically, and for performing other numerical experiments using a language that is mostly compatible with Matlab. It may also be used as a batch-oriented language. Since it is part of the GNU Project, it is free software under the terms of the GNU General Public License.

Octave is one of the major free alternatives to Matlab, others being Scilab and FreeMat.Scilab, however, puts less emphasis on (bidirectional) syntactic compatibility with Matlab than Octave does.

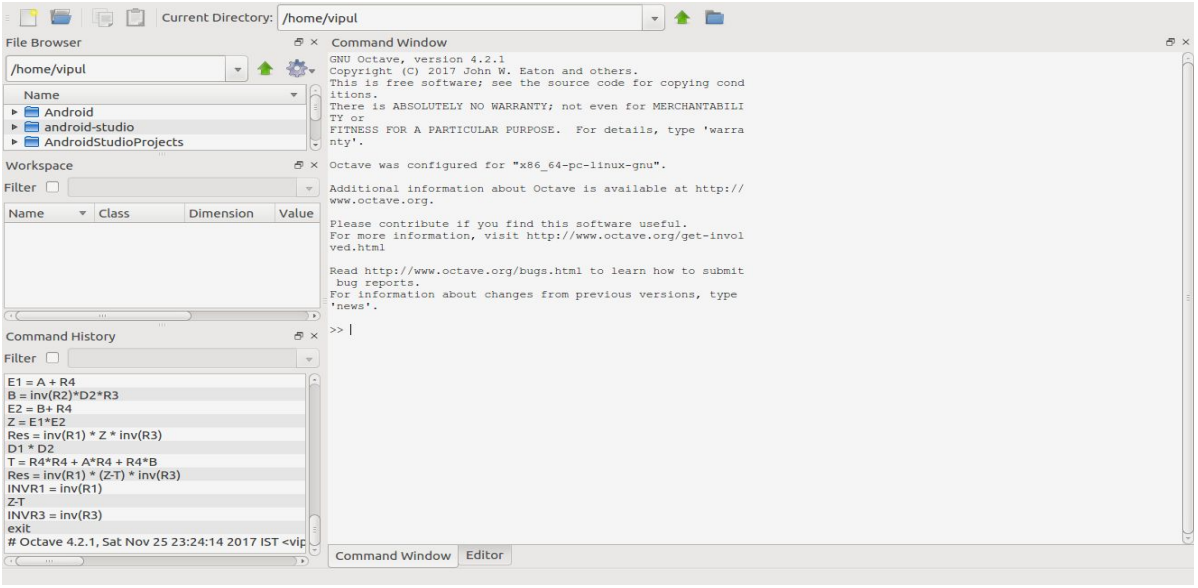


Fig 9 GNU Octave GUI Interface

CHAPTER 6

CONCLUSION & REFERENCES

CONCLUSION & FUTURE SCOPE

We have successfully implemented a mechanism for Matrix Multiplication on Encrypted Data and also carried out verification of the same.The proposed application simultaneously fulfills the goals of correctness and high-efficiency. It has wide range of applications in this ever evolving world of cloud based systems. The scheme add to security and safety of the clients data without compromising the efficiency.

Further Enhancements may include :

- A mechanism to carry out the verification before the Decryption process.
- An error correcting mechanism to correct the bits that get altered by the adversary while is being transmitted.
- Finding ways and methodologies to compute Matrix Multiplication in an even faster way.

REFERENCES

[1]Xiaofeng Chen, Xinyi Huang, Jin Li, Jianfeng Ma, Wenjing Lou, and Duncan S. Wong "New Algorithms for Secure Outsourcing of Large-Scale Systems of Linear Equations" Ieee Transactions On Information Forensics and Security, VOL. 10, NO. 1, JANUARY 2015 ,pp 69-74

[2]Xinyu Lei , Xiaofeng Liao ,Tingwen Huang , Feno Heriniaina "Achieving security, robust cheating resistance, and high-efficiency for outsourcing large matrix multiplication computation to a malicious cloud" 10May 2014, pp 205-216

[3]Shaojing Fu, Yunpeng Yu, Ming Xu, "A Secure Algorithm for Outsourcing Matrix Multiplication Computation in the Cloud", (April 02 2017),pp 27-32

[4]Raphael Yuster and Uri Zwick Department of Mathematics, University of Haifa.,"Fast Sparse Matrix Multiplication"ACM Transactions on Algorithms, Vol. 1, No. 1, July 2005, pp. 2–13.