Homomorphic Encryption for Modern Security Applications

Presented By Khalil Hariss
Supervised By
Dr. Maroun Chamoun **Dr. Abed Ellatif Samhat**

OUTLINE

1. Introduction.

2. Homomorphic Encryption in Real World Applications.

3. Homomorphic Properties.

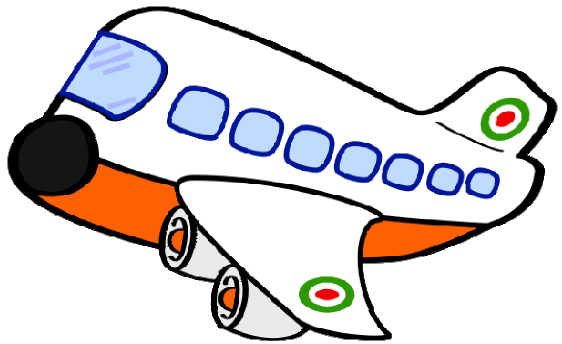
4. Homomorphic Function Example.

5. Objectives and Challenges.

Introduction

Introduction

Introduction



Introduction

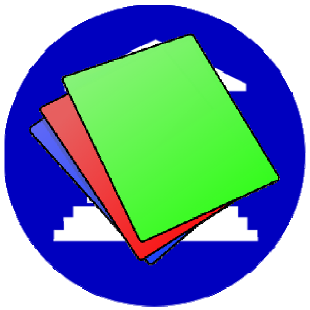
Semester I

Course	Note
Analysis	15/100
Algebra	10/100
Computer Science	5/100
Mechanic	25/100
Optic	11/100
Chemistry	6/100

Make up exam

Course	Note
Analysis	95/100
Algebra	90/100
Computer Science	85/100
Mechanic	100/100
Optic	99/100
Chemistry	89/100

Introduction

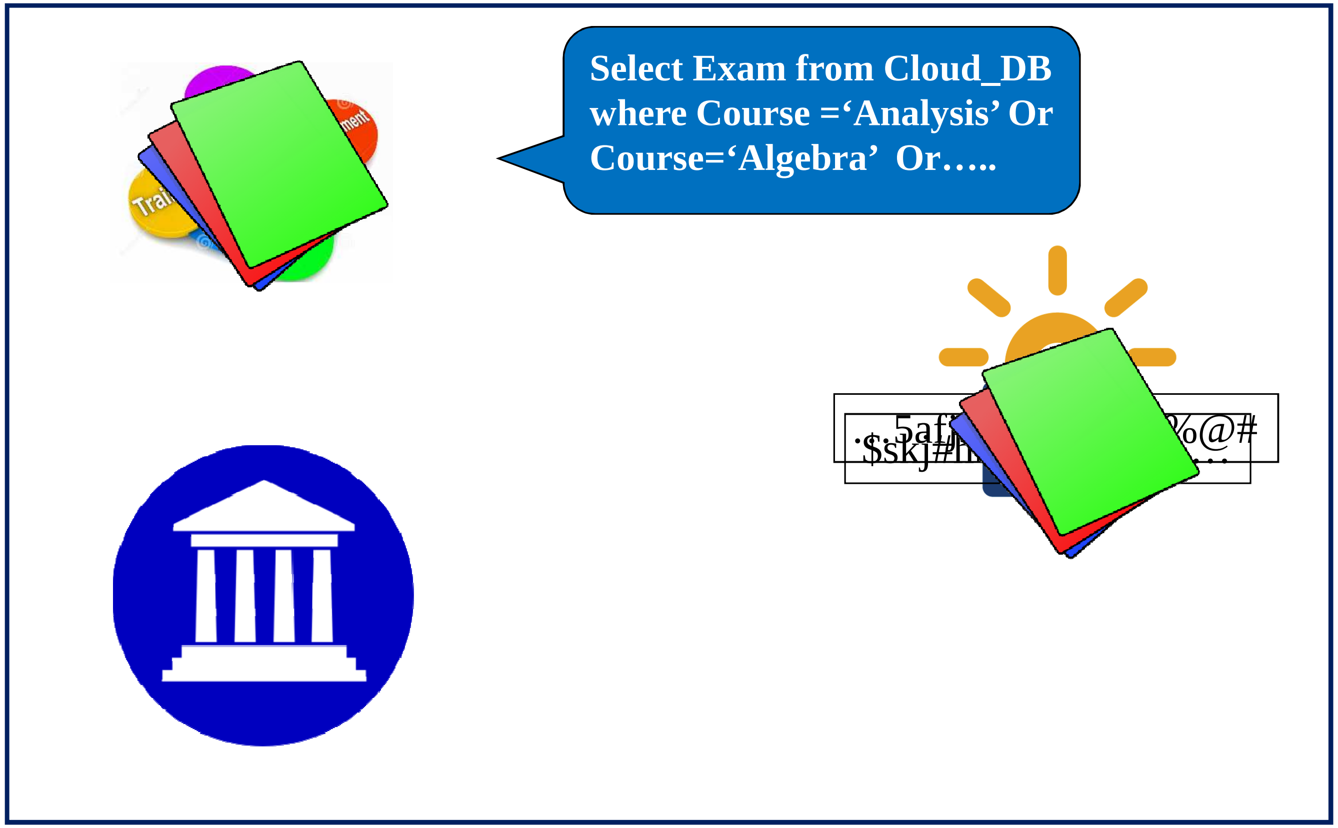


Introduction

We have …..,
We need a
solution

Homomorphic
Encryption is a
good solution

Introduction



Cloud Scenario

Cloud & Problems	What we wish to build
Cloud Computing is a data storing technique that gives opportunities for out-sourcing of storage	A new scheme that allows us to store encrypted data on the cloud
Cloud computing offers flexibility and cost saving	Keep the data encrypted on the cloud: no need to ship it back and forth to be decrypted
Main disadvantage of Cloud computing is the risk of being exposed to privacy and security issues	Send encrypted query to the cloud and allow the cloud to process it
A lot of clients retain from risking to store their sensitive data to the cloud	Cloud returns encrypted answers which will be decrypt on the client side

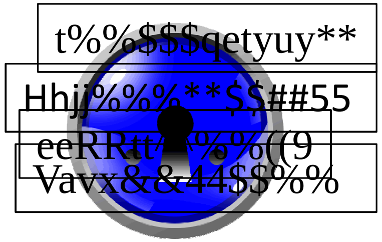
Real World Applications: e-Votes System

e-voter

candidates	A	B	C	D	E	F
Votes	0	1	0	1	0	0

candidates	A	B	C	D	E	F
Votes	0	1	0	0	0	1

e-voter



e-voter

candidates	A	B	C	D	E	F
Votes	1	0	1	0	0	0

e-voter

candidates	A	B	C	D	E	F
Votes	0	0	1	0	1	0

Real World Applications: e-Votes System

e-voter

Authorities
data center

e-voter

e-voter

candidates	A	B	C	D	E	F
Votes	58	99	45	47	15	9

Sh^^**&&0((33$$55hfolp

Add operation on
encrypted Data

e-voter

Real World Applications: DNA Matching

Jul
%%

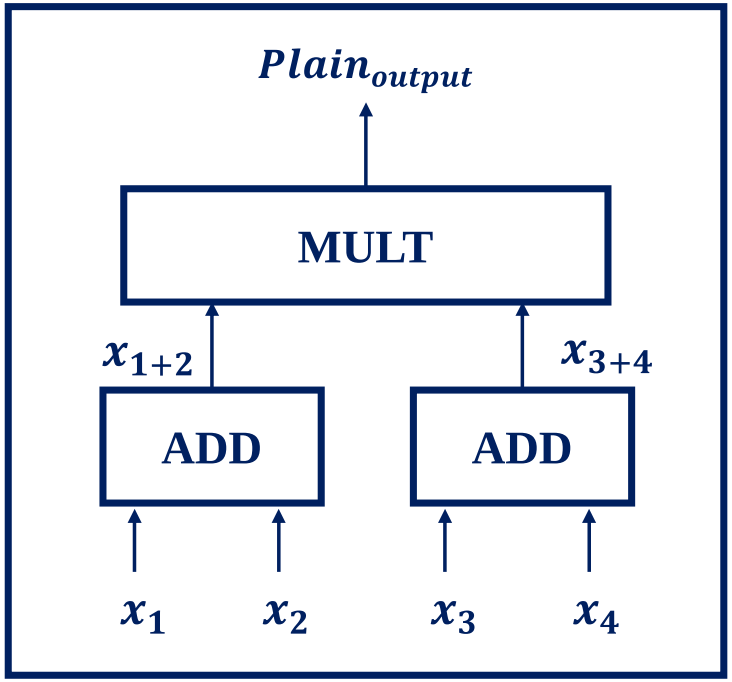
ahh%
ssgjc

Real World Applications: DNA Matching

Match

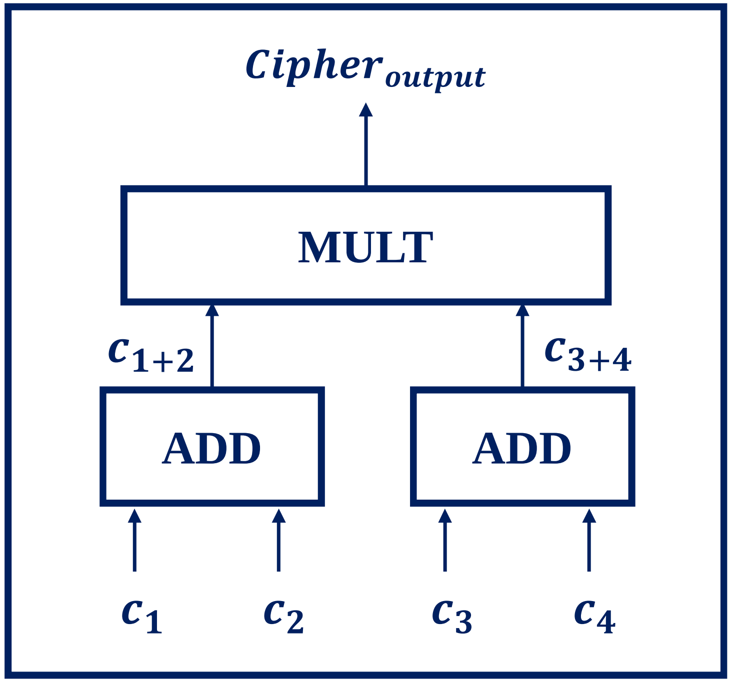
Svcgsc%%##&&
@@&&$$$wfkjw
125ddbjsvjh%%3#

$$Cipher_{output} = Enc(Plain_{output})$$



Plaintext Space

$$c_i = Enc(x_i), i \in \{1, 2, 3, 4\}$$



Ciphertext Space

Homomorphic Properties

What would happen if we have these two basic properties?

Addition

$$Enc\big((x_1+x_2), mod\ M\big) = \big(Enc(x_1) + Enc(x_2), mod\ N\big) = (c_1 + c_2, mod\ N)$$

Multiplication

$$Enc\big((x_1 * x_2), mod\ M\big) = \big(Enc(x_1) * Enc(x_2), mod\ N\big) = (c_1 * c_2, mod\ N)$$

$$Cipher_{output} = (c_{1+2}) * (c_{3+4}) = (c_1 + c_2) * (c_3 + c_4) = \big(Enc(x_1) + Enc(x_2)\big) * \big(Enc(x_3) + Enc(x_4)\big)$$
$$= \big(Enc(x_1 + x_2) * Enc(x_3 + x_4)\big) = Enc\big((x_1 + x_2) * (x_3 + x_4)\big) = Enc(Plain_{output})$$

Homomorphic Function Example

MORE Approach

- **MORE: Matrix Operation for Randomization and Encryption**

$E(m,k) = K^{-1}\begin{bmatrix} m & 0 \\ 0 & r \end{bmatrix}K$, where m the plaintext, r is a random integer in a ring Z_N, K is an invertible matrix in a ring Z_N $(2x2)$.

- The decryption process is simply given by : $KE(m,k)K^{-1} = KK^{-1}\begin{bmatrix} m & 0 \\ 0 & r \end{bmatrix}KK^{-1} = \begin{bmatrix} m & 0 \\ 0 & r \end{bmatrix}$, since the symmetric secret K is known by the two users.
- This idea can lead to a fully homomorphic symmetric encryption algorithm:

$$E(m_1) + E(m_2) = K^{-1}\begin{bmatrix} m_1 & 0 \\ 0 & r_1 \end{bmatrix}K + K^{-1}\begin{bmatrix} m_2 & 0 \\ 0 & r_2 \end{bmatrix}K = K^{-1}\begin{bmatrix} m_1 + m_2 & 0 \\ 0 & r' \end{bmatrix}K$$
$$= E(m_1 + m_2).$$

$$E(m_1).E(m_2) = K^{-1}\begin{bmatrix} m_1 & 0 \\ 0 & r \end{bmatrix}K.K^{-1}\begin{bmatrix} m_2 & 0 \\ 0 & r \end{bmatrix}K = K^{-1}\begin{bmatrix} m_1.m_2 & 0 \\ 0 & r' \end{bmatrix}K$$
$$= E(m_1.m_2).$$

Objectives and Challenges

Objectives
1- State of Art.
2- Design and Realization of New Homomorphic Schemes.
3- Implementation of the New Schemes with Cryptanalysis.
4- Implementation of Homomorphic Schemes in Real World Applications.

Challenges
1- Execution Time and Storage Overhead.
2- Level Of Security.
3- Suitable Environment for Implementation.
4- Mathematical Complexity.

Thanks for your attention