$\in \mathbb{Z}_p^*$

HOMOMORPHIC ENCRYPTION

AN INTRODUCTION

$\mathbb{Z}_p^*$

from a view point of Algebra

$\mathbb{Z}_{p-1}$

Presentation by Sandeep Chatterjee

Student, MTech Computer Science
Indian Statistical Institute

α^a

April 19th 2023 | Kolkata

click to play & watch

MOTIVATION

we all recieve dozens of e-mails daily, some genuine and few spam

but, how do e-mails services, recognize that ?

Subject: YOU WON A FREE VACATION to the Lakshadweep! ☀

Hey there!
Congratulations! You've been randomly selected to win an ALL-EXPENSES-PAID vacation to the beautiful Bahamas!
This exclusive offer includes:
Luxurious beachfront accommodation for 7 nights!
Round-trip airfare for TWO!
Daily gourmet meals and unlimited drinks!
All you have to do is claim your prize!
Click Here: [SUSPICIOUS LINK] ⚠

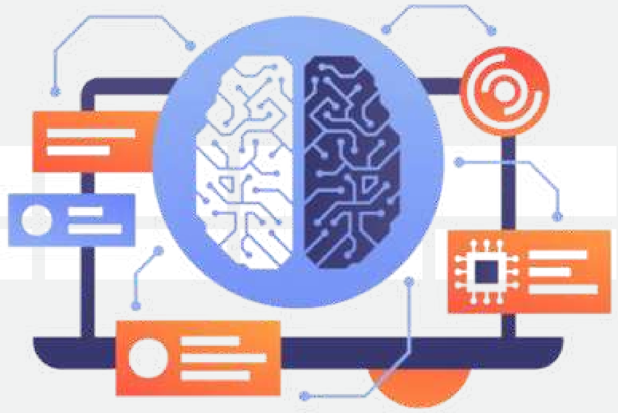
Spam!

Dear Fellow Algebra Lover,

Greetings! We're venturing into exciting new territory, we are working on homomorphic encryption-decryption system, a true game-changer.
We'll keep you posted on our progress, and who knows, maybe you'll have some brilliant insights to share along the way.
please keep this message encrypted as well, so that our communication remains secret

With utmost confidentiality,

Not Spam ✓



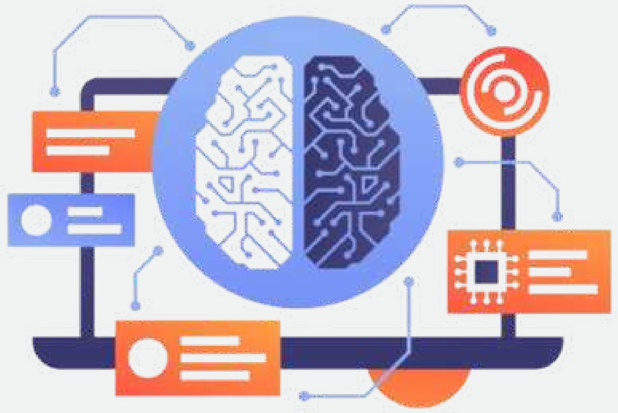
MOTIVATION

one naïve soluton is to encrypt our e-mails
in this case e-mail suppose it is encrypted by 1 shift cipher

$$Enc(a) \leftarrow a + 1 \pmod{|\Sigma|}, a \in \Sigma$$

Subject: YOU WON A FREE VACATION to the Lakshadweep! ☀

Hey there!
Congratulations! You've been randomly selected to win an ALL-EXPENSES-PAID vacation to the beautiful Bahamas!
This exclusive offer includes:
Luxurious beachfront accommodation for 7 nights!
Round-trip airfare for TWO!
Daily gourmet meals and unlimited drinks!
All you have to do is claim your prize!
Click Here: [SUSPICIOUS LINK] ⚠



Dear Fellow Algebra Lover,

Greetings! We're venturing into exciting new territory, we are working on homomorphic encryption-decryption system, a true game-changer.
We'll keep you posted on our progress, and who knows, maybe you'll have some brilliant insights to share along the way.
please keep this message encrypted as well, so that our communication remains secret

With utmost confidentiality,

Efbs Gpmmp Bmhbcsb Mpwfs,

Hsfuifout! Xfs'fs wfovsfjoh joup fjydjuz ofydjodz, xfs bsf xpsljoht pshmf sjtfozpgz-pfdqfmjdz tpztubn, b usvf hbnf-dibnfsbmf dvhzt. Xfmm lffu

zpv qpttfe po pvs qsjdfqt, boe xip ipopt, nbczf zpv'mm ibwf tpnf csfmjjot joup tibsf bmoh obohjoh uif xbz. qmfbtf lffu uift nfttbhf fodszqufe bt xfmn, tp uibu pvs dpnnvojujft sfbmmjot tfdvsfu

Xjui vtnptu dpogjefoujcmz

confidential!!!

April 19th 2023 | Kolkata

Sandeep Chatterjee

MOTIVATION

we will encrypt all the messages using a secure protocol such as AES-256

U2FsdGVkX1+TkBf2w
KkITm9JEXG/MoChkht
r/MRy7s19fRczQaXrCG
qbAMxEv2Ul
bcGMI6LmI5NAXd7S6I
Dmf67+8rD8GvhUTW2
9T2TbIQOBFtN31Es4s
NLEvW4LZT0g
S8jU9b4hB5kb1y0HH6
LrWh9ghhI=

U2FsdGVkX19h9Cfv0H
Wu6yCdwIu4V04ImWXv
DLIArPlcP7lYZzqGKZym
j7d3uiof
GVFLCJ8wRDxqzZM5M6
Aq2yigME1fBy4q12XZ5I
N+VXVKqC+00bL+Kw
wQ8Mv3Rf+
44g9IBBZtd66GFX+OxD
6b8lcEL4=

but the question is then, can our classifier still recognize the e-mails ?

Sandeep Chatterjee

MOTIVATION

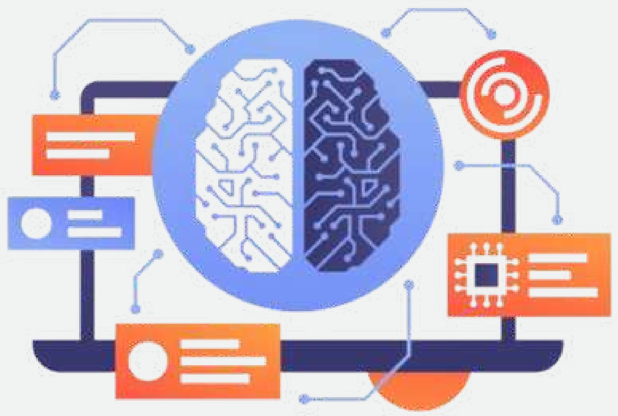
it’s expected to happen

$$E : \mathcal{W} \mapsto \mathcal{W}$$

$$\mathcal{C} : \mathcal{W}^* \rightarrow \{0, 1\}$$

$$c_1, c_2, \ldots, c_n \leftarrow E(w_1, w_2, \ldots, w_n)$$

U2FsdGVkX1+TkBf2w
KkITm9JEXG/MoChkht
r/MRy7s19fRczQaXrCG
qbAMxEv2Ul
bcGMI6LmI5NAXd7S6I
Dmf67+8rD8GvhUTW2
9T2TbIQOBFtN31Es4s
NLEvW4LZT0g
S8jU9b4hB5kb1y0HH6
LrWh9ghhI=



U2FsdGVkX19h9Cfv0H
Wu6yCdwIu4V04ImWXv
DLIArPlcP7lYZzqGKZym
j7d3uiof
GVFLCJ8wRDxqzZM5M6
Aq2yigME1fBy4q12XZ5I
N+VXVKqC+00bL+Kw
wQ8Mv3Rf+
44g9IBBZtd66GFX+OxD
6b8lcEL4=

$$\mathcal{C}(c_i) \neq \mathcal{C}(w_i), \text{ where } c_i = E(w_i)$$

$$\mathcal{C}(< c_1, c_2, \ldots, c_n >) \neq \mathcal{C}(< w_1, w_2, \ldots, w_n >)$$

Sandeep Chatterjee

MOTIVATION

So, we will look into its solution

We see that, after the encryption operation is performed, the data is opaque for any further operations

MOTIVATION

Conventional encryption techniques, while effective at securing data, pose a significant obstacle to computational algorithms. When all inputs are encrypted using traditional methods, other operations are rendered ineffective as they lose the underlying structure.

This dilemma underscores the critical need for innovative solutions that reconcile data privacy with computational functionality. Enter homomorphic encryption, a groundbreaking cryptographic technique that enables computations to be performed directly on encrypted data without the need for decryption

ABSTRACT

In this presentation, a gentle introduction to homomorphic encryption is presented, a cutting-edge cryptographic technique with profound implications for data privacy and computational security. Through an exploration from an algebraic perspective, we will delve into the fundamental principles underlying homomorphic encryption, elucidating its mechanisms and capabilities. By examining its algebraic foundations, we aim to demystify this powerful cryptographic tool and shed light on its practical applications. Join us as we embark on a journey to uncover the transformative potential of homomorphic encryption in safeguarding privacy while enabling meaningful computations on encrypted data.

PRELIMINARIES

Algebraic Homomorphisms

Definition (Group Homomorphism)

Let $(G, *)$ and $(H, \diamond)$ be groups. The map $\phi : G \to H$ is a homomorphism iff $\phi(x \diamond y) = \phi(x) \diamond \phi(y)\ \forall\ x, y \in G$

Definition (Ring Homomorphism)

Let R and S be rings with addition and multiplication. The map $\phi : R \to S$ is a homomorphism iff

1 ϕ is a group homomorphism on the additive groups $(R, +)$ and $(S, +)$

$$\phi(a+b) = \phi(a) + \phi(b)\ ,\quad \forall\ a, b \in R$$

2 ϕ preserves multiplication

$$\phi(xy) = \phi(x).\phi(y)\ ,\qquad \forall\ x, y \in R$$

Sandeep Chatterjee

PRELIMINARIES

Algebraic Homomorphisms

What does this give us?

Consider the map $\phi : \mathbb{Z} \to \mathbb{Z}_n$
sending **k** to **k**. We've seen that this is a homomorphism of additive groups, and can easily check that multiplication is preserved. Indeed,

$$\phi(a) = \phi(1 + 1 + \cdots + 1) = \phi(1) + \phi(1) + \cdots + \phi(1) = a\phi(1) = a$$

PRELIMINARIES

Algebraic Homomorphisms

What does this give us?

The evaluation map e_k is a function from R[x] to R.
For any polynomial f ∈ R[x] and k ∈ R, we set $e_k(f)=f(k)$
This is a ring homomorphism!
Let $f(x) = a_n x^n + \cdots a_0 x^0,$ and $g(x) = b_n x^n + \cdots b_0 x^0$, where the $a_i, b_i \in R$.

Since we know that e_k is an additive homomorphism, we only need to check that it is multiplicative on monomials. But that's easy:

$$\begin{aligned} e_k((ax^n)(bx^m)) &= e_k(abx^{n+m}) \\ &= abk^{n+m} = e_k(ax^n)e_k(bx^m). \end{aligned}$$

PRELIMINARIES

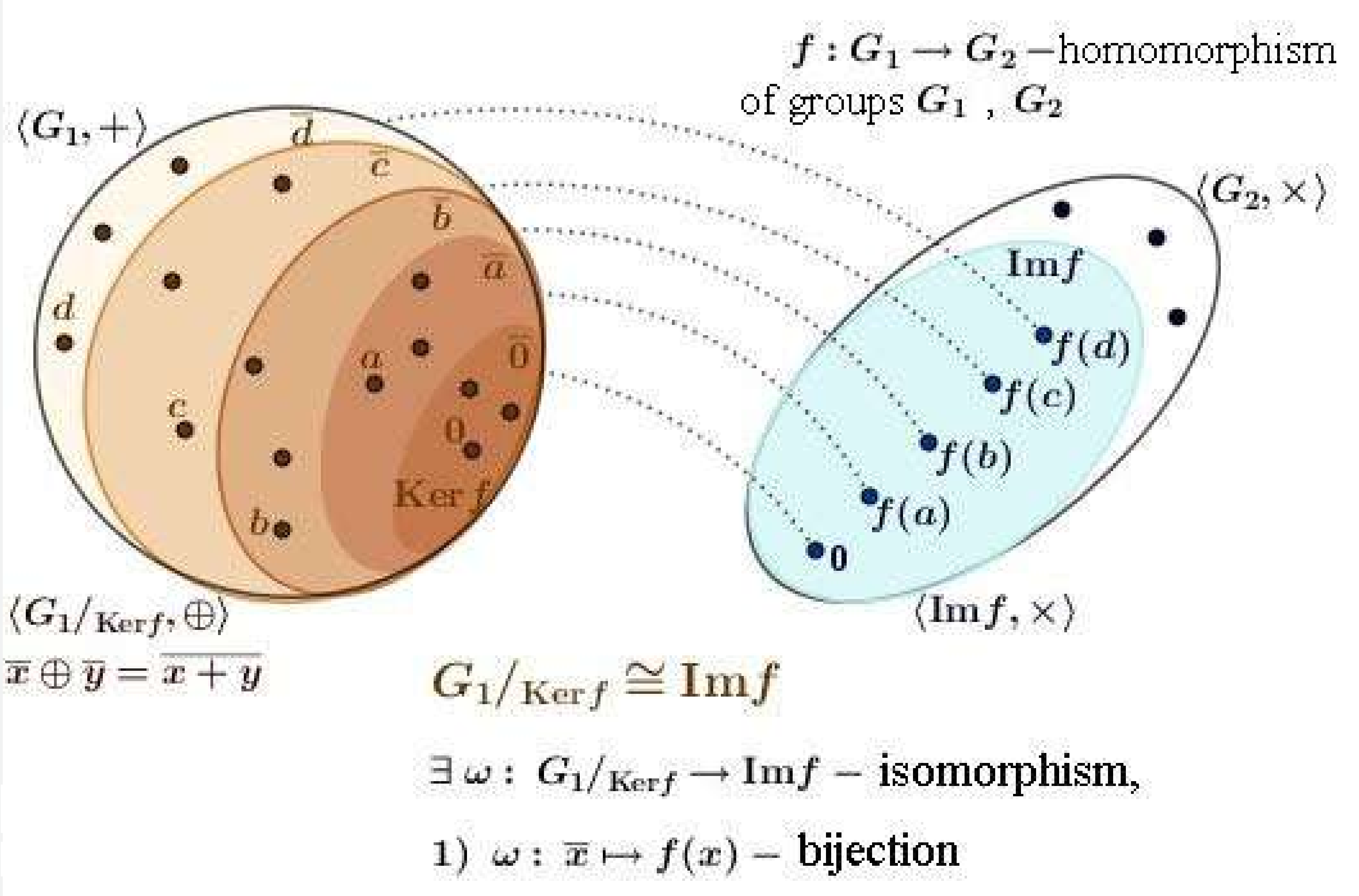


Fig : Visualizing Group Visualization

credits : Scientific Visualization, 2019, volume 11, number 5, pages 83 - 100

HOMOMORPHISMS

But how does homormorphism relate to Crytography ?

A homomorphic encryption function allows for the manipulation of encrypted data with out the seemingly inherent loss of the encryption.

Applications

• E-Cash • E-Voting • Private information retrieval • Cloud computing

A fully homomorphic encryption function (two operations) has been an open problem in cryptography for 30+ years. The first ever system was proposed by Craig Gentry in 2009. However, encryption systems that respect one operation have been utilized for decades.

EXAMPLE

RSA Cryptosystem

Let n = pq where p and q are primes. Pick a and b such that ab ≡ 1 (mod φ(n)).
n and b are public while p, q and a are private

The Homomorphism: Suppose x1 and x2 are plaintexts. Then,

$$e_K(x) = x^b \pmod{n}$$

$$d_K(y) = y^a \pmod{n}$$

The Homomorphism: Suppose x and y are plaintexts. Then,

$$e_K(x) \cdot e_K(y) = x^b \cdot y^b \pmod{n} \equiv (x \cdot y)^b \pmod{n} = e_K(x \cdot y)$$

$$e_K(x) \cdot e_K(y) \equiv e_K(x \cdot y)$$

RSA Cryptosystem

EXAMPLE

Alice	on public network	Bob
		pick large primes p and q
		compute **RSA modulus** $n = pq$
		pick **RSA exponent** $e \in \mathbb{N}$
		with $\gcd(e, \phi(n))$ = 1
download k_e	$\leftarrowtail$ public key k_e $\leftarrowtail$	publish $k_e = (n, e)$
		compute $d = e^{-1} \pmod{*} \phi(n)$
message $m \in \mathcal{M}$		
compute $c = e_{(n,e)}(m)$		
$= m^e \pmod{*} n$		
transmit c	$\rightarrowtail$ ciphertext c $\rightarrowtail$	receive c
		compute $m = d_{(n,d)}(c)$
		$= c^d \pmod{*} n$

Credits : Yet Another Introductory Number Theory Textbook - Cryptology Emphasis (Poritz)

HISTORY

On Data Banks and Privacy Homomorphisms

1978

ON DATA BANKS AND PRIVACY HOMOMORPHISMS

Ronald L. Rivest
Len Adleman
Michael L. Dertouzos

Massachusetts Institute of Technology
Cambridge, Massachusetts

I. INTRODUCTION

Encryption is a well-known technique for preserving the privacy of sensitive information. One of the basic, apparently inherent, limitations of this technique is that an information system working with encrypted data can at most store or retrieve the data for the user; any more complicated operations seem to require that the data be decrypted before being operated on. This limitation follows from the choice of encryption functions used, however, and although there are some truly inherent limitations on what can be accomplished, we shall see that it appears likely that there exist encryption functions which permit encrypted data to be operated on without preliminary decryption of the operands, for many sets of interesting operations. These special encryption functions we call "privacy homomorphisms"; they form an interesting subset of arbitrary encryption schemes (called "privacy transformations").

HISTORY

On Data Banks and Privacy Homomorphisms

- **Rivest, Adleman and Dertouzos, 1978**

- **Introduced idea of “Privacy Homomorphisms”**
- **“...it appears likely that there exist encryption functions which permit encrypted data to be operated on without preliminary decryption.”**

- **Encrypted data of loan company**
  - **What is the size of the average loan?**
  - **How many loans over $5,000?**
- **Introduced four possible encryption functions (RSA was one of them)**

HISTORY

Blind Signatures for Untraceble Payments

1983

BLIND SIGNATURES FOR UNTRACEABLE PAYMENTS

David Chaum

Department of Computer Science
University of California
Santa Barbara, CA

INTRODUCTION

Automation of the way we pay for goods and services is already underway, as can be seen by the variety and growth of electronic banking services available to consumers. The ultimate structure of the new electronic payments system may have a substantial impact on personal privacy as well as on the nature and extent of criminal use of payments. Ideally a new payments system should address both of these seemingly conflicting sets of concerns.

On the one hand, knowledge by a third party of the payee, amount, and time of payment for every transaction made by an individual can reveal a great deal about the individual's whereabouts, associations and lifestyle. For example, consider payments for such things as transportation, hotels, restaurants, movies, theater, lectures, food, pharmaceuticals, alcohol, books, periodicals, dues, religious and political contributions.

On the other hand, an anonymous payments systems like bank notes and coins suffers from lack of controls and security. For example, consider problems such as lack of proof of payment, theft of payments media, and black payments for bribes, tax evasion, and black markets.

A fundamentally new kind of cryptography is proposed here, which allows an automated payments system with the following properties:

(1) Inability of third parties to determine payee, time or amount of payments made by an individual.

HISTORY

Blind Signatures for Untraceble Payments

- **David Chaum, 1982**
- **Calls for payment system with:**
  - **Anonymity of payment**
  - **Proof of payment**
- **Analogy to secure voting**
  - **Place vote in a carbon envelope**
  - **The signer can then sign the envelope, consequently signing the vote with out ever knowing what the vote is**
- **Although no mention of a private homomorphism, the paper helps introduce the need for secure voting as well as the relationship between e-cash and e-voting**

ELGAMAL CRYPTOSYSTEM

Definition

Let p be a prime and pick α $\in \mathbb{Z}_p^*$ **such that α is a generator of** $\mathbb{Z}_p^*$ **.**
Pick a and β such that β $\equiv \alpha^a$**(mod p). p, α and β are public; a is private.**

Let r $\in \mathbb{Z}_{p-1}$ **be a secret random number. Then,**

$$e_K(x, r) = (\alpha^r \mod p, x \cdot \beta^r \mod p)$$

Sandeep Chatterjee

ELGAMAL CRYPTOSYSTEM

$$e_K(x, r) = (\alpha^r \mod p, x \cdot \beta^r \mod p)$$

Homomorphism?

Let x and y be plaintexts. Then,

$$e_K(x, r_1) \cdot e_K(y, r_2) = (\alpha^{r_1} \mod p, x \cdot \beta^{r_1} \mod p) \cdot (\alpha^{r_2} \mod p, y \cdot \beta^{r_2} \mod p)$$

$$= (\alpha^{r_1}\alpha^{r_2} \mod p, x\beta^{r_1}y\beta^{r_2} \mod p)$$

$$= (\alpha^{r_1+r_2} \mod p, (xy)\beta^{r_1+r_2} \mod p)$$

$$= e_K(x \cdot y, r_1 + r_2)$$

$$e_K(x, r_1) \cdot e_K(y, r_2) \equiv e_K(x \cdot y, r_1 + r_2)$$

ELGAMAL CRYPTOSYSTEM

$$e_K(x, r) = (\alpha^r \mod p, x \cdot \beta^r \mod p)$$

Homomorphism?

Let x and y be plaintexts. Then,

$$e_K(x, r_1) \cdot e_K(y, r_2) \equiv e_K(x \cdot y, r_1 + r_2)$$

But, there is a problem here.

The Problem: This homomorphism is multiplicative

- **E-cash and e-voting would benefit from an additive homomorphism**

ELGAMAL CRYPTOSYSTEM

$$e_K(x, r) = (\alpha^r \mod p, \alpha^x \beta^r \mod p)$$

Solution?

Modify ElGamal

- **Put the plaintext in the exponent**

But, there is a problem here.

The Problem: This homomorphism is multiplicative

- **E-cash and e-voting would benefit from an additive homomorphism**

ELGAMAL CRYPTOSYSTEM

$$e_K(x, r) = (\alpha^r \mod p, \alpha^x \beta^r \mod p)$$

Homomorphism?

Let x and y be plaintexts. Then,

$$e_K(x, r_1) \cdot e_K(y, r_2) \equiv e_K(x + y, r_1 + r_2)$$

The problem with this modification is that $d_K = \alpha^x$, introducing the discrete logarithm problem into the decryption. For large enough texts, this becomes impractical. We would like another cryptosystem which takes advantage of this additive property of exponentiation, but does so with out extra decryption time.

Solution: the Paillier Cryptosystem

Sandeep Chatterjee

PAILLIER CRYPTOSYSTEM

- Introduced by Pascal Paillier in Public-Key Cryptosystems Based on Composite Degree Residuosity Classes, 1999
- Probabilistic, asymmetric algorithm
- Decisional composite residuosity assumption

Given composite n and integer z, it is hard to determine if y exists such that

$$\exists_? y : z \equiv y^n \pmod{n^2}$$

- Homomorphic and self-blinding
- Extended by Damgard and Jurik in 2001

$$z \equiv y^n \mod n^{s+1}$$

PAILLIER CRYPTOSYSTEM

Definition

Pick two large primes p and q and let n = pq. Let λ denote the Carmichael function, that is, λ(n) = lcm(p − 1, q − 1). Pick random $g \in \mathbb{Z}_{n^2}$ such that $L(g^\lambda \pmod{n^2})$ is invertible modulo n (where $L(u) = \frac{u-1}{n}$. n and g are public; p and q (or λ) are private. For plaintext x and resulting ciphertext y, select a random $r \in \mathbb{Z}_n^*$. Then

$$e_K(x, r) = g^m r^n \mod n^2$$

$$d_K(y) = \frac{L(y^\lambda \mod n^2)}{L(g^\lambda \mod n^2} \mod n$$

$$e_K(x, r_1) \cdot e_K(y, r_2) \equiv e_K(x + y, r_1 + r_2)$$

Sandeep Chatterjee

E-VOTING

Pallier Example

Suppose Alice, Bob and Oscar are running in an election. Only 6 people voted in the election, and the results are tabulated below

Vote	Soumik	Rajdeep	Sandeep
1			✓
2		✓	
3		✓	
4			✓
5	✓		
6			✓

→

00 00 01 = 1
00 01 00 = 4
00 01 00 = 4
00 00 01 = 1
01 00 00 = 16
00 00 01 = 1

E-VOTING

Pallier Example

Let p = 5 and q = 7. Then n = 35, n^2 = 1225 and λ = 12. g is chosen to be 141. For the first vote x1 = 1, r is randomly chosen as 4. Then,

$$e_K(x_1, r_1) = e_K(1, 4) = 141^1 \cdot 4^{35} = 141 \cdot 324 = 359 \bmod 1225$$

All votes, r values and resulting encryptions are shown below

x	r	$e_k(x, r)$
1	4	359
4	17	173
4	26	486
1	12	1088
16	11	541
1	32	163

Sandeep Chatterjee

E-VOTING

Pallier Example

Then, In order to sum the votes, we multiply the encrypted data modulo n^2

$$359 \cdot 173 \cdot 486 \cdot 1088 \cdot 541 \cdot 163 \bmod 1225 = 983$$

We then decrypt:

$$L(y^\lambda \bmod n^2) = L(983^{12} \bmod 1225) = \frac{36-1}{35} = 1$$

$$L(g^\lambda \bmod n^2) = L(141^{12} \bmod 1225) = \frac{456-1}{35} = 13$$

$$\begin{aligned} d_K(y) &= \left(L(y^\lambda \bmod n^2)\right)\left(L(g^\lambda \bmod n^2)\right)^{-1} \bmod n \\ &= 1 \cdot 13^{-1} \bmod 35 \\ &= 27 \end{aligned}$$

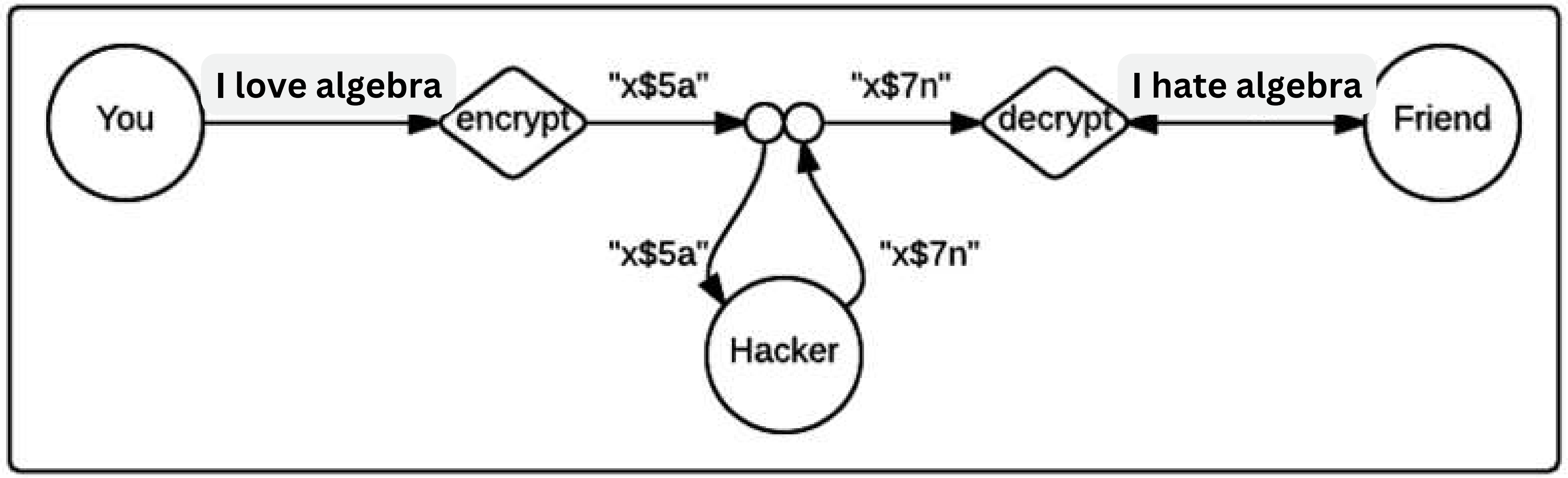
We convert 27 to (01 02 03) for the final results.

Vote	Soumik	Rajdeep	Sandeep
Total	1	2	**3**

SECURITY-NOTION

Are homomorphic encryptions secure ?
or are we losing anything in trade of these features?

Homomorphic encryption is malleable by design. A malleable crypto-system is one in which anyone can intercept a cipher text, transform it into another cipher text, and then decrypt that into a plain text that makes sense. Malleability is generally considered undesirable in a crypto-system.



Sandeep Chatterjee

SECURITY-NOTION

"Security" depends on the attack-model and goal for rigourous cryptanalysis.
In basic models of simple adversaries such as Cipher-only attack · Known-plaintext attack, it is as secure as normal encryption

Homomorphic systems should be malleable. Maybe You want to build a system that simply adds exclamation marks to whatever you send your friend. But, you don't want the system to know what you're sending your friend, that's a secret.
Malleable systems allow for multiple parties, especially in cloud-based environments, to operate on data without ever exposing it.

MORE APPLICATIONS

- Another Application: Private Information Retrieval
- Idea first introduced by Chor, Goldreich, Kushilevitz and Sudan in 1997
- The problem:
  - How can the user access an item from a database with out the database knowing which item it is? (Private Information Retrieval)
  - How can the user do this with out knowing about any other item of the database? (Symmetric Private Information Retrieval)
- The additive homomorphic properties of Paillier allow for the indexing and filtering of an encrypted database

HOMOMORPHISM

Homomorphic refers to homomorphism in algebra: the encryption and decryption functions can be thought of as homomorphisms between plaintext and ciphertext spaces.

Encryption with an additional evaluation capability for computing over encrypted data without access to the secret key. The result of such a computation remains encrypted.

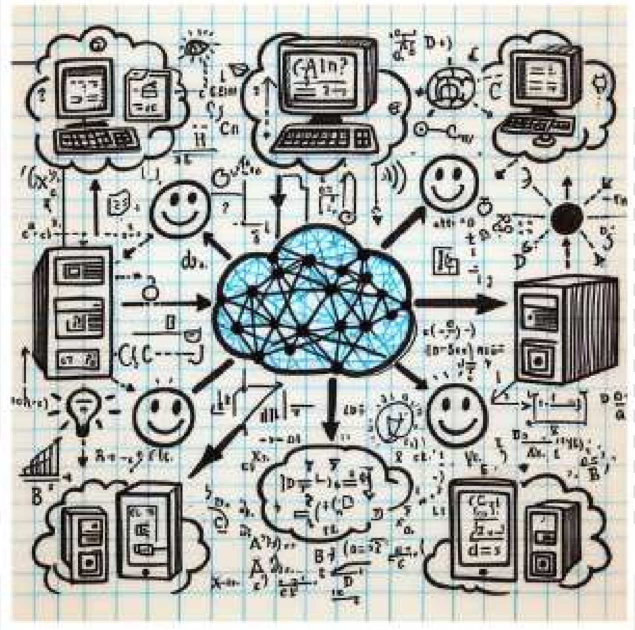
It includes multiple types of encryption schemes that can perform different classes of computations with different capabilities

HOMOMORPHIC ENCRYPTION

Scheme Type	Capability
Partially homomorphic encryption	**encompasses schemes that support the evaluation of circuits consisting of only one type of gate, e.g., addition or multiplication.**
Somewhat homomorphic encryption	**schemes can evaluate multiple types of gates, but only for a subset of circuits**
Fully homomorphic encryption (FHE)	**allows the evaluation of arbitrary circuits composed of multiple types of gates of unbounded depth and is the strongest notion of homomorphic encryption.**

HOMOMORPHIC ENCRYPTION

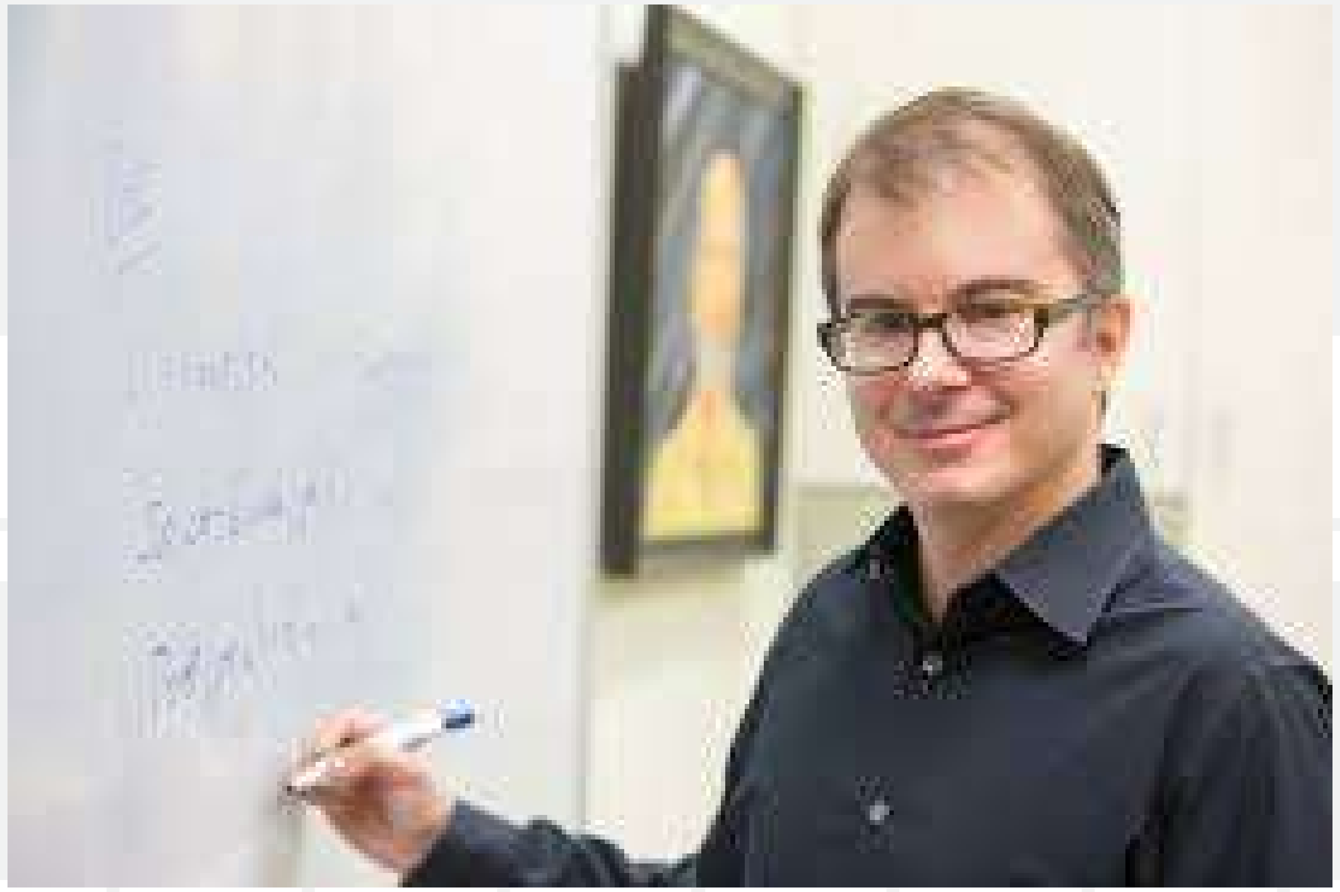
- Up until now, the homomorphic systems described have been partially homomorphic (PHE)
- They preserve the structures of multiplication or division, but cannot do both
    - If a fully homomorphic encryption was implemented, then any arbitrary computation could be performed on a ciphertext, preserving the encryption as if the computation was performed on the plaintext
    - The additive and multiplicative preservation of a Ring Homomorphism modulo 2 directly correspond to the XOR and AND operations of a circuit
- Applications:
    - Private queries on search engines - The search engine would be able to return encrypted data with out every decrypting the query
    - Cloud Computing - Storing encrypted data on the cloud is seemingly useless; no manipulation of the data can be obtained with out allowing the cloud access and/or decrypting the data off the cloud



FULLY HOMOMORPHIC ENCRYPTION

CRAIG GENTRY

Fully homomorphic systems are homomorphic systems in which any kind of mathematical operation can be performed on the cipher text. Fully homomorphic systems do exist today, and their optimizations since 2009 have made them practical for some applications. Craig Gentry was the first to suggest that they could be theoretically possible. He was able to create a system that was homomorphic in two ways, and those two ways allowed full homomorphism.



FULLY HOMOMORPHIC ENCRYPTION

CRAIG GENTRY

Gentry uses the analogy of a jewelry shop owner in his thesis to describe why fully homomorphic systems should be, and are, possible. Imagine that Alice is a jewelery store owner. She has employees that assemble products from raw materials like diamonds and gold. But, she's worried about the possibility of theft. So, she designs boxes that have gloves attached to them. Employees can stick their hands into the box to assemble the products, but they cannot take anything out of the box because only Alice has the key. So, Alice's employees can do operations on the secure data (the jewelry) without ever having the possibility of taking that secure data out.

Gentry's system incorporates an amount of noise into the cryptographic process. Each successive encryption introduces more noise into the system, which is why Gentry's initial design is impractical (though it was later improved upon). It is impractical to use noise because eventually the system needs to be restarted because the added noise makes the entire system much slower. This system relies on ideal Lattice Based Cryptography to simplify much of the system's design.

FULLY HOMOMORPHIC ENCRYPTION

CRAIG GENTRY

- Centers around a function which introduces a certain level of noise into the encryption
  - Each operation on the ciphertext results in compounding noise
  - Resolved with the bootstrapability of the encryption
    - Each re-encryption cuts down the noise
    - Analogy to Alice’s jewelry shop
- Involves operations on Ideal Lattices
  - Allows for less complex circuit implementation
  - Correspond to the structure of Rings

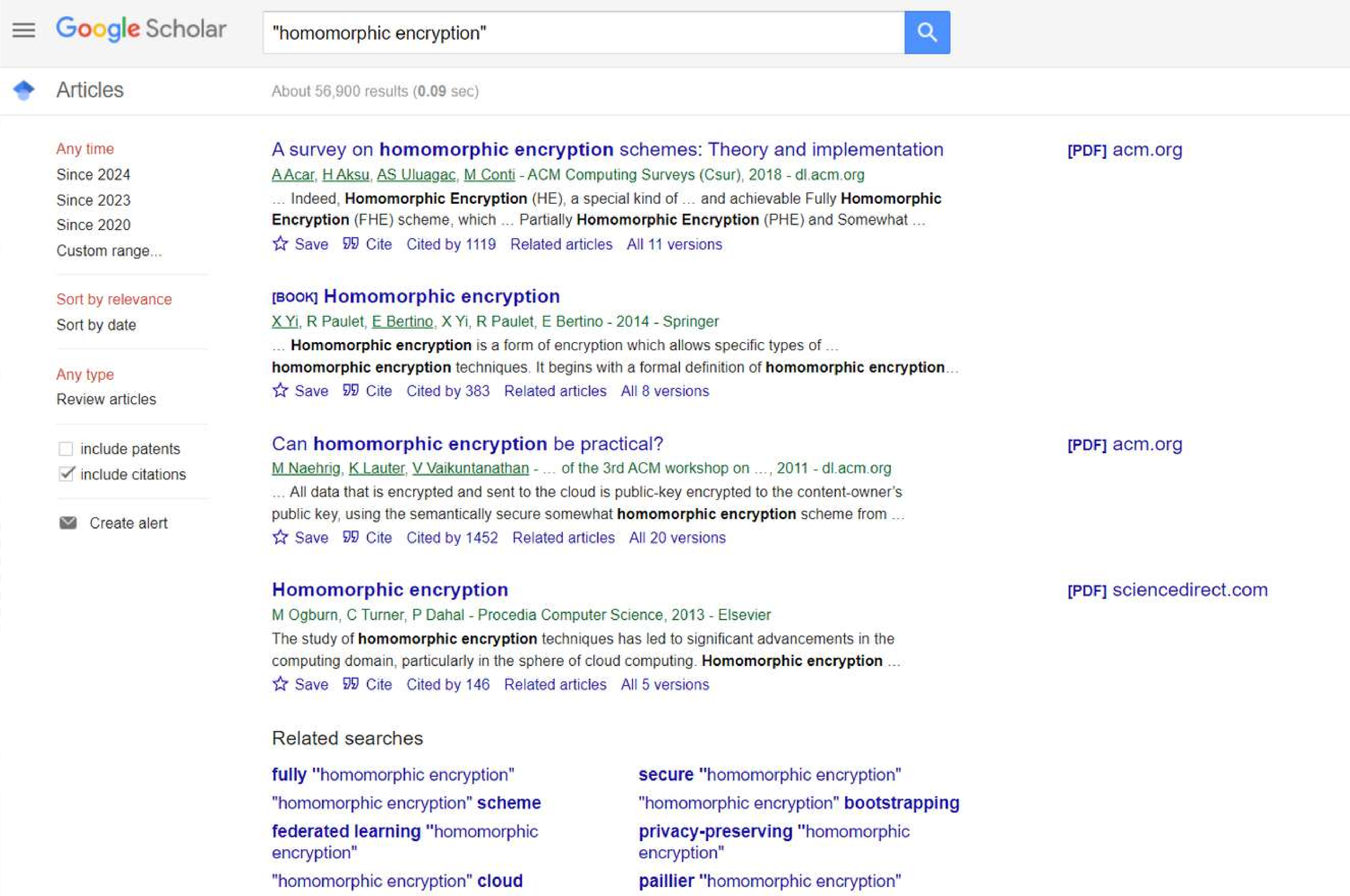
FULLY HOMOMORPHIC ENCRYPTION

CRAIG GENTRY

- However, the combination of the noise production followed by the noise reduction makes the scheme completely impractical
  - Complexity grows as more and more operations are performed (inherent limitation of the algorithm)
  - Gentry has stated that in order to perform one search on Google using this encryption, the amount of computations needed would increase by a trillion
- More schemes have been introduced to try and decrease this complexity, but all rely on the same
- Despite this impracticality, Gentry’s discovery is an amazing breakthrough in cryptography and proves that (at least theoretical) fully homomorphic encryption schemes exist

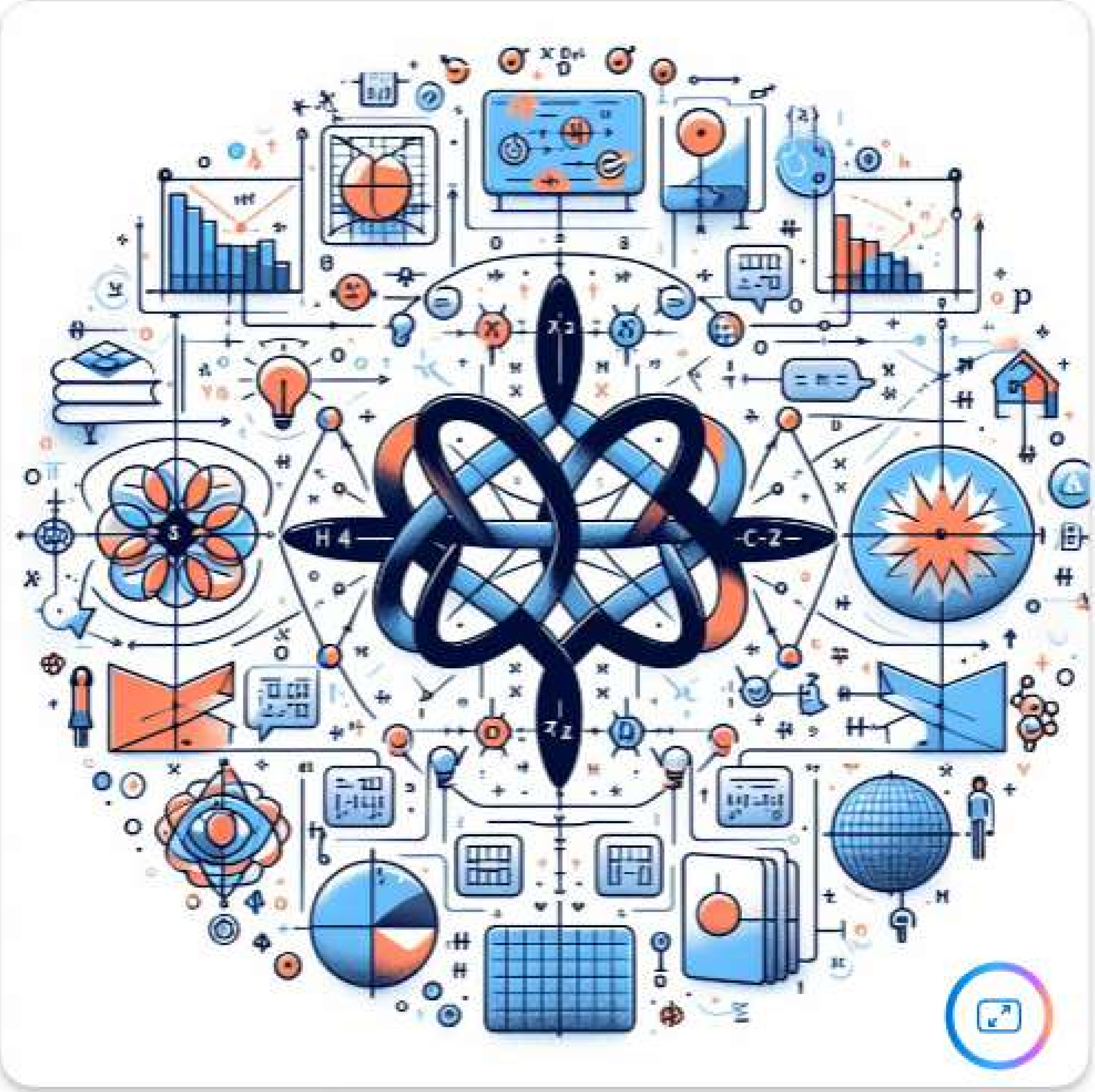
STILL AN ACTIVE AREA OF RESEARCH

FOR MATHEMATICIANS, CRYPTOLOGISTS, DATA SCIENTISTS



REFERENCES

- R. L. Rivest, L. Adleman, and M. L. Dertouzos. “On data banks and privacy homomorphisms.” Foundations of Secure Computation, 1978.
- Craig Gentry. “Computing Arbitrary Functions of Encrypted Data.” Association for Computing Machinery, 2010.
- Pascal Paillier. “Public-Key Cryptosystems Based on Composite Degree Residuosity Classes.” EUROCRYPT 1999.
- “Fully HE-based (FHE) processing remains 1,000 to 1,000,000 times slower than equivalent plaintext operations.” – Ulf Mattsson, Protegrity Corp., June 2021
- Boneh, Dan, and Victor Shoup. "A graduate course in applied cryptography." *Draft 0.6* (Jan 2023); Stanford



THANK YOU SO MUCH

FOR LISTENING

1394 , {n, e}: {3127, 3}

(RSA encrytion for “thank you so much”, with the public key e in mod φ(n))

April 19th 2023 | Kolkata

You can reach out to me at sandeepchatterjee66@gmail.com

Sandeep Chatterjee