КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ТАРАСА ШЕВЧЕНКА

Факультет комп'ютерних наук та кібернетики

Кафедра теоретичної кібернетики

Кваліфікаційна робота

На здобуття ступеня бакалавра

за спеціальністю 122 Комп'ютерні науки

на тему:

Гомоморфне шифрування для захисту даних в хмарних та туманних технологіях

Виконав студент 4-го курсу
Мальований Дмитро Борисович _____ (підпис)

Науковий керівник:
професор, доктор фіз-мат. наук
Пашко Анатолій Олексійович _____ (підпис)

Засвідчую, що в цій роботі немає запозичень праць інших авторів без відповідних посилань.

Студент _____ (підпис)

Роботу розглянуто й допущено до захисту
на засіданні кафедри
теоретичної кібернетики
"____"_____ 2023р
протокол №____
Завідувач кафедри
Крак Юрій Васильович _____ (підпис)

Київ - 2023

РЕФЕРАТ

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАЧЕННЯ

FHE – Fully homomorphic encryption (Повне гомоморфне шифрування).

SHE – Somewhat homomorphic encryption scheme

RSA – Криптографічний алгоритм з відкритим ключем, який базується, розрахунковій складності великих полупростих чисел.

PKI – Public key infrastructure - набір інструментів які використовують пару (приватний, публічний) ключ, та в якій між користувачами передається тільки публічні ключі, залишаючи приватні анонімними.

Boolean Circuit – Булева схема - це математична модель, що використовується для представлення та обробки булевих функцій. Вона складається з логічних елементів, які з'єднані між собою для виконання логічних операцій над двійковими входами $\{0, 1\}$ та формування двійкових виходів. Схема складається з взаємопов'язаних логічних елементів, таких як (AND, NOT, OR).

ВСТУП

FHE або повне гомоморфне шифрування, це тип шифрування яке дозволяє виконувати розрахунки на зашифрованих даних, не вимагаючи, щоб вони були розшифровані для цього. Результатом розрахунків або ж гомоморфної операції над даними є зашифровані дані, які можуть бути розшифровані ключем з тої ж самої пари, з якої вони були зашифровані.

Завдяки особливості виконувати операції над зашифрованими даними без попереднього дешифрування, FHE стає гарним рішенням в задачах передачі даних в незахищених середовищах, в не авторизованих середовищах, або при передачі над чутливих даних, які не повинні бути видимі для сторони яка займається їх обробкою.

Оцінка сучасного стану об'єкта дослідження або розробки

Вперше технологія FHE була запропонована в 1978 в році, але майже 30 років не було авторитетних досліджень на цю тему і на той момент вже існувала система RSA, яка була краща за багатьма параметрам. Починаючи з 2009 року дослідження та розробки на тему гомоморфного шифрування дуже актуальні й розвиток цієї технології відбувається надзвичайно швидко, покращуючи швидкість виконання операцій, швидкість дешифрування, та розширюючи область застосування шляхом додавання більш комплексних гомоморфних операцій. На цей час існує багато рішень на типові проблеми з використовуванням FHE, які конкурують між собою в різних аспектах, та постійно розвиваються.

Актуальність роботи та підстави для її виконання

Безпечність передачі даних в незахищених середовищах та авторизація отримувача були завжди дуже важливими, рідко хто нехтує цим, оскільки не хоче, щоб їх данні були скомпрометовані або перехоплені. Окрім цього все частіше, за потребою складних обчислень, користувачі звертаються до віддалених машин, також відомі як хмари. Звісно кожен користувач хоче, щоб їх данні були захищенні під час передачі, та хоче бути впевнений, що він передає данні саме туди, куди планував.

Для забезпечення вище описаних вимог, користувач використовує чинні технології, такі як PKI. Єдина не вирішена проблема PKI або інших технологій, це вимога повного дешифрування даних, це означає що при отриманні зловмисником доступу до хмари або віддаленого сервера, у нього буде доступ до не зашифрованих даних. Хоча сучасні хмари дуже добре захищенні, розраховувати на те що зловмисник не зможе отримати до них доступ - не варто.

Для розв'язання проблеми, яку не вирішує PKI, чудово підходить FHE, оскільки сервер, зберігає і виконує операції над даними в зашифрованому вигляді, тому навіть якщо зловмисник отримає доступ до сервера або хмари, отримати дані в нього не вдасться.

Звісно є і деякі обмеження у використанні FHE: по-перше, операції над даними обов'язково повинні бути гомоморфні, по-друге, алгоритм застосування операції над зашифрованими даними дуже повільний. Якщо задача вимагає обробку великої кількості даних, або операція повинна бути не гомоморфна, то можливо краще подумати в сторону застосування інших криптосистем.

Мета й завдання роботи

Написати пізніше

Об'єкт і методи дослідження

об'єкт і методи дослідження або розроблення

Можливі сфери застосування

Гомоморфне шифрування може бути застосоване в будь-якій сфері де потрібна обробка даних, та для виконання цієї задачі використовується віддалений сервер, або хмара. Використання FHE, гарантує безпечну передачу та обробку без попереднього дешифрування даних, але при цьому накладає обмеження на операцію обробки, яка повинна бути гомоморфна, та значно знижує час обробки.

Розділ 1
ОГЛЯД ТЕХНОЛОГІЇ FHE

1.1 Визначення

В цій секції описана термінологія, яка використовується в дослідженнях FHE. Деякі з визначень були взяті напряму з документів FHE, інші були перефразовані для того, щоб спростити формальність і зробити їх більш застосованими до обраної задачі.

Нехай є простір вхідного (чистого) тексту $\mathcal{P} = \{0, 1\}$, та сімейства функцій $F = f_1, f_2, ..., f_n$ де $f_n(x) = f(x_1, x_2, x_3, ..., x_k)$ це Булеві функції k аргументів: $f : P^n \to P$. Ми будемо називати F, сімейством Булевих схем (Boolean circuit) C, і використовувати звичайний запис функції $C(m_1, m_2, ..., m_n)$, для позначення оцінки Булевої схеми на кортежі $(m_1, m_2, ..., m_n)$.

Визначення 1.1.1 ($\mathcal{C}$-схема розрахунків, або ж просто $\mathcal{C}$-схема [2])**.** Нехай $\mathcal{C}$ це множина Булевих схем, тоді $\mathcal{C}$-схема розрахунків, для $\mathcal{C}$ це набір функцій (GEN, ENC, EVAL, DEC) які задовільняють наступним твердженням:

GEN$(1^\lambda, \alpha)$ - алгоритм генерації ключів, на вхід він приймає, параметр шифрування λ, та допоміжний параметр α. Результат виконання алгоритму це триплет ключів (pk, sk, evk), де ключ pk використовується для шифрування, sk для дешифрування, та evk для виконування розрахунків.

ENC(pk, m) - алгоритм шифрування, на вхід він приймає ключ шифрування pk та фрагмент не зашифрованого (чистого) тексту m. Результат виконання алгоритму це шифр c.

EVAL$(evk, C, c_1, c_2..., c_n)$ - алгоритм розрахунків. На вхід він отримує, ключ розрахунків evl та Булеву схему $C \in \mathcal{C}$, та вхідні аргументи, які можуть бути як шифром, так і результатом виконання минулих розрахунків. Результат виконання алгоритму це результат виконання розрахунків.

DEC(sk, c) - алгоритм дешифрування. На вхід приймає, ключ дешифрування sk, та шифр, або результат виконання розрахунків. Результат виконання алгоритму це не зашифрований (чистий) текст m.

Для подальшого опису властивостей, треба визначити простори даних, які є результатами, або вхідними параметрами описаних алгоритмів:

Нехай $\mathcal{X}$ буде описувати простір *чистого шифру*, $\mathcal{Y}$ - простір результатів виконання розрахунків, і $\mathcal{Z} = \mathcal{X} \cup \mathcal{Y}$. $\mathcal{Z}^*$ - містить кортежі довільної довжини, які складаються з елементів $\mathcal{Z}$. Простори ключів згенерованих **GEN**, позначимо як $\mathcal{K}_p, \mathcal{K}_s, \mathcal{K}_e$ для pk, sk, evk відповідно.Алгоритм **GEN** приймає на вхід параметр в унарній нотації 1^λ та опціональний допоміжний параметр λ з простору $\mathcal{A}$. Також, $\mathcal{C}$ містить простір *дозволених* булевих схем, а $\mathcal{P}$, як було зазначено раніше, область вхідного *чистого (незашифрованого)тексту.*

Тепер можна описати область роботи наведених вище алгоритмів:

$$\begin{aligned}
&\textbf{GEN} : \mathbb{N} \times \mathcal{A} \to \mathcal{K}_p \times \mathcal{K}_s \times \mathcal{K}_e \\
&\textbf{ENC} : \mathcal{K}_p \times \mathcal{P} \to \mathcal{X} \\
&\textbf{EVAL}: \mathcal{K}_e \times \mathcal{C} \times \mathcal{Z}^* \to \mathcal{Y} \\
&\textbf{DEC} : \mathcal{K}_s \times \mathcal{Z} \to \mathcal{P}
\end{aligned}$$

Тоді $\mathcal{X}$ та $\mathcal{Y}$ можна визначити наступним чином:

$$\begin{aligned}
\mathcal{X} &= \{c \mid \textbf{ENC}(pk, m) = c,\ m \in \mathcal{P}\} \\
\mathcal{Y} &= \{z \mid \textbf{EVAL}(evk, C, c_1, c_2, ..., c_n) = z,\ c_i \in \mathcal{Z},\ C \in \mathcal{C}\}
\end{aligned}$$

В деяких схемах, ключі розрахунків та шифрування однакові, але часто це і не так, тому в визначеннях було наведено більш спільний випадок.

В оригінальних документах FHE [2] не було зазначено, що алгоритм розшифровування **DEC** повинен мати можливість працювати з результатом виконання алгоритму шифрування **ENC** - $\mathcal{X}$, і було зазначено, що данні можуть бути розшифровані після виконання розрахунків над ними **EVAL** - $\mathcal{Y}$. Для можливості розшифровування, зразу після зашифровування було запропоновано мати *чисту Булеву схему* або ж по суті функцію $f(x) = x$, для виконання розрахунків і отримання даних які вже можна буде розшифровувати. Більшість сучасних FHE схем, дозволяють проводити операції дешифрування даних, над якими не було проведено розрахунків, тому я не буду заглиблюватись в цю тему.

1.1.1 Атрибути та властивості

Тут представлені характеристики методів гомоморфного шифрування. Ми встановлюємо такі властивості, як компактність і конфіденційність схеми, які забороняють спрощені рішення задачі гомоморфного шифрування, з одного боку, і вимагають таких властивостей, як коректність, для того, щоб навіть називати це схемою шифрування.

Визначення 1.1.2 (Коректне розшифровування [1])**.** $\mathcal{C}$-схама має атрибут коректного розшифрування якщо виконується наступне твердження:

$$\mathbf{DEC}(sk, \mathbf{ENC}(pk, m)) = m,$$
$$\text{де } pk, sk, evk \leftarrow \mathbf{GEN}(1^\lambda, \alpha),\ \alpha \in \mathcal{A},\ m \in \mathcal{P}.$$

Це означає, що ми повинні мати можливість безпомилково розшифровувати зашифрований текст.

Визначення 1.1.3 (Коректні розрахунки [1])**.** $\mathcal{C}$-схема коректно розраховує всі Булеві схеми $C \in \mathcal{C}$, якщо виконується наступне твердження:

$$\mathbf{DEC}(sk, \mathbf{EVAL}(evk, C, c_1, c_2, ..., c_n)) \ = \ C(m_1, m_2, ...m_n),$$
$$pk, sk, evk \leftarrow \mathbf{GEN}(1^\lambda, \alpha),\ \alpha \in \mathcal{A},\ c_i \in \mathcal{X} \text{ та } m_i \leftarrow \mathbf{DEC}(sk, c_i)$$

Це визначення означає, що розрахунки над зашифрованими даними з подальшим розшифровуванням повинні бути однакові з результатом розрахунків над не зашифрованими даними.

Будемо називати $\mathcal{C}$-схему *коректною* якщо для неї будуть виконуватись 1.1.2 та 1.1.3 твердження.

Визначення 1.1.4 (Компактність $\mathcal{C}$-схеми)**.** $\mathcal{C}$-схема вважається компактною якщо існує поліном p, такий що, для будь-якого кортежу $(pk, sk, evk) \leftarrow \mathbf{GEN}(1^\lambda, \alpha)$, $\alpha \in \mathcal{A}$, будь-якої Булевої схеми $C \in \mathcal{C}$ та шифру $c_i \in \mathcal{X}$, розмір результату виконання $\mathbf{EVAL}(evk, C, c_1, c_2, ..., c_n)$ не більше від $p(\lambda)$ бітів, в не залежності від Булевої схеми.

Визначення 1.1.4, показує що під час гомоморфних операцій розмір результату не повинен збільшуватись, і залежить тільки від параметра безпеки λ.

Визначення 1.1.5 (Компактно розрахункова $\mathcal{C}$-схема [5])**.** $\mathcal{C}$-схема компактно розраховує всі Булеві схема $C \in \mathcal{C}$, якщо вона компактна 1.1.4 та *коректна.*

Безпека схеми

Далі, важливо зупинитись на безпеці та конфіденційності схеми. Безпеку схеми можна розділити на дві компоненти: семантична безпека, та обфускація схеми. Якщо обфускація використовується коли алгоритм шифрування секретний, і вразливий, то семантична безпека описує розподіл вихідних даних з **EVAL** та **ENC**.

Визначення 1.1.6 (Приватне гомоморфне шифрування схеми [5](2.16))**.** $\mathcal{C}$-схема вважається безпечною, якщо для будь-якого кортежу $(pk, sk, evk) \leftarrow \mathbf{GEN}(1^\lambda, \alpha)$, $\alpha \in \mathcal{A}$, будь-якої Булевої схеми $C \in \mathcal{C}$ та шифру $c_i \in \mathcal{X}$, такого що $m_i \leftarrow \mathbf{DEC}(sk, c_i)$ існує два розподіли:

$$Dist_1 = \mathbf{EVAL}(evk, C, c_1, c_2, ..., c_n)$$
$$Dist_2 = \mathbf{ENC}(pk, C(c_1, c_2, ..., c_n)$$

які повинні бути статистично або обчислювально нерозрізнені. Ці вимоги показують, що розподіл виконання обчислень Булевої схеми над шифром $Dist_1$ повинен бути однаковий (статистично, обчислювально) з розподілом, отриманим шляхом зашифровування *чистого* тексту, який насамперед являється результатом виконання Булевої операції над незашифрованими даними $Dist_2$.

Часто термін безпечної системи можна зустріти як *Сильно гомоморфна система*[4].

1.1.2 Класифікація

Оскільки не всі схеми FHE мають однакові властивості, цей розділ показує як схеми класифікуються, в залежності від того, які схеми вони можуть обчислювати.

Визначення 1.1.7 (Частково Гомоморфна схема або $\mathcal{C}$-Гомоморфізм [2])**.** $\mathcal{C}$-схема називається, частково гомоморфною (SHE), якщо вона має коректне шифрування 1.1.2, та коректне обчислення 1.1.3.

Для частково гомоморфних $\mathcal{C}$-схем нема вимог до компактності, тому з кожним гомоморфним розрахунком розмір вихідного шифру може збільшуватись. Також нема ніяких вимог до множини Булевих операцій які можуть бути використовувані для розрахунків.

Визначення 1.1.8 (Обмежено-рівнева Гомоморфна схема)**.** $\mathcal{C}$-схема називається Обмежено-рівневою, якщо алгоритм генерації ключів **GEN** приймає додатковий параметр $\alpha = d$, який означає максимальну глибину Булевої схеми, яка може бути обчислена. Також застосовані вимоги до компактності, коректності, і те що розмір вихідних даних розрахунків не повинен залежати від d.

Визначення 1.1.9 (Повна Гомоморфна схема)**.** Повною гомоморфною схемою, називають $\mathcal{C}$-схему, до якої застосовані вимоги, коректності, компактності, та вона може обчислювати Булеву схему з множини усіх схем, або ж будь-яку схему.

1.1.3 Композиція розрахунків

Часто, задача потребує декілька послідовних розрахунків, тобто результат певної Булевої схеми повинен слугувати вхідними даними для наступної схеми, або ж простими словами можна це назвати - композиція. Кожну операцію розрахунків над шифром **EVAL** будемо називати *етапом розрахунків.*

З визначення коректних розрахунків 1.1.3 видно що вхідні дані для алгоритму обчислення **EVAL** повинні належати множині $\mathcal{X}$ - або ж множині *чистого шифру*, який є результатом алгоритму **ENC**. Цей розділ описує вимоги, виконуючи які алгоритм розрахунку схеми **EVAL**, може приймати на вхід як результат виконання інших розрахунків $\mathcal{Z}$, так і *чистий шифр* $\mathcal{X}$:

EVAL$(evk, C, c_1, c_2, ..., c_n)$, де $(pk, sk, evk) \leftarrow$ **GEN**$(1^\lambda, \alpha)$, $\alpha \in \mathcal{A}$, $C \in \mathcal{C}$ та $c_i \in \mathcal{X} \cup \mathcal{Z}$

В літературі *розрахунки з етапами* називають ***гомоморфним шифруванням з i-стрибками***(i-hop homomorphic encryption [8], [6])

Визначення 1.1.10 (Розрахунки з етапами)**.** Обчислення $\mathbf{C}_{i,n}$ в i етапів, та шириною n, визначається множиною Булевих схем $\{C_{kl}\}$, де $1 \leq k \leq i, 1 \leq l \leq$ n, та C_{kl} має kn вхідних даних. За вхідними даними $m_{01}, m_{02}, ..., m_{0n}$ ми обчислюємо:

$m_{kl} = C_{kl}(m_{01}, m_{02}, ..., m_{0n}, ..., m_{k-1,1}, ..., m_{k-1,n})$, де $1 \leq k \leq i, 1 \leq l \leq n$.

Результат розрахунків з етапом після **EVAL** та **DEC** буде *чистий текст* $m_{i1}, m_{i2}, ..., m_{in}$. Визначимо початковий *чистий текст* як $\overrightarrow{m_0}$, та вихідний *чистий текст* як $\overrightarrow{m_i}$, тоді можна записати співвідношення $\overrightarrow{m_i} = C_{i,n}(\overrightarrow{m_0})$.

Нехай $(pk, sk, evk) \leftarrow \mathbf{GEN}(1^\lambda, \alpha)$, $\alpha \in \mathcal{A}$, та $c_{i1}, c_{i2}, ..., c_{in} \in \mathcal{X}$, тоді шифр $\{c_{kl}\}$, $1 \leq k \leq i, 1 \leq l \leq n$ обчислюється рекурсивно наступним чином:

$$c_{kl} = \mathbf{EVAL}(evk, C_{kl}, c_{01}, ..., c_{0n}, ..., c_{k-1,1}, ..., c_{k-1,n})$$

Результат розрахунків з етапом над зашифрованими даними, буде шифр $c_{i1}, c_{i2}, ..., c_{in}$. Позначивши початковий (вхідний) шифр як $\overrightarrow{c_0}$ та результівний шифр як $\overrightarrow{c_i}$ можна описати співвідношення яке описує нотацію алгоритму **EVAL** з декількома виходами: $\overrightarrow{c_i} = \mathbf{EVAL}(evk, C_{1,n}, \overrightarrow{c_0})$

З вище описаного визначення 1.1.10 можна зробити висновок що вхідними даними для будь-якого етапу, окрім першого, може бути тільки результат попереднього етапу.

Визначення 1.1.11 (Коректність гомоморфного шифрування з і-стрибками)**.**

1.2 Обмеження

1.3 Відомі області застосування

ВИСНОВКИ

Контент

ПЕРЕЛІК ДЖЕРЕЛ

1. Frederik Armknecht, Colin Boyd, Christopher Carr, Kristian Gjøsteen, Angela Jäschke, Christian A. Reuter, and Martin Strand. A guide to fully homomorphic encryption. Cryptology ePrint Archive, Paper 2015/1192, 2015. https://eprint.iacr.org/2015/1192.

2. Zvika Brakerski and Vinod Vaikuntanathan. Efficient fully homomorphic encryption from (standard) lwe. Cryptology ePrint Archive, Paper 2011/344, 2011. https://eprint.iacr.org/2011/344.

3. Andrei Bulatov. *Boolean Circuits*. duke.edu, https://users.cs.duke.edu/ reif/courses/complectures/Bulatov/32.pdf, Feb 2021.

4. Michael Clear, Arthur Hughes, and Hitesh Tewari. Homomorphic encryption with access policies: Characterization and new constructions. In *Progress in Cryptology – AFRICACRYPT 2013*, pages 61–87. Springer Berlin Heidelberg, 2013.

5. Craig Gentry. *A fully homomorphic encryption scheme*. PhD thesis, Stanford University, 2009. crypto.stanford.edu/craig.

6. Craig Gentry, Shai Halevi, and Vinod Vaikuntanathan. i-hop homomorphic encryption and rerandomizable yao circuits. Cryptology ePrint Archive, Paper 2010/145, 2010. https://eprint.iacr.org/2010/145.

7. Marc Joye. Sok: Fully homomorphic encryption over the [discretized] torus. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2022(4):661–692, Aug. 2022.

8. Ron Rothblum. Homomorphic encryption: From private-key to public-key. In Yuval Ishai, editor, *Theory of Cryptography*, pages 219–234, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

9. Harsha Tirumala Rutgers University, Swastik Kopparty. *Boolean Circuits and Formulas.* rutgers.edu, https://sites.math.rutgers.edu/ sk1233/courses/topics-S20/lec1.1.pdf, 2020.

10. Marten van Dijk, Craig Gentry, Shai Halevi, and Vinod Vaikuntanathan. Fully homomorphic encryption over the integers. Cryptology ePrint Archive, Paper 2009/616, 2009. https://eprint.iacr.org/2009/616.

ДОДАТКИ

контент