УДК 004.056

Вадим ЛОГОШ, Дмитро СМІРНОВ, Роман ХОМЯК

Західноукраїнський національний університет

ПОПУЛЯРНІ БІБЛІОТЕКИ ТА ФРЕЙМВОРКИ ГОМОМОРФНОГО ШИФРУВАННЯ

Вступ. Гомоморфне шифрування дозволяє проводити обчислення безпосередньо над зашифрованими даними. Це дає змогу стороннім сервісам, наприклад, хмарним платформам, обробляти конфіденційну інформацію, не розшифровуючи її. Результат обчислень також залишається зашифрованим і доступним лише власнику секретного ключа, що усуває ризик компрометації даних під час обробки.

Гомоморфні криптосистеми поділяються на частково гомоморфні (PHE), обмежено гомоморфні (SHE/LHE) та повністю гомоморфні (FHE) схеми. Частково гомоморфні забезпечують виконання лише одного типу математичної операції над шифротекстами (лише додавання *або* лише множення). Обмежено гомоморфні дозволяють обмежену кількість як додавань, так і множень (наприклад, довільна кількість додавань і лише одне множення), тобто підтримують обидва типи операцій, але з обмеженою глибиною складання операцій.

Повністю гомоморфні шифрування не мають таких обмежень – вони дозволяють виконувати будь-які обчислення (довільні схеми з додаваннями і множеннями) над зашифрованими даними необмежену кількість разів. FHE-системи, по суті, забезпечують можливість реалізувати на шифротекстах довільну функцію (вони є тюринг-повними), тоді як PHE та обмежені SHE – ні.

Мета. Аналіз сучасних бібліотек та фреймворків гомоморфного шифрування та порівняння їх основних характеристик.

1. Аналіз бібліотек та фреймворків гомоморфного шифрування

Сьогодні існує ряд відкритих бібліотек, які реалізують схеми і надають розробникам зручні інструменти для роботи з гомоморфним шифруванням. Нижче проаналізовано найпоширеніші з них та дано коротку характеристику.

1. Microsoft SEAL. Популярна відкрита бібліотека від Microsoft, що підтримує схеми BFV (Brakerski/Fan–Vercaut.) та CKKS(Cheon -Kim et al.). Орієнтована на простоту використання: надає високорівневий API для виконання гомоморфних обчислень, дозволяючи будувати повністю зашифровані сховища даних і сервіси обробки без розкриття ключів [1].

Написана на C++ (доступні обгортки для .NET, Python), оптимізована для швидкодії, має детальну документацію і приклади.

2. PALISADE. Бібліотека з відкритим кодом, розроблена консорціумом за підтримки DARPA. Підтримує кілька гомоморфних схем – BGV, BFV, CKKS, а також бульові TFHE/FHEW, у тому числі в багатокористувацькому (Multiparty) режимі. Відрізняється модульною архітектурою і гнучкістю налаштувань. На основі PALISADE у 2022 р. створено нову об'єднану платформу OpenFHE.

3. Helib. Одна з перших FHE-бібліотек, розроблена IBM (перший випуск – 2013/2014, публічний реліз – 2016 р.). Реалізує схеми BGV і CKKS (додані

пізніше) та підтримує bootstrapping для BGV. Написана на C++ з відкритим кодом, HElib була націлена передусім на дослідників, надаючи гнучкий, хоч і відносно низькорівневий інтерфейс. IBM продовжує вдосконалювати HElib, зокрема оптимізуючи швидкодію.

4. Бібліотека TFHE. Спеціалізована бібліотека для схеми TFHE (Torus FHE). Розроблена командою дослідників (Chillotti, Gama, Georgieva, Izabachène) і вперше опублікована у 2016–2017 рр. Орієнтована на швидкі булеві операції з частим bootstrapping. Забезпечує виконання логічних схем (AND, OR, XOR тощо) над шифрованими бітами за кілька мілісекунд. Реалізована на C++, використовує швидку операцію БПФ над тором і інші оптимізації. Підтримує також багатокористувацький режим. Недоліком є обмеженість до побітових операцій – для роботи з великими числами чи векторами рекомендується комбінувати її з іншими бібліотеками (або використовувати гібридні підходи).

5. HEAAN (HeaAn). Відкрита бібліотека для схеми CKKS, розроблена дослідниками Сеульського національного університету (авторами CKKS). Назва розшифровується як *"*Homomorphic Encryption for Arithmetic of Approximate Numbers*"*. Перший випуск – 2016 р., згодом підтримку проекту продовжила корейська компанія CryptoLab [3].

HEAAN реалізує всі основні можливості CKKS: гомоморфні додавання, множення, масштабування, пакування/розпакування векторів. В ній однією з перших з'явилася реалізація bootstrapping для CKKS, що дозволяє виконувати необмежену кількість операцій на зашифрованих речових числах. HEAAN оптимізована для високої точності і швидкості обчислень, підтримує GPU-акселерацію для основних операцій над поліномами. Її часто використовують у дослідженнях, пов'язаних з приватними обчисленнями в AI, оскільки вона швидко впроваджує найновіші алгоритмічні вдосконалення CKKS.

6. OpenFHE. новітній фреймворк (перший реліз – липень 2022) для гомоморфного шифрування, який об'єднує напрацювання кількох попередніх бібліотек [4].

OpenFHE розроблено командою експертів з різних установ під егідою організації Duality Technologies, за участі спільноти (проект під патронатом NumFocus). Бібліотека є наступником PALISADE і включає в себе підтримку всіх основних схем: BGV, BFV, CKKS для арифметичних обчислень, а також схем TFHE і FHEW для булевих операцій. OpenFHE від початку спроектована з урахуванням можливості *bootstrapping* для всіх схем (тобто підтримує перезавантаження і для BGV/BFV/CKKS, чого раніше не було «з коробки») [4].

Великі технологічні компанії вкладаються в розвиток FHE, розробляючи інструменти для спрощення його використання. Наприклад, Microsoft випустила бібліотеку SEAL, яка допомагає інтегрувати гомоморфне шифрування у прикладні рішення (вже реалізовані пілотні проекти з повністю зашифрованого аналізу даних у партнерстві з фінтех-компаніями). Google розробила інструмент Private Join and Compute для захищеного спільного аналізу даних, а також FHE Transpiler – компілятор, що перетворює звичайний код на еквівалентні гомоморфні обчислення [5].

IBM активно працює над прискоренням HElib і пропонує хмарні сервіси з підтримкою FHE. Отже, є підстави вважати, що повністю гомоморфне

шифрування поступово виходитиме за межі дослідницьких лабораторій і інтегруватиметься у реальні системи, забезпечуючи новий рівень безпеки даних.

В таблиці 1 наведено основні гомоморфні схеми шифрування – як часткові (PHE), так і повні (FHE). Порівнюються їх тип, підтримувані гомоморфні операції, криптографічна основа, стійкість та рік появи.

Таблиця 1.1 – Основні гомоморфні схеми шифрування

Схема	Тип шифрування	Підтримувані операції	Безпека (основа)	Рік
RSA	PHE (мультиплікативна)	Множення шифротекстів (без обмежень)	Факторизація цілого n (не стійка проти квантових атак)	1978
Paillier	PHE (адитивна)	Додавання шифротекстів; множення на константу	Композитний модуль (n^2); (не постквантова)	1999
Gentry FHE	FHE (повна схема)	Додавання, множення (необмежено завдяки bootstrapping)	Ідеальні ґратки (LWE) + підмножина суми; постквантова	2009
BGV	FHE (на ґратках)	Додавання, множення (рівнева або з bootstrap)	RLWE (кільцеві ґратки); постквантова	2011
BFV	FHE (на ґратках)	Додавання, множення (рівнева або з bootstrap)	RLWE (scale-invariant варіант); постквантова	2012
TFHE	FHE (булева)	Булеві операції (бітові над шифротекстами)	Гратки GSW на торі (реал. в кільці); постквантова	2016
CKKS	FHE (наближені обчислення)	Додавання, множення над зашифрованими дійсними числами	RLWE (дод. округлення при операціях); постквантова	2017

Висновок. Досліджено стан розвитку алгоритмів гомоморфного шифрування. Проведено порівняльний аналіз сучасних бібліотек та фреймворків гомоморфного шифрування. Розкрито можливості та обмеження, зокрема, підтримувані гомоморфні операції, криптографічна основа, та стійкість.

Перелік використаних джерел.

1. Microsoft SEAL. Режим доступу: https://www.microsoft.com/en-us/research/project/microsoft-seal/#:~:text=Microsoft%20SEAL-powered%20by%20open,their%20key%20with%20the%20service

2. FHE Libraries: Established Cryptographic Building Blocks. https://el3ctrum.com/uncategorized/fhe-libraries-established-cryptographic-building-blocks/#:~:text=Background%20%26%20Developers%3A%20HEAAN%20stands,the%20library's%20performance%20and%20features

3. OpenFHE. Режим доступу: https://openfhe.org

4. Private Join and Compute. Режим доступу: https://github.com/google/private-join-and-compute