УДК 004.056

ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ ЗАСТОСУВАННЯ ГОМОМОРФНИХ КРИПТОСИСТЕМ В РЕКОМЕНДАЦІЙНИХ СИСТЕМАХ ВЕБ-СЕРВІСІВ

DOI 10.36994/2788-5518-2021-02-02-14

Корчинський В.В., *д.т.н., проф. Державний університет інтелектуальних технологій і зв'язку, Одеса, Україна. vladkorchin@ukr.net*

Кільдішев В.Й., *к.т.н., доц. Державний університет інтелектуальних технологій і зв'язку, Одеса, Україна. kildishev@ukr.net*

Онищук В.В., *Державний університет інтелектуальних технологій і зв'язку, Одеса, Україна. hjekarstone@gmail.com*

Аль-Файюми Халед, *Державний університет інтелектуальних технологій і зв'язку, Одеса, Україна. khaled@alfaiomi.com*

Анотація. *У статті розглядається імітаційна модель веб-сервісів з гомоморфною криптосистемою. Існуючі рекомендаційні системи веб-сервісів для персоналізації контенту потребують обробки великого об'єму конфіденційної інформації користувачів. Такі веб-сервіси мають можливість зчитувати дані у незашифрованому виді, що є основним їх недоліком і збільшує ймовірність несанкціонованого доступу третіми особами. Для завдання імітаційного моделювання веб-сервісів була обрана відповідна обчислювальна система із заданими параметрами і реалізації повністю гомоморфних криптосистем у складі програмної бібліотеки PALISADE. У дослідженні використовувались криптосистеми Brakerski-Gentry-Vaikuntanathan (BGV), Brakerski/Fan-Vercauteren (BFV), Cheon-Kim-Kim-Song (CKKS) та FHEW. На основі розробленої програми мовою C++ визначено об'єм пам'яті для зберігання шифротекстів та часові витрати на виконання математичних операцій повністю гомоморфними криптосистемами. Аналіз продуктивності сучасних повністю гомоморфних криптосистем показав, що суттєве сповільнення простих математичних операцій робить недоцільним використання рекомендаційних систем із-за потреби надвеликої кількості обчислювальних ресурсів. Проведені дослідження на основі побудованої імітаційної моделі показали, що невелика надійність і повільно діюча обробка даних пов'язані з великою складністю алгоритмів цих систем. Це доводить необхідність спрощення програмного коду моделі рекомендаційної системи і повністю гомоморфної криптосистеми для зменшення кількості операцій, що виконуються веб-сервісом. Також, рекомендаційна система не може взаємодіяти з зашифрованими даними без допомоги повністю гомоморфної криптосистеми. Для вирішення цієї проблеми потрібно зробити заміну певної кількості математичних функцій у програмному коді рекомендаційної системи на ті, що забезпечать обробку шифротекстів.*

Ключові слова: *веб-сервіс, гомоморфні криптосистеми, рекомендаційні системи, дані користувачів, швидкість.*

RESEARCH OF THE EFFICIENCY APPLICATION OF HOMOMORPHIC CRYPTOSYSTEMS IN RECOMMENDATION SYSTEMS OF WEB SERVICES

Korchynskyi Volodymyr, *Dr.habil. Ass. Prof. State University of Intellectual Technologies and Communications, Odessa, Ukraine. vladkorchin@ukr.net*

Kildishev Vitalii, *Ph.D,, Ass. Prof. State University of Intellectual Technologies and Communications, Odessa, Ukraine. kildishev@ukr.net*

Onyshchuk Valentyn, *State University of Intellectual Technologies and Communications, Odessa, Ukraine. hjekarstone@gmail.com*

Khaled Alfaion., *State University of Intellectual Technologies and Communications, Odessa, Ukraine . khaled@alfaiomi.com*

Abstract. *The article is considered the simulation model of web service with homomorphic cryptosystem. Existing recommendation systems of web services for content personalization is required the processing of large amounts of confidential user information. Such web services have the ability to read data in unencrypted form, which is their main drawback and increases the probability of unauthorized access by third parties. For the task of simulation modeling of the web service was chosen the appropriate computer system with the set parameters and realization of completely homomorphic cryptosystems as a part of the PALISADE software library. The following list of cryptosystems was used: Brakerski-Gentry-Vaikuntanathan (BGV); Brakerski/Fan-Vercauteren (BFV); Cheon-Kim-Kim-Song (CKKS); FHEW. Based on the developed program in C ++ is determined the amount of memory for storing encrypted texts and performed time measurements, which is spent completely homomorphic cryptosystems to perform mathematical operations. The computational duration of fully homomorphic cryptosystems was measured using the standard C ++ library's "std: chrono" API, and the amount of encrypted text was measured using the "massif" utility, which is included in Valgrind software for Linux-based operating systems. Analysis of the productivity of modern fully homomorphic cryptosystems has shown that significant slowdown in simple mathematical operations makes it virtually impossible to use recommendation systems due to the need for too many computing resources. Conducted researches based on the simulation model have shown that, the low reliability and slow data processing are associated with the high complexity algorithms of these systems. This proves the necessary to simplify the software code model of the recommendation system and completely homomorphic cryptosystem to reduce the number of operations, which are performed by the web service. Also, the recommendation system cannot interact with encrypted data without help of completely homomorphic cryptosystem. To solve this problem, it is necessary to replace a certain number of mathematical functions in the program code of the recommendation system with those that will provide processing of encrypted texts.*

Keywords: *web service, homomorphism cryptosystems, recommendation systems, user data, speed.*

Вступ

Існуючі рекомендаційні системи веб-сервісів для персоналізації контенту потребують обробки великого об’єму конфіденційної інформації користувачів [1-3]. Такі веб-сервіси мають можливість зчитувати незашифровані дані

користувачів, що є основним їх недоліком і збільшує ймовірність несанкціонованого доступу третіми особами [1, 2]. У 2009 році Крейгом Джентрі була здійснена спроба вирішити проблему по забезпеченню довіри користувачів до веб-сервісів шляхом створення першої робочої схеми повністю гомоморфної криптосистеми (ПГК) [4]. На сьогодні основними недоліками існуючих схем ПГК є низька надійність [5] та мала швидкість обробки [6] даних, що, як наслідок, потребує надвелику кількість обчислювальних ресурсів. В роботі [7] запропоновано різні програмні реалізації існуючих схем ПГК, що спрямовані на спробу вирішити зазначені недоліки. Проте, завдання по збільшенню швидкодії та надійності обробки даних залишаються відкритими, що пов'язано з великою складністю алгоритмів як ПГК [5, 6], так і рекомендаційних систем [12]. Тому виникає завдання оптимізації алгоритмів рекомендаційних систем і ПГК, що потребує детального дослідження і вдосконалення їх програмних кодів. Існують такі програмні бібліотеки, що реалізують ПГК: PALISADE, Microsoft SEAL, HElib, FHEW/TFHE та інші.

PALISADE [8] підтримує майже всі сучасні схеми ПГК: Brakerski-Gentry-Vaikuntanathan (BGV), Brakerski/Fan-Vercauteren (BFV), Cheon-Kim-Kim-Song (CKKS) та FHEW. Перевагами PALISADE є постійне її вдосконалення великими IT-стартапами. Таку програмну бібліотеку можна використовувати для дослідження особливостей впровадження і продуктивності роботи ПГК.

За допомогою бібліотеки PALISADE було створено середовище шифрувальника – об'єкт-клас "CryptoContext". Після цього було обрано вид ПГК та задані її параметри. Далі було згенеровано ключі шифрування для проведення відповідних операцій. Слід зазначити, що схеми ПГК не підтримують роботу з даними у їх початковому форматі, тому перед шифруванням потрібно їх перетворення:

1) BFV та BGV у цілочисленні вектори;
2) CKKS у вектори дійсних/комплексних чисел;
3) FHEW у логічну (булеву) схему.

Далі за допомогою засобів PALISADE потрібно створити об'єкт "Plaintext" і помістити туди створені вектори. Для збереження логічних схем призначено об'єкт "LWEPlaintext". Після цього за допомогою функції "encrypt" об'єкта "CryptoContext" відбувається шифрування і зберігання результату у об'єкт "Ciphertext". Також в цьому об'єкті міститься не тільки шифротекст, а і відомості про параметри об'єкту "CryptoContext", за допомогою якого були зашифровані дані. Це дозволяє на будь-якому іншому пристрої створити об'єкт "CryptoContext" з такими самими параметрами і розшифрувати дані за допомогою ключа. Функція "decrypt" об'єкта "CryptoContext" виконує розшифрування даних, тобто перетворює об'єкт "Ciphertext" у "Plaintext". Слід зазначити, що ПГК не тільки виконують математичні операції, а і усувають накопичений шум у шифротекстах. Тому веб-сервіси не зможуть виконувати обробку об'єктів "Ciphertext" (шифротекстів) за допомогою стандартних математичних функцій мов програмування. Для таких задач використовуються

функції бібліотеки PALISADE. Це означає, що необхідно у початковому коді рекомендаційної системи веб-сервісу замінити певні математичні функції на аналогічні з бібліотеки PALISADE. Таким чином, відбувається інтеграція PALISADE у рекомендаційну систему веб-сервісу.

У PALISADE загальні особливості роботи майже не відрізняються від інших програмних бібліотек з реалізованими ПГК.

Для завдання імітаційного моделювання веб-сервісу обрано комп'ютер з чотириядерним процесором Intel Core i5-3550 3,3 GHz, оперативною пам'яттю DDR3 об'ємом 16 ГБ, операційною системою (ОС) Windows 10 Home версії 21H1 та середовищем для запуску гостьових ОС – Oracle VM Virtualbox 6.1.26 r145957 (надалі Virtualbox). Для роботи Virtualbox було виділено чотири ядра CPU та 4 ГБ оперативної пам'яті. У середовище Virtualbox встановлено KDE Neon (ОС на основі Linux) і PALISADE.

На основі розробленої тестової програми (C++) виконано заміри:

1) часу у секундах, що витрачають ПГК у складі PALISADE на виконання математичних операцій;

2) об'єму пам'яті у байтах, що була зайнятою шифротекстами.

Вимірювання тривалості обчислень ПГК виконувалось за допомогою API стандартної бібліотеки мови C++ "std:chrono" [9]. Обчислення об'єму шифротекстів відбувалось за допомогою утиліти "massif", яка входить у програмне забезпечення Valgrind [11] для ОС на основі Linux, так як стандартний метод C++ "sizeof" [10] це не дозволяє.

Дослідження продуктивності побудованої імітаційної моделі веб-сервісу з гомоморфним шифруванням

Для тестування ПГК BGV та BFV було обрано три цілочисельних вектори V1, V2 та V3. Кожен вектор складався з 12 цілих чисел. Було розраховано суму векторів (V1+V2+V3), добуток векторів (V1 × V2 × V3) та циклічний чотирикратний зсув елементів вектора V1 з переносом. В табл. 1 надано результати швидкості обчислень з використанням та без використання ПГК. Порівняння зайнятих просторів оперативної пам'яті результатами обчислень з ПГК та без ПГК представлено в табл. 2.

Таблиця 1.

Швидкість обчислень з використанням та без використання ПГК

Операція з векторами	Час обчислення (с)		
	Без ПГК за допомогою стандартних функцій мови C++	BGV	BFV
V1 + V2 + V3	4,69 × 107	7,91 × 10-4	1,35 × 10-3
V1 × V2 × V3	4,35 × 107	7,83 × 10-2	1,11 × 10-1
Зсув елементів V1	1,46 × 106	2,07 × 10-2	9,7 × 10-2

Таблиця 2.

Порівняння зайнятих просторів оперативної пам'яті результатами обчислень з ПГК та без ПГК

Операція з векторами	Об'єм пам'яті, зайнятої вектором або класом Ciphertext, що містив результат обчислення (байт)		
	Без ПГК за допомогою стандартних функцій мови C++	BGV	BFV
V1 + V2 + V3	96	393,216	393,216
V1 × V2 × V3	96	262,144	196,608
Зсув елементів V1	96	131,072	196,608

Для тестування ПГК CKKS було обрано два вектори з дійсними числами - V1 та V2. Кожен вектор складався з 8 дійсних чисел. Було розраховано (табл. 3) суму векторів (V1 + V2), добуток векторів (V1 × V2) та циклічний двократний зсув елементів вектора V1 з переносом. В табл. 4 надано результати порівняння зайнятих просторів оперативної пам'яті результатами обчислень з ПГК та без ПГК

Таблиця 3.

Швидкість обчислень з використанням та без використання ПГК

Операція з векторами	Час обчислення (с)	
	Без ПГК за допомогою стандартних функцій мови C++	CKKS
V1 + V2	5,14 × 107	7,19 × 10-4
V1 × V2	2,79 × 107	7,83 × 10-2
Зсув елементів V1	9,13 × 107	9,91 × 10-2

Таблиця 4.

Порівняння зайнятих просторів оперативної пам'яті результатами обчислень з ПГК та без ПГК

Операція з векторами	Об'єм пам'яті, зайнятої вектором або класом Ciphertext, що містив результат обчислення (байт)	
	Без ПГК за допомогою стандартних функцій мови C++	CKKS
V1 + V2	64	262,144
V1 × V2	64	131,072
Зсув елементів V1	64	131,072

Для тестування ПГК FHEW було обчислено логічну схему (1 AND 1) OR (1 AND (NOT 1)). Результати обчислень надано в табл. 5 і 6.

Таблиця 5.

Швидкість обчислень з використанням та без використання ПГК

Булева схема	Час обчислення (с)	
	Без ПГК за допомогою стандартних функцій мови C++	FHEW
Логічна схема	5,9 × 108	0,849331

Таблиця 6.

Порівняння зайнятих просторів оперативної пам'яті результатами обчислень з ПГК та без ПГК

Булева схема	Об'єм пам'яті, зайнятої булевським об'єктом C++ або класом Ciphertext, що містив результат обчислення (байт)	
	Без ПГК за допомогою стандартних функцій мови C++	FHEW
Логічна схема	1	8,192

Висновки

Результати досліджень продуктивності сучасних ПГК показали, що суттєве сповільнення простих математичних операцій в 103 – 105 разів зробить практично неможливим використання рекомендаційних систем.

Для бази з незашифрованими даними з 50000 унікальних записів створення списку персоналізованих пропозицій може зайняти у рекомендаційної системи приблизно від кількох секунд до десятків хвилин.

Об'єм пам'яті для зберігання зашифрованих даних за допомогою ПГК збільшується у порівнянні з незашифрованими даними у 2–8 рази. Це доводить необхідність спрощення програмного коду моделі рекомендаційної системи і ПГК для зменшення кількості операцій, що виконуються веб-сервісом.

Рекомендаційна система не може взаємодіяти з зашифрованими даними без допомоги ПГК. Для вирішення цієї проблеми потрібно зробити заміну певної кількості математичних функцій у програмному коді РС на ті, що забезпечать обробку шифротекстів.

Література

1. DeLeon, Haley. The Ethical and Privacy Issues of Recommendation Engines on Media Platforms: [Електронний ресурс]. – Режим доступу: https://towardsdatascience.com/the-ethical-and-privacy-issues-of-recommendation-engines-on-media-platforms-9bea7bcb0abc

2. Chen, Li. Evaluating recommender systems from the user's perspective: survey of the state of the art / Pearl Pu, Li Chen, Rong Hu. – Springer Science+Business Media B.V., 2012. – 332–344 pp.

3. Kasula, Chaithanya. Netflix Recommender System. A Big Data Case Study [Електронний ресурс]. – Режим доступу: https://towardsdatascience.com/netflix-recommender-system-a-big-data-case-study-19cfa6d56ff5

4. Gentry, Craig. A Fully Homomorphic Encryption Scheme (Ph.D. thesis) / Craig Gentry. – Symposium on the Theory of Computing (STOC), 2009. – 169-178 pp.

5. Li, Baily. On the Security of Homomorphic Encryption on Approximate Numbers [Текст] / Baily Li, Daniele Micciancio. – IACR ePrint Archive, 2020.
6. Acar, Abbas. A Survey on Homomorphic Encryption Schemes: Theory and Implementation [Текст] / Abbas Acar, Hidayet Aksu, A. Selcuk Uluagac, Mauro Conti. – ACM Computing Surveys, Vol. 51, No. 4, Article 79, 2018.
7. Homomorphic Encryption Standardization. An Open Industry, Government, Academic Consortium to Advance Secure Computation: [Електронний ресурс]. – Режим доступу: https://homomorphicencryption.org/
8. PALISADE Homomorphic Encryption Software Library: [Електронний ресурс]. – Режим доступу: https://palisade-crypto.org/
9. Data & time utilities – std::chrono: [Електронний ресурс]. – Режим доступу:
10. https://en.cppreference.com/w/cpp/chrono
11. Sizeof operator: [Електронний ресурс]. – Режим доступу:
12. https://en.cppreference.com/w/cpp/language/sizeof
13. Valgrind – instrumentation framework for building dynamic analysis tools: [Електронний ресурс]. – Режим доступу: https://valgrind.org/
14. Онищук, В.В. Підвищення ефективності модуля push-повідомлень online-сервісів шляхом персоналізації контенту та сегментації клієнтської бази. [Текст] : магістерська робота / В.В. Онищук. – ОНАЗ ім. О.С. Попова, 2020. – С. 42–62.

References

1. DeLeon, Haley. The Ethical and Privacy Issues of Recommendation Engines on Media Platforms: [Elektronnyi resurs]. – Rezhym dostupu: https://towardsdatascience.com/the-ethical-and-privacy-issues-of-recommendation-engines-on-media-platforms-9bea7bcb0abc
2. Chen, Li. Evaluating recommender systems from the user's perspective: survey of the state of the art / Pearl Pu, Li Chen, Rong Hu. – Springer Science+Business Media B.V., 2012. – 332–344 pp.
3. Kasula, Chaithanya. Netflix Recommender System. A Big Data Case Study [Elektronnyi resurs]. – Rezhym dostupu: https://towardsdatascience.com/netflix-recommender-system-a-big-data-case-study-19cfa6d56ff5
4. Gentry, Craig. A Fully Homomorphic Encryption Scheme (Ph.D. thesis) / Craig Gentry. – Symposium on the Theory of Computing (STOC), 2009. – 169-178 pp.
5. Li, Baily. On the Security of Homomorphic Encryption on Approximate Numbers [Tekst] / Baily Li, Daniele Micciancio. – IACR ePrint Archive, 2020.
6. Acar, Abbas. A Survey on Homomorphic Encryption Schemes: Theory and Implementation [Tekst] / Abbas Acar, Hidayet Aksu, A. Selcuk Uluagac, Mauro Conti. – ACM Computing Surveys, Vol. 51, No. 4, Article 79, 2018.
7. Homomorphic Encryption Standardization. An Open Industry, Government, Academic Consortium to Advance Secure Computation: [Elektronnyi resurs]. – Rezhym dostupu: https://homomorphicencryption.org/
8. PALISADE Homomorphic Encryption Software Library: [Elektronnyi resurs]. – Rezhym dostupu: https://palisade-crypto.org/
9. Data & time utilities – std::chrono: [Elektronnyi resurs]. – Rezhym dostupu:
10. https://en.cppreference.com/w/cpp/chrono
11. Sizeof operator: [Elektronnyi resurs]. – Rezhym dostupu:
12. https://en.cppreference.com/w/cpp/language/sizeof
13. Valgrind – instrumentation framework for building dynamic analysis tools: [Elektronnyi resurs]. – Rezhym dostupu: https://valgrind.org/
14. Onyshchuk, V.V. Pidvyshchennia efektyvnosti modulia push-povidomlen online-servisiv shliakhom personalizatsii kontentu ta sehmentatsii kliientskoi bazy. [Tekst] : mahisterska robota / V.V. Onyshchuk. – ONAZ im. O.S. Popova, 2020. – s. 42–62.