Міністерство освіти і науки України

Харківський національний університет радіоелектроніки

Факультет комп’ютерної інженерії та управління
(повна назва)

Кафедра електронних обчислювальних машин
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА

Пояснювальна записка

Рівень вищої освіти другий (магістерський)

Розробка та впровадження механізмів захисту персональних даних у хмарних обчисленнях

(тема)

Виконав:

студент II курсу, групи СПм-23-2

Попов А.О.
(прізвище, ініціали)

Спеціальність

123 «Комп’ютерна інженерія»
(код і повна назва спеціальності)

Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма

Комп’ютерні системи та мережі
(повна назва освітньої програми)

Керівник: доц. Сорокін А.Р.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри ЕОМ _____ Коваленко А.А.
(підпис) (прізвище, ініціали)

2025 р.

Харківський національний університет радіоелектроніки

Факультет комп’ютерної інженерії та управління

Кафедра електронних обчислювальних машин

Рівень вищої освіти другий (магістерський)

Спеціальність 123 «Комп’ютерна інженерія»
(код і повна назва)

Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма Комп’ютерні системи та мережі
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
“____” _____ 20__ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

студенту Попову Артему Олексійовичу
(прізвище, ім’я, по батькові)

1. Тема роботи Розробка та впровадження механізмів захисту персональних даних у хмарних обчисленнях

затверджена наказом по університету від “ 22 ” листопада 2024 р. № 1236 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 20 січня 2025 р.

3. Вхідні дані до роботи 1) Тема роботи; 2) Хмарні сервіси; 3) Методи шифрування

4. Перелік питань, що потрібно опрацювати у роботі 1) Аналіз предметної області 2) Принципи побудови захисного комплексу для хмарних баз даних 3) Реалізація комплексу технології для забезпечення безпеки хмарних баз даних 4) Реалізація програми 5) Висновки

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) 18 слайдів

6. Консультанти розділів роботи (заповнюється за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Аналіз проблеми та огляд існуючих рішень	26.11.24-01.12.24	
2	Вибір технології розробки та	02.12.24-05.12.24	
	інструментальних засобів		
3	Розробка алгоритмічного забезпечення	06.12.24-20.12.24	
4	Розробка програмних модулів	21.12.24-30.12.24	
5	Відлагодження програмних модулів	31.12.24-07.01.25	
6	Оформлення матеріалів кваліфікаційної роботи	08.01.25-15.01.25	
7	Подання кваліфікаційної роботи керівникові та	16.01.25-19.01.25	
	її попередній захист		
8	Подання кваліфікаційної роботи на	20.01.25	

Дата видачі завдання 25 листопада 2024 р.

Студент _____
(підпис)

Керівник роботи _____ доц. Сорокін А.Р.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 72 с., 6 рис., 4 табл., 2 дод., 10 джерел.

ХМАРНІ ОБЧИСЛЕННЯ, ЗАХИСТ ДАНИХ, КРИПТОГРАФІЯ, МАШИННЕ НАВЧАННЯ, RSA, K-MEANS, КОНФІДЕНЦІЙНІСТЬ, КЛАСТЕРИЗАЦІЯ.

Метою кваліфікаційної роботи є розробка та впровадження механізмів захисту персональних даних у хмарних обчисленнях, що поєднує криптографічні методи для забезпечення конфіденційності даних та алгоритми машинного навчання для їх аналізу.

У ході виконання кваліфікаційної роботи планується розробити механізми, які поєднуватимуть криптографічні методи для забезпечення конфіденційності даних та алгоритми машинного навчання для їх аналізу. Основна увага буде приділена реалізації двох ключових компонентів: шифрування даних за допомогою алгоритму RSA та кластеризації даних за допомогою алгоритму K-Means.

Очікувані результати включають створення механізмів, які демонструватимуть ефективне поєднання криптографічних методів та алгоритмів машинного навчання. Ці механізми можуть бути використані для захисту та аналізу даних у хмарних середовищах, що є критично важливим для сучасних організацій, які працюють з великими обсягами чутливої інформації.

ABSTRACT

Master’s thesis: 72 pages, 6 figures, 4 tables, 2 appendices, 10 sources.

CLOUD COMPUTING, DATA PROTECTION, CRYPTOGRAPHY, MACHINE LEARNING, RSA, K-MEANS, CONFIDENTIALITY, CLUSTERING.

The purpose of the qualification work is to develop and implement mechanisms for protecting personal data in cloud computing, combining cryptographic methods to ensure data confidentiality and machine learning algorithms for their analysis.

During the qualification work, it is planned to develop mechanisms that will combine cryptographic methods to ensure data confidentiality and machine learning algorithms for their analysis. The main attention will be paid to the implementation of two key components: data encryption using the RSA algorithm and data clustering using the K-Means algorithm.

The expected results include the creation of mechanisms that will demonstrate the effective combination of cryptographic methods and machine learning algorithms. These mechanisms can be used to protect and analyze data in cloud environments, which is critical for modern organizations that work with large amounts of sensitive information.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

ОС – операційна система

BP-XOR – Ворота BP-XOR (Belief Propagation XOR Gates).

ЕОМ – Електронні обчислювальні машини.

IaaS – Інфраструктура як послуга.

K-Means – Алгоритм кластеризації даних.

LDPC – Коди з низькою щільністю перевірок на парність.

LT – Трансформаційні коди Лубі.

PaaS – Платформа як послуга.

RDA – Віддалений аудит даних.

RSA – Криптографічний алгоритм Рівеста-Шаміра-Адлемана.

SaaS – Програмне забезпечення як послуга.

ВСТУП

Хмарні обчислення — це одна з найбільш революційних технологій сучасності, яка кардинально змінює підходи до обробки даних, зберігання інформації та розгортання програмних рішень. Дані технології мають на меті дозволити користувачам через мережу Інтернет отримувати доступ відкриває нові можливості для бізнесу, досліджень та повсякденного життя. Основна ідея полягає в тому, що користувачі можуть використовувати потужні обчислювальні ресурси, не володіючи фізичною інфраструктурою, такою як сервери чи сховища даних. Це робить хмарні обчислення доступними для потенційної аудиторії якою є корпорації, а також малі підприємства.

Такі обчислення базуються на моделі, яка забезпечує зручний доступ до спільного пулу ресурсів, таких як обчислювальні потужності, мережі та програмне забезпечення. Ці ресурси можуть швидко надаватися та звільнятися, що робить хмару ідеальним рішенням для організацій, які потребують гнучкості та масштабованості. Користувачі можуть зосередитися на своїх основних завданнях, не турбуючись про технічну підтримку чи оновлення інфраструктури. Наприклад, компанія може швидко розгорнути новий додаток або збільшити обсяги зберігання даних, не інвестуючи у дороге обладнання.

Однією з ключових переваг хмарних обчислень є економія витрат. Організації можуть уникнути значних капітальних інвестицій у власну інфраструктуру, оплачуючи лише ті ресурси, які вони фактично використовують. Це особливо важливо для стартапів та малого бізнесу, які часто обмежені у фінансових ресурсах. Крім того, хмара дозволяє швидко адаптуватися до змінних потреб, що особливо важливо для бізнесу, який стикається зі сезонними коливаннями навантаження. Наприклад, інтернет-магазин може збільшити обчислювальні потужності під час різдвяних розпродажів, а потім зменшити їх, коли навантаження знизиться.

Хоча хмарні обчислення пропонують багато можливостей, вони також несуть у собі серйозні загрози. Головною проблемою є безпека даних, які можуть стати легкою здобиччю для хакерів через їхнє віддалене зберігання. Крім того, існує ризик порушення конфіденційності, особливо коли дані зберігаються за межами країни та підпадають під дію інших законів про захист даних. Це створює правові колізії, наприклад, для європейських компаній, які використовують американські хмарні сервіси.

Іншим важливим аспектом є залежність від інтернет-з'єднання. Якщо інтернет-з'єднання відсутнє або повільне, доступ до хмарних ресурсів стає неможливим. Це може бути критичним для організацій, які працюють у віддалених регіонах або мають високі вимоги до доступності даних. Крім того, користувачі хмарних послуг часто не мають повного контролю над інфраструктурою, що може бути проблемою для організацій із специфічними вимогами до безпеки чи конфігурації.

Хмарні обчислення також класифікуються за типами середовищ, які включають гібридні, приватні та публічні хмари. Наприклад, публічні хмари є у доступі для загального використання, в той час як приватними хмарами може користуватись тільки одна організація. А ось гібридні хмари поєднують обидва підходи, дозволяючи організаціям використовувати переваги кожного з них. Наприклад, критично важливі дані можуть зберігатися в приватній хмарі, а менш важливі — у публічній. Це дозволяє збалансувати вимоги до безпеки та економії витрат.

Майбутнє хмарних обчислень виглядає надзвичайно перспективним. З появою нових технологій, таких як штучний інтелект, машинне навчання та Інтернет речей, хмара стає основним інструментом обробки великих обсягів даних. Крім того, зростає популярність edge computing — підходу, який обробляє ближче до джерел, щозменшує затримки та покращує продуктивність. Наприклад, edge computing може використовуватися для обробки даних з IoT-пристроїв у реальному часі, що особливо важливо для таких галузей, як промисловість чи транспорт.

Однак для повноцінного впровадження хмарних обчислень необхідно подолати низку викликів, зокрема в галузі безпеки, конфіденційності та стандартизації. Безпека даних залишається одним із головних пріоритетів, оскільки будь-який інцидент з безпекою може призвести до серйозних фінансових втрат, підриву довіри клієнтів та репутаційних ризиків.

Крім того, необхідно розробити стандарти, які дозволять забезпечити сумісність між різними хмарними платформами та полегшать міграцію даних.

Хмарні обчислення сьогодні є рушійною силою цифрової трансформації, що дозволяє організаціям будь-якого масштабу швидко адаптуватися до змін умов ринку, підвищувати ефективність бізнес-процесів та впроваджувати інновації. Завдяки гнучкості та масштабованості хмарних платформ, компанії можуть легко збільшувати або зменшувати обчислювальні ресурси залежно від поточних потреб, що дозволяє оптимізувати витрати на IT.

Однак для того, щоб хмара стала повністю надійним інструментом, необхідно продовжувати розвивати технології, вдосконалювати стандарти та вирішувати проблеми, пов’язані з безпекою та конфіденційністю. Лише тоді хмарні обчислення зможуть реалізувати свій повний потенціал, відкриваючи нові горизонти для інновацій та розвитку.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Опис хмарних сервісів

Хмарні сервіси — це набір технологій і послуг, що надаються через Інтернет. Завдяки ним користувач отримує доступ до обчислювальних потужностей, зберігання даних, програмного забезпечення. Для цього немає необхідності володіти фізичними пристроями та інфраструктурою. Ключовою особливістю є те, що через мережу Інтернет здійснюється доступ до хмарних ресурсів, що в свою чергу дозволяє зменшити витрати на закупівлю та утримання власних серверів, а також забезпечує високу масштабованість і гнучкість.

Хмарні сервіси можуть бути представлені різними моделями, серед яких найпоширенішими є "IaaS", "PaaS" та "SaaS". Вони відрізняються між собою тим, що мають на меті надання різних рівнів обчислювальних потужностей або програмних рішень через Інтернет, що дозволяє підприємствам та організаціям знижувати витрати на апаратні ресурси та програмне забезпечення (Рисунок 1.1).

Рисунок 1.1 – Порівняння «IaaS», «PaaS» та «SaaS»

Завдяки "IaaS", що розшифровується як «Інфраструктура як послуга» користувачі отримують доступ до віртуалізованих обчислювальних ресурсів, таких як мережі, сховища, а також сервери, на яких можна розгортати та управляти власними операційними системами та додатками. Одиними з найбільш відомих прикладів IaaS — це Google Cloud, Amazon Web Services та Microsoft Azure.

В свою чергу "PaaS" — це модель, що забезпечує платформу для тестування, імплементації програмних продуктів або створення нового продукту. Ваховуючи апаратні ресурси, вона також охоплює інструменти для розробки, бази даних та інші елементи, потрібні для розробки додатків. Microsoft Azure App Services та Google App Engine є прикладами таких рішень.

А ось Програмне забезпечення як послуга "SaaS" пропонує надання доступу користувачам вже до готових продуктів через Інтернет. Власники програмного забезпечення забезпечують повний контроль над програмою та її оновленнями, а користувачам не потрібно турбуватися про установку, підтримку чи оновлення програм. Популярні приклади — Google Workspace, Microsoft 365, Dropbox.

Всі ці сервіси стають основою для сучасних підприємств та користувачів завдяки наступним перевагам.

1 Масштабованість: хмара дозволяє змінювати ресурси відповідно до потреб. Якщо бізнес розширюється або має пік навантаження, кількість необхідних обчислювальних потужностей можна збільшити без фізичних інвестицій у нове обладнання.

2 Економічність: хмарні сервіси дозволяють значно знизити витрати на придбання та обслуговування фізичних серверів. Це дозволяє компаніям економити на витратах на апаратне забезпечення, енергоспоживання та персонал, який займається обслуговуванням серверів.

3 Безпека: великий хмарний постачальник зазвичай інвестує значні кошти в безпеку, маючи на меті захистити дані своїх клієнтів. Це включає

використання шифрування даних, контроль доступу та постійний моніторинг.

4 Доступність і гнучкість: хмарні сервіси забезпечують доступ до даних та програм з будь-якого місця, де є Інтернет, що дозволяє користувачам працювати з даними. [1]

1.2 Існуючі моделі розгортання хмарних сервісів

Розгортання хмарних сервісів залежить від кількості користувачів, що мають доступ до системи, і ступеня контролю, який необхідний організаціям над інфраструктурою.

1.2.1 Публічна хмара

Публічна хмара є найпоширенішим варіантом хмарного середовища, де постачальник хмарних послуг надає свої ресурси (сервери, сховища, бази даних тощо) для використання кількома клієнтами одночасно. Публічні хмари доступні через інтернет і не обмежуються лише окремими підприємствами чи користувачами. Всі апаратні ресурси належать постачальнику хмарних послуг, і він повністю керує ними.

Основні переваги публічної хмари:

- легкий доступ до ресурсів без необхідності інвестувати в фізичне обладнання;
- зниження витрат на утримання та обслуговування інфраструктури;
- гнучкість у масштабуванні ресурсів залежно від потреб;
- приклади публічних хмар: Amazon Web Services, Google Cloud Platform, Microsoft Azure.

1.2.2 Приватна хмара

Приватна хмара є моделлю, при якій хмарна інфраструктура використовується виключно одним підприємством чи організацією. Вона може бути розгорнута як на власних серверах, так і на сторонніх інфраструктурах, але важливо, що ресурси доступні тільки для однієї організації.

Переваги приватної хмари:

- повний контроль над безпекою та конфіденційністю даних;
- можливість налаштування інфраструктури під нагальні потреби;
- вищий рівень персоналізації та інтеграції;
- приватна хмара підходить для організацій, що мають високі вимоги до безпеки або роботи з конфіденційними даними.

1.2.3 Гібридна хмара

Гібридна хмара поєднує елементи публічної та приватної хмар, дозволяючи зберігати частину даних та додатків у публічній хмарі, а інші — в приватному середовищі. Так можна оптимізувати витрати, залишаючи критично важливі дані в приватних хмарах, тоді як інші дані обробляються в публічних хмарах для зниження вартості. Переваги гібридної хмари:

- гнучкість у виборі ресурсів залежно від потреб;
- можливість інтегрувати локальну інфраструктуру з хмарними сервісами;
- підвищена безпека для критичних даних, зберігання яких вимагає високого рівня конфіденційності.

1.3 Основні властивості хмарних сервісів

Основні властивості хмарних сервісів визначають їх унікальні

можливості та переваги порівняно з традиційною локальною інфраструктурою. Ці характеристики дозволяють ефективно обробляти і зберігати дані, адаптувати обчислювальні потужності та забезпечувати безпеку інформації в сучасних умовах зростаючих кіберзагроз. Ключові властивості хмарних сервісів включають масштабованість, доступність, безпеку, гнучкість, економічність і еластичність.

1.3.1 Масштабованість

Масштабованість — це одна з найважливіших властивостей хмарних технологій, яка дозволяє збільшувати чи зменшувати обсяг використовуваних ресурсів залежно від потреб користувача або бізнесу. Хмарна інфраструктура дозволяє швидко розширювати обчислювальні потужності, зберігати великі обсяги даних та обробляти значну кількість запитів без необхідності закупівлі додаткового обладнання. Масштабованість є критично важливою для компаній, які мають сезонні або динамічні зміни навантаження, наприклад, для e-commerce платформ, які відчувають різке зростання навантаження під час великих розпродажів або свят.

Типи хмарних сервісів зазвичай поділяється на два основні типи:

- горизонтальна масштабованість — передбачає додавання нових серверів або обчислювальних одиниць до існуючої системи. Це дозволяє розподілити навантаження між кількома машинами та підвищити стійкість системи;
- вертикальна масштабованість — полягає в збільшенні потужності існуючих серверів шляхом додавання оперативної пам’яті, збільшення кількості ядер процесора або обсягу сховища. Вертикальна масштабованість менш гнучка, оскільки має фізичні обмеження, але вона може бути корисною для вирішення тимчасових завдань або у випадках, коли система не може використовувати горизонтальну масштабованість.

1.3.2 Доступність

Доступність — це здатність хмарного сервісу забезпечувати постійний і безперебійний доступ до даних та обчислювальних потужностей. Більшість провайдерів хмарних послуг надають високий рівень доступності через механізми резервування даних, розміщення їх на кількох серверах, а також впровадженням аварійних процедур на випадок відмови апаратного забезпечення. Цей показник вимірюється зазвичай у відсотках часу безперервної роботи.

Стратегії забезпечення доступності:

- резервування та реплікація даних: дані зберігаються на кількох серверах або навіть у кількох центрах обробки даних в різних регіонах. Це дозволяє забезпечити доступність навіть у разі відмови однієї або декількох фізичних одиниць;
- аварійне відновлення: хмарні провайдери впроваджують плани аварійного відновлення, щоб у разі серйозної аварії відновити працездатність сервісів у найкоротші терміни;
- моніторинг та автоматичне перемикання: системи постійно моніторяться, а у випадку проблеми автоматично перемикаються на резервний центр обробки даних або сервер, що забезпечує безперервність роботи. [6]

1.3.3 Безпека

Безпека є важливою властивістю хмарних сервісів, особливо з огляду на високий ризик кіберзагроз. Сучасні хмарні технології використовують численні механізми захисту даних, зокрема шифрування, автентифікацію користувачів, контроль доступу та багатофакторну автентифікацію. Хмарні провайдери інвестують значні кошти у безпеку, розробляючи та вдосконалюючи системи захисту від зовнішніх атак і внутрішніх загроз.

Методи забезпечення безпеки:

- шифрування даних: всі дані в хмарі зазвичай шифруються як під час передачі, так і під час зберігання, що унеможливлює їх прочитання без ключа шифрування;
- система контролю доступу: забезпечує обмеження доступу до даних лише для авторизованих користувачів і додатків;
- механізми захисту від DDoS атак: спеціалізовані програми моніторингу та фільтрації мережевого трафіку допомагають захистити сервери від навантажень, спричинених кібератаками;
- багатофакторна автентифікація: знижує ризик несанкціонованого доступу, вимагаючи від користувача надання додаткового коду, що надсилається на його мобільний пристрій або інший засіб автентифікації.

1.3.4 Гнучкість

Гнучкість хмарних технологій дозволяє швидко адаптувати ресурси та інфраструктуру до змінюваних умов та потреб бізнесу. Компанії можуть налаштовувати хмарні середовища відповідно до власних вимог, тестувати нові додатки, змінювати налаштування обчислювальних потужностей або сховищ, а також експериментувати з новими технологіями без необхідності значних інвестицій у фізичну інфраструктуру.

Ця властивість є особливо цінною для стартапів і компаній, які прагнуть швидко розширювати свою діяльність або масштабувати обчислювальні потужності, щоб не відставати від зростаючого попиту. [2]

1.4 Методи шифрування

Конфіденційність даних є однією з найважливіших проблем сучасних хмарних технологій, особливо для користувачів, які зберігають конфіденційну інформацію. Для її забезпечення активно застосовуються

методи автентифікації та контролю доступу. Основна проблема полягає в обмеженій довірі до провайдерів хмарних послуг, що посилюється ризиком внутрішніх загроз. Просте шифрування даних не завжди відповідає сучасним вимогам, таким як підтримка складних пошукових операцій або паралельної модифікації даних.

Гомоморфне шифрування є важливим напрямом у криптографії, оскільки виконує обчислення над зашифрованими даними без їхнього розшифрування. Перше повне рішення у цій сфері було запропоновано Дженрі, що стало значним проривом. Проте, висока обчислювальна складність та значні вимоги до ресурсів суттєво обмежують практичне використання цього підходу.

Для підвищення ефективності були розроблені гібридні системи, що поєднують кілька методів шифрування, таких як RSA і 3DES. Ці системи використовують переваги кожного алгоритму: RSA забезпечує надійну автентифікацію, тоді як 3DES (рисунок 1.2) ефективний для шифрування великих обсягів даних. Такий підхід дозволяє зменшити ризики, пов’язані з управлінням ключами, і підвищити загальну безпеку. [7]

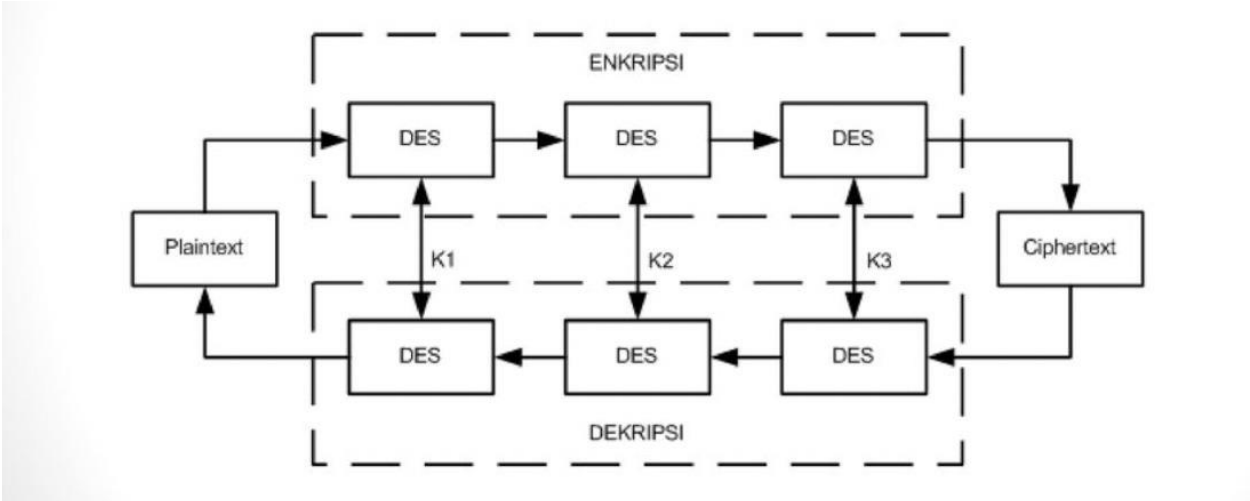

Рисунок 1.2 – Шифрування 3DES

Для покращення продуктивності у хмарному середовищі

застосовуються методи часткового гомоморфного шифрування. Наприклад, алгоритм TSFS дозволяє спрощувати шифрування баз даних, хоча зростання кількості ключів може збільшувати обчислювальні витрати.

Однією з перспективних технік є асиметричне шифрування баз даних, яке забезпечує додатковий рівень безпеки через повторне шифрування. Комутативне шифрування, запропоноване Хуаном і Цзо, додає ще більше гнучкості, оскільки дозволяє обробляти дані незалежно від порядку ключів. Подібні техніки особливо корисні для реалізації безпечних пошукових систем у хмарних середовищах. [8]

Іншою інновацією є ранжований пошук за кількома ключовими словами, що дозволяє користувачам ефективно виконувати запити до зашифрованих даних. Використання криптографічних геш-функцій додає захист від компрометації даних, зменшуючи ризики перехоплення трафіку.

Розподілене зберігання є перспективним підходом для забезпечення надійності даних у хмарному середовищі. Методика Шаміра дозволяє ділити дані на фрагменти, кожен з яких захищений окремим ключем. Це значно ускладнює несанкціонований доступ. Крім того, технологія "безпека як послуга" забезпечує багаторівневий захист шляхом шифрування та розподілу даних у різних хмарних сховищах.

Гібридні підходи поєднують кілька механізмів для забезпечення безпеки даних. Наприклад, трирівневий підхід включає автентифікацію користувача, шифрування даних і прискорене дешифрування. Використання RSA для обміну ключами у поєднанні з іншими алгоритмами дозволяє значно підвищити ефективність та гнучкість системи.

Для підвищення конфіденційності також використовуються методи приховування, що включають змішування реальних даних із фальшивими. Це дозволяє ускладнити аналіз даних навіть у випадку їхнього перехоплення. [3]

2 ПРИНЦИПИ ПОБУДОВИ ЗАХИСНОГО КОМПЛЕКСУ ДЛЯ ХМАРНИХ БАЗ ДАНИХ

Розвиток хмарних обчислень спричинив зміни в багатьох галузях, де зростання кількості даних і потреба у високій обчислювальній потужності зробили хмарні технології одним із найзатребуваніших рішень сучасності. Проте, разом з цими перевагами виникають і серйозні виклики в сфері безпеки, що обумовлені розподіленою природою хмарного середовища та відсутністю повного контролю користувача над інфраструктурою. У цьому розділі ми детально розглянемо основні принципи побудови комплексної системи захисту для хмарних баз даних, що включають багаторівневий підхід, використання технологій віртуалізації, протидію сучасним кіберзагрозам і інтеграцію різних компонентів захисної інфраструктури. Дотримання цих принципів забезпечує високу стійкість хмарних баз даних до потенційних атак і загроз, гарантує надійний захист конфіденційної інформації та знижує ризики, пов'язані з витоком даних.

2.1 Багаторівнева система захисту

Забезпечення безпеки даних у хмарі вимагає впровадження багаторівневих механізмів, де кожен рівень захищає різні аспекти інфраструктури та відповідає за конкретний тип загроз. Цей підхід базується на концепції "глибокого захисту", який дозволяє захистити базу даних від різних типів атак і мінімізувати можливі наслідки у випадку порушення одного з рівнів безпеки. Багаторівневий захист передбачає наявність декількох бар'єрів між зловмисником і конфіденційними даними, що підвищує загальну надійність системи.

Перший і фундаментальний рівень у цій системі — це шифрування даних. Воно дозволяє запобігти несанкціонованому доступу до даних навіть у разі

фізичного викрадення носія. Шифрування дозволяє захищати дані як у процесі їх передавання, так і при зберіганні, що критично важливо для хмарних баз даних. У контексті хмарних технологій часто використовують асиметричне шифрування "RSA" та симетричне шифрування "AES" (рисунок 2.1), що є стандартами безпеки для великих обчислювальних середовищ. Застосування таких технологій підвищує стійкість системи до атак на канали передачі даних і забезпечує захист у разі втрати фізичного носія, наприклад, серверного диска.

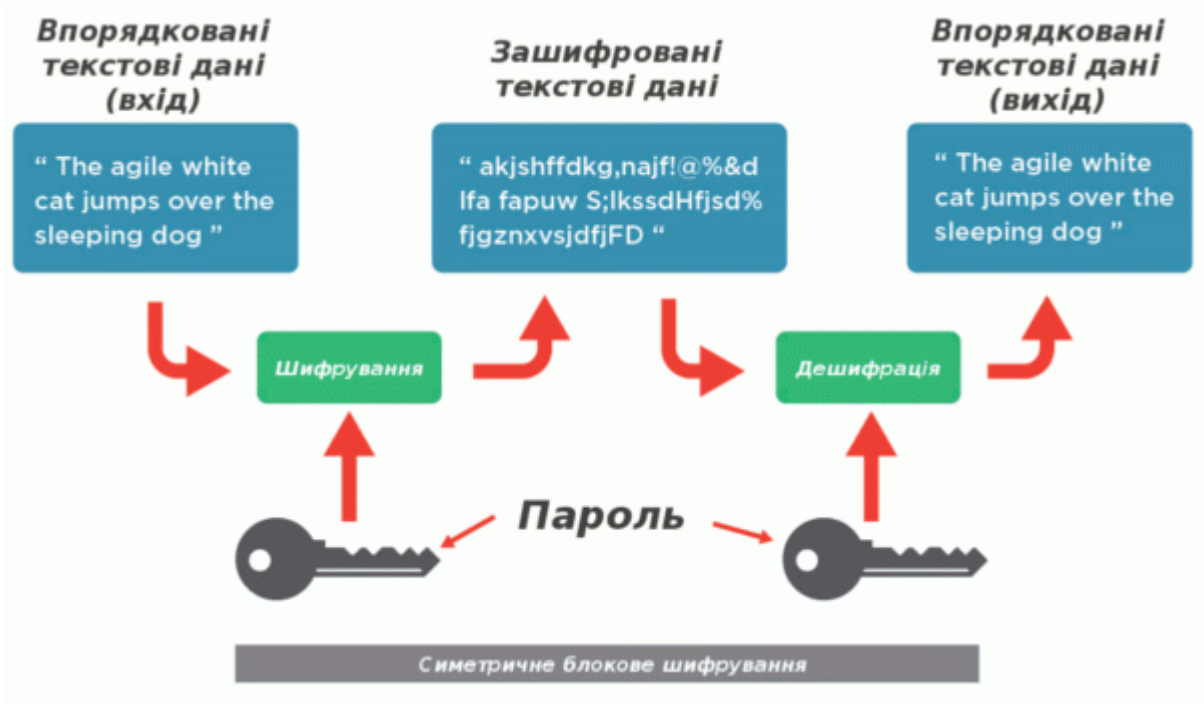

Рисунок 2.1 – Алгоритм AES

На другому рівні багаторівневої системи знаходиться контроль доступу до даних. Система має забезпечувати надійну аутентифікацію користувачів, а також точний розподіл прав доступу на основі ролей. Це означає, що користувачі можуть отримувати доступ лише до тих даних, які є для них релевантними, що знижує ризик витоку інформації. В сучасних хмарних середовищах впровадження багатофакторної аутентифікації стає стандартом, оскільки цей підхід дозволяє захистити облікові записи навіть у разі

компрометації пароля. Політики доступу можуть налаштовуватись з використанням моделі RBAC, що дозволяє встановлювати правила на рівні конкретних ролей і прив'язувати права доступу до функціональних обов'язків користувачів, мінімізуючи ймовірність зловживань.

Третій рівень багаторівневої захисної системи охоплює моніторинг і аналіз подій. У хмарних середовищах важливим завданням є своєчасне виявлення підозрілих дій та запобігання можливим інцидентам. Системи моніторингу на основі штучного інтелекту та машинного навчання можуть автоматично аналізувати великий обсяг подій та сигналів, що дозволяє швидко виявляти аномальну поведінку. Для цього застосовуються платформи SIEM, які дозволяють зібрати й обробити інформацію з різних джерел, таких як журнали подій, логи сервісів і мережевий трафік, для швидкого виявлення та реагування на можливі атаки. Інфографіка: загальна архітектура SIEM-систем.

Таким чином, багаторівнева система захисту хмарних баз даних забезпечує комплексний підхід до безпеки, включаючи захист на рівні фізичного зберігання даних, доступу до інформації та активного моніторингу подій, що відбуваються в системі. Це дозволяє значно знизити ризики та мінімізувати наслідки кіберінцидентів. [4]

2.2 Технології ізоляції та віртуалізації

Застосування технологій віртуалізації та ізоляції є одним з ключових принципів побудови захисної архітектури для хмарних баз даних. Віртуалізація дозволяє забезпечити ізоляцію робочих процесів у межах однієї фізичної інфраструктури. Це не лише забезпечує ефективне використання ресурсів, але й підвищує безпеку системи, оскільки у випадку компрометації однієї віртуальної машини інші залишаються недоторканими.

У випадку хмарних баз даних часто використовуються контейнери, які ізолюють програмне середовище на рівні операційної системи. Кожен

контейнер може містити окремі додатки, сервіси або частини бази даних, що дозволяє локалізувати ризики і зменшити ймовірність поширення шкідливого впливу. Віртуалізація дозволяє також налаштовувати мережеву ізоляцію і налаштування, що забезпечує контроль над доступом до контейнерів і захист від мережевих загроз. Окрім контейнерної віртуалізації, популярністю користуються технології віртуальних машин, які дозволяють створювати незалежні середовища з окремими операційними системами для кожного робочого процесу.

Додатково важливою є роль технології "Trusted Execution Environment" (Рисунок 2.2), що представляє собою фізичне середовище, захищене від зовнішніх впливів і призначене для обробки конфіденційних даних. TEE дозволяє обробляти чутливі операції в ізольованому середовищі, навіть якщо інші частини інфраструктури можуть бути скомпрометовані. Це особливо актуально для обробки фінансових даних, медичних записів і іншої інформації, яка потребує високого рівня захисту. [9]

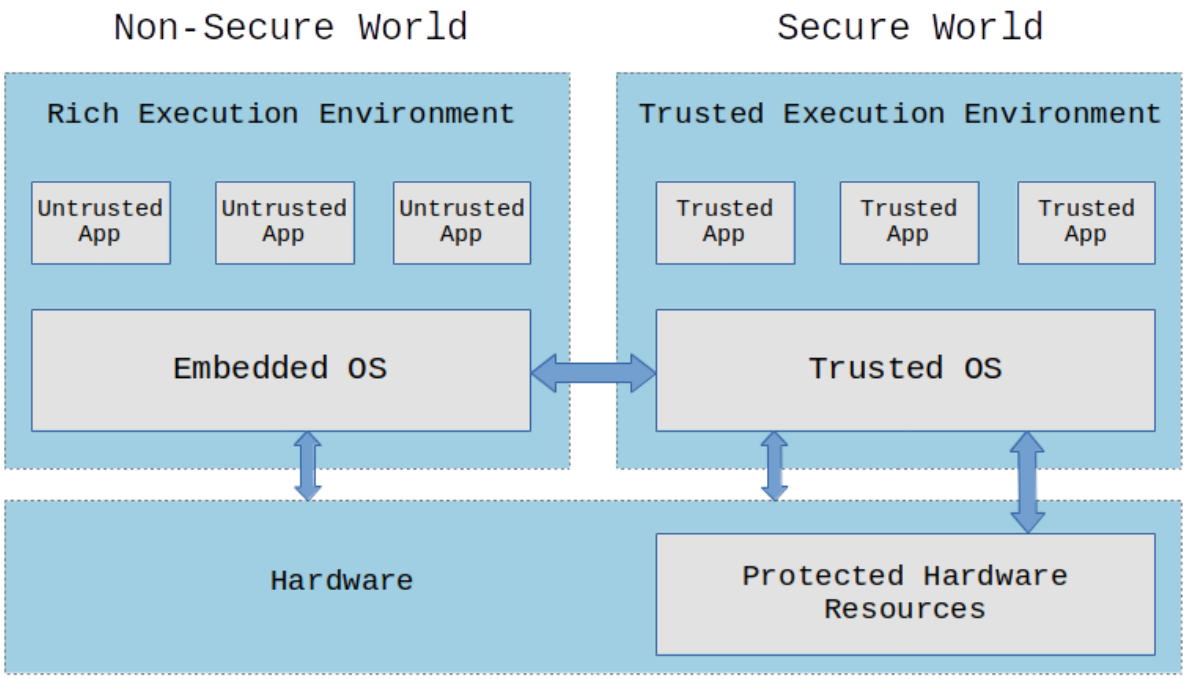

Рисунок 2.2 – Алгоритм TEE

2.3 Протидія сучасним загрозам і кібернападам

Сьогоднішні хмарні бази даних стикаються з безпрецедентним рівнем загроз, які включають DDoS-атаки, спроби проникнення за допомогою зловмисного "ПЗ", атаки на основі соціальної інженерії та інші сучасні типи кіберзагроз. Для захисту від таких атак у хмарних системах застосовується комплексний підхід, що включає в себе різноманітні методи виявлення та запобігання.

Для запобігання DDoS-атакам, які можуть суттєво впливати на доступність хмарної бази даних, використовуються масштабовані рішення для захисту мережевих ресурсів. Серед них — спеціалізовані фільтри та інтелектуальні системи для аналізу мережевого трафіку, які можуть виявити аномальні активності, пов’язані з атаками типу DDoS, і миттєво блокувати їх. Відповідні системи моніторингу та попередження допомагають своєчасно реагувати на загрози і забезпечують сталість хмарних сервісів.

Іншим важливим напрямком захисту є застосування машинного навчання для виявлення підозрілих шаблонів у поведінці користувачів. Штучний інтелект може виявляти потенційно зловмисні дії на основі аналізу поведінкових даних та аномальних активностей, таких як різкі зміни в характері запитів або спроби доступу з незвичних місць чи пристроїв. Моделі на основі машинного навчання дозволяють не тільки підвищити точність виявлення загроз, але й адаптуватися до нових типів атак.

2.4 Інтеграція компонентів захисту у хмарну інфраструктуру

Для забезпечення єдності та злагодженості компонентів безпеки у хмарних середовищах необхідна інтеграція систем, що дозволяє автоматизувати обробку подій та підвищити рівень захищеності. При впровадженні комплексної безпеки в хмарних базах даних ключовим елементом є розгортання систем централізованого управління. Такі системи

об’єднують інформацію з різних джерел, створюючи єдину панель управління для всіх аспектів безпеки, включаючи контроль доступу, моніторинг, аналіз подій та захист від вторгнень.

Для забезпечення ефективної інтеграції безпекових компонентів важливо враховувати особливості кожної системи, яка є частиною хмарної інфраструктури. Наприклад, системи управління ідентифікацією та доступом забезпечують надійну аутентифікацію та авторизацію, інтегруючись з хмарними сервісами для захисту облікових записів і контролю над доступом до даних. IAM-системи працюють у тісному поєднанні з системами контролю мережевих доступів, що дозволяє встановлювати різні рівні доступу для користувачів залежно від їхніх ролей і рівня довіри.

Крім того, для забезпечення безпеки під час роботи з великими масивами даних у хмарних середовищах необхідне застосування технологій контейнерної оркестрації, таких як Kubernetes. Це дозволяє більш ефективно управляти розподіленими середовищами та ізолює робочих процесів, контролю доступу і моніторингу. Контейнерні системи дозволяють розподіляти компоненти на кілька рівнів, зменшуючи ризики компрометації та підвищуючи загальний рівень безпеки. Системи на базі Kubernetes дозволяють організувати централізоване управління всіма контейнерними середовищами, що забезпечує злагодженість роботи і підвищує контроль над окремими компонентами системи.

Інтеграція засобів аналізу подій і виявлення загроз із загальною системою захисту дозволяє отримати оперативну інформацію про потенційні вразливості. Системи управління інцидентами забезпечують швидкий відгук на загрози, що дозволяє зменшити ризики і мінімізувати шкоду від атак, реагуючи на них у режимі реального часу. Платформи для реагування на інциденти на основі штучного інтелекту здатні визначати ймовірні точки входу загроз, автоматично відстежуючи дії вразливих користувачів, змінюючи політики доступу або ізолюючи потенційно небезпечні компоненти інфраструктури.

Завдяки інтеграції цих компонентів можна забезпечити єдину систему безпеки, що дозволить хмарній базі даних функціонувати з урахуванням сучасних загроз, знижуючи ризик витоку даних і підвищуючи загальний рівень захисту інформації. [5]

3 РЕАЛІЗАЦІЯ КОМПЛЕКСУ ТЕХНОЛОГІЙ ДЛЯ ЗАБЕЗПЕЧЕННЯ БЕЗПЕКИ ХМАРНИХ БАЗ ДАНИХ

Традиційно, загальні технології захисту конфіденційності поділяються на три основні категорії: конфіденційність за допомогою політики, конфіденційність за допомогою статистичного аналізу, конфіденційність за допомогою криптографії, але наше дослідження виявило, що різні архітектурні моделі слід розділити на дві основні області, тобто криптографічні та некриптографічні підходи.

3.1 Заходи конфіденційності, орієнтовані на криптографію

Хмарне зберігання даних зберігає інформацію, як у сейфі, де хмарне сховище виконує роль контролю доступу до цього сейфа. Насправді, дані зазвичай зашифровані, але часто хмарний сервер має ключ для дешифрування, який управляє правами доступу для кожного користувача. Це є критичною проблемою у випадку з конфіденційними даними, такими як адміністративні документи (наприклад, рахунки, платіжні листи або посвідчення особи), або, в більш загальному випадку, персональні дані. Це стає особливо складним у випадку конфіденційних документів, які належать бізнесу, тобто спільно використовуваних між співробітниками або з торговими партнерами. Насправді, цю проблему можна легко вирішити, просто зашифрувавши дані перед відправленням до хмарного сховища. Для вирішення цієї проблеми було запропоновано різні архітектури, які працюють на основі двох політик: оцінки довіри та предикатного шифрування.

3.1.1 Оцінка довіри/Авторизоване шифрування

Ця політика вимагає побудови довіри користувачів до оцінки надійних

систем, зокрема відповідно до загальноприйнятих критеріїв. Щодо довіри користувачів, контракт, що стосується використання системи управління ключами, повинен вказувати юрисдикцію, закони якої стосуються цієї системи.

3.1.2 Використання деталізованого управління правами

Для деталізованого управління правами використовується передовий криптографічний інструмент, відомий як схема "перешифрування через проксі". На основі цієї схеми було модифіковано підхід, який дозволяє клієнтам динамічно керувати своїми спільними документами у деревоподібній структурі. Пізніше було представлено реалізацію такої системи, яка включає використання смартфонів для завантаження, завантаження та обміну документами клієнтів. Система зосереджена на деталізованому управлінні правами, тобто на забезпеченні та обміні правами доступу на основі пріоритету користувача.

Основна проблема полягає в деталізованому управлінні правами. Зазвичай стандартна схема перешифрування через проксі має властивість "все або нічого". Якщо ключ перешифрування генерується клієнтом A, то проксі може перешифрувати для клієнта B будь-який документ, який спочатку був зашифрований для клієнта A. Однак клієнт A не може обмежити, що саме проксі може перешифрувати, окрім як довіряючи йому. У такому випадку, якщо простір для зберігання структурований у вигляді дерева, клієнт A може бажати поділитися лише конкретною папкою F_x або конкретними файлами $f_{x,\gamma}$, але не всіма файлами. Цю проблему можна вирішити за допомогою умовної схеми перешифрування. У цій схемі кожен завантажений файл має унікальну умову, яка визначається під час процесу шифрування. Для отримання зашифрованого тексту використовується публічний ключ клієнта A (pk_a), і результат шифрування буде мати вигляд:

$$C = Encrypt(pk_a, data, condition) \quad (3.1)$$

У іншому випадку, якщо клієнт A бажає надати свої права клієнту B для папки F_2, то ключ перешифрування обчислюється від A до B за умови, що він пов’язаний із F_2. Цей ключ позначається як $rk_a \rightarrow_\beta \hat{} F_2$ і надсилається проксі, що дозволяє здійснювати вертикальне перетворення між користувачами.

Третій випадок потребує більшої уваги, оскільки необхідно забезпечити певний шлях для повернення до кореня від конкретного файлу. Для вирішення цієї проблеми після вертикального перетворення додається горизонтальне перетворення всередині дерева. Воно використовує додаткові ключі перешифрування, такі як $rk\_F_1 \rightarrow F_2$ або $rk\_F_2 \rightarrow F_3$, які є модифікованими зашифрованими текстами з прив’язаними умовами (рисунок 3.1).

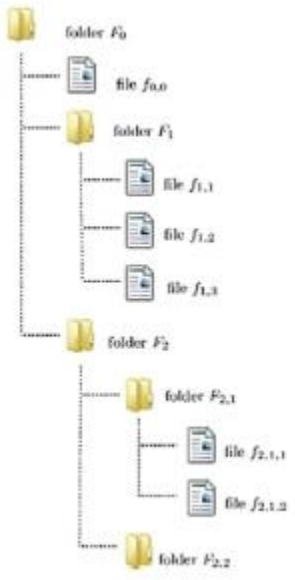

Рисунок 3.1 - Деревоподібна структура для повторного шифрування ключів

Ці кроки базуються на простій ідеї: для кожної пари або (папка, папка) на шляху від файлу до кореня клієнт A повинен обчислити ключ

перешифрування. Перешифрування через проксі виявилося ефективним інструментом, який успішно забезпечує конфіденційність, дозволяючи використовувати ненадійні платформи для зберігання чутливих документів. Цю схему можна адаптувати для інтеграції додаткових функцій, таких як дедуплікація, індексація або складні обчислення над зашифрованими даними.

3.1.3 Використання віддаленого аудиту даних (RDA)

Техніка RDA (Remote Data Auditing) належить до категорії криптографічних методів, оскільки забезпечує ймовірнісну або детерміновану гарантію цілісності даних. Вона включає такі властивості:

Ефективність: аудит даних із мінімально можливою обчислювальною складністю.

Публічна перевірка: можливість делегувати процес аудиту довіреній стороні (TPA), що зменшує обчислювальне навантаження на клієнта.

Ймовірність виявлення: можливість виявлення потенційних пошкоджень даних.

Для підтримки конфіденційності необхідно вирішити критичну проблему, пов’язану з аудитом даних. Наприклад, коли цифровий сертифікат у системі PKI закінчується, потрібно оновити ключі для хмарних користувачів та автентифікаторів. Yannan та інші запропонували механізм оновлення автентифікаторів та ключів із збереженням конфіденційності файлів за допомогою нульового розголошення. Цей механізм поєднує системи нульового розголошення, гомоморфні лінійні автентифікатори та перепідпис проксі. Він складається з п’яти алгоритмів: KeyGen, AuthGen, KeyUpdate, AuthUpdate та інтерактивної системи доказу між доводячим (Prover) і перевіряючим (Verifier).

KeyGen(1^k): приймає параметр безпеки k і повертає загальний рядок crs, який є неявним входом для всіх інших алгоритмів.

AuthGen(crs): генерує публічний ключ pk та секретний ключ sk для

користувача. Користувач публікує pk і зберігає sk у секреті.

AuthGen(sk, F): приймає секретний ключ sk та файл F, повертає набір автентифікаторів $\{D_i\}$ для файлу та параметр перевірки φ.

KeyUpdate(pk_{l-1}, sk_{l-1}): приймає старі ключі (pk_{l-1}, sk_{l-1}) і повертає нові ключі (pk_l, sk_l).

AuthUpdate(pk_l, sk_l, ft_{l-1}, φ): приймає нові ключі (pk_l, sk_l), старий тег файлу ft_{l-1} та параметр φ, повертає новий тег файлу ft_l та ключ оновлення β_l.

Доказ (P, V): Інтерактивний протокол між доводячим (P) і перевіряючим (V). Вхідні дані включають публічний ключ pk та параметр φ. P також має файл F та автентифікатори $\{D_i\}$. На кінці протоколу V повертає 1 або 0, що вказує на цілісність файлу. Позначення $\langle P, V \rangle = 1$ означає, що V приймає доказ.

Для схеми публічного аудиту зі збереженням конфіденційності важливими вимогами є:

Повнота: якщо дані не змінені, протокол завжди повертає 1.

Конфіденційність: TPA не отримує інформації про вміст файлу, окрім публічно доступних даних, таких як випадкове ім’я файлу.

Цей підхід підтверджено на основі оцінки властивостей, і його безпека є ефективною для практичного використання.

Цей інтерактивний протокол між доводячим (P) та перевіряючим (V) приймає спільні вхідні дані (P, V), які включають публічний ключ pk та параметр перевірки φ. P також має додаткові вхідні дані: файл $F = (m_1, m_2, ..., m_n)$ та набір автентифікаторів $\{D_i\}$ для цього файлу. На кінці протоколу V повертає біт 1 або 0, що вказує на те, чи збережений файл незмінним. Для зручності позначення використовується $\langle P, V \rangle = 1$, щоб показати, що V повертає 1 після взаємодії з P. Параметри (pk, φ) опускаються, коли контекст зрозумілий.

Для схеми публічного аудиту зі збереженням конфіденційності важливими вимогами є надійність, повнота та конфіденційність даних. Повнота означає, що якщо дані не змінені, інтерактивний протокол завжди

повертає ⟨P, V⟩ = 1, коли хмарний сервер та TPA дотримуються протоколу чесно. Концепція нульового розголошення гарантує, що TPA не отримує інформації про вміст файлу, окрім публічно доступних даних, таких як випадкове ім’я файлу. Це підтверджено на основі оцінки властивостей, і безпека схеми, включаючи надійність, є ефективною та придатною для практичного використання.

3.1.4 Використання BP-XOR Gates

Коди LT, LDPC та техніки цифрових фонтанів привернули значну увагу як з боку промисловості, так і з боку академічних досліджень у минулі роки. BP-XOR gates використовуються як ефективний підхід для оцінки довіри в галузі криптографії.

Щоб використати основні ідеї компетентного процесу декодування belief propagation (BP) у кодах LDPC та LT, у статті запропоновано BP-XOR коди та використано їх для проектування трьох класів схем розподілу секретів: псевдо-BP-XOR схеми розподілу секретів, LDPC схеми розподілу секретів та BP-XOR схеми розподілу секретів. Шляхом встановлення еквівалентності між моделлю реберно-забарвлених графів та BP-XOR схемами розподілу секретів другого ступеня автори розробили нові та ідеальні 2-з-n BP-XOR схеми розподілу секретів.

Використовуючи методи проектування масивних кодів, також можна розробити інші (n, k) порогові LDPC схеми розподілу секретів. У ефективних LDPC або BP-XOR схемах розподілу секретів, які були побудовані, для фази відновлення секрету та фази розподілу секрету потрібна лише лінійна кількість операцій XOR (виключне АБО) над бінарними рядками.

Для порівняння, схеми розподілу секретів Шаміра потребують $O(n^2)$ операцій у полі для фази відновлення секрету та $O(n \log n)$ операцій у полі для фази розподілу секрету. Крім того, автор стверджує, що такі схеми досягають оптимальної складності оновлення для схем розподілу секретів.

Таблиця 3.1 - Порівняння підходів до оцінювання на основі довіри

Архітектура	Canard	Yannan Li	Wang
Підхід	Використання сучасних криптографічних інструментів (напр., "проксі повторне шифрування")	Формалізація моделі конфіденційності з нульовим розголошенням для аудиту з оновленням ключів	Використання ідей декодування на основі поширення вірувань (BP) у кодах LDPC та LT
Концепція реалізації	Середовище обміну даними через модифікацію парадигми повторного шифрування	Використання однонаправленого повторного підписування	Розробка кодів BP-XOR для схем спільного доступу (BP-XOR, псевдо-BP-XOR, LDPC)
Основні області проблем	Динамічне управління деревовидною структурою для спільного доступу до документів	Доведення надійності та конфіденційності з нульовим розголошенням	Оптимальна складність оновлення
Переваги	Інтеграція додаткових функцій (індексація, видалення дублікатів)	Зменшення витрат на передачу та обчислення	Припущення, що сервери не змовляються
Недоліки	Потенційна повільність для великих файлів	Непередбачуваність для великих наборів даних	Складність запобігання атакам змови

Під складністю оновлення для схеми розподілу секрету мається на увазі середня кількість бітів у частках учасників, які потрібно оновити, коли змінюється певний біт головного секрету (таблиця 3.1).

3.1.5 Використання імовірнісних гібридних логік

У дослідженнях було поєднано базову гібридну логіку з кількісною логікою невизначеності за допомогою оператора задоволення. Ця техніка виділяється серед інших підходів завдяки своїм особливим характеристикам.

Логіка є досить виразною та гнучкою, щоб представляти багато існуючих критеріїв конфіденційності, таких як k-анонімність, логічна безпека, l-різноманітність, t-близькість та δ-розкриття. Основний внесок цієї логіки полягає у двох аспектах. По-перше, уніфікованість фреймворку пояснює загальний принцип, що стоїть за різними вимогами до конфіденційності, та підкреслює їхні відмінності. Наприклад, різниця між синтаксичними та семантичними критеріями конфіденційності легко спостерігається за допомогою логічних специфікацій. По-друге, універсальність фреймворку розширює сферу застосування специфікацій конфіденційності. Зокрема, можна вказати різні вимоги для різних осіб, що дозволяє досягти персоналізованої специфікації конфіденційності. Наприклад, можна використовувати щоб виразити різні вимоги до конфіденційності для осіб ii та jj. Крім того, логіка дозволяє довільні комбінації існуючих вимог до конфіденційності, що дає змогу виражати складні критерії конфіденційності. Наприклад, можна використовувати , щоб виразити, що для особи ii потрібні як логічна безпека, так і kk-анонімність.

Оскільки несподівані атаки можуть виникати час від часу, існуючі критерії можуть бути недостатніми; отже, може знадобитися вказати нові критерії. Наприклад, критерій логічної безпеки можна поєднати з δδ-розкриттям, щоб вимагати, щоб формули в Sec(i), а не просто ff-атоми, задовольняли критерію конфіденційності δδ-розкриття.

Таблиця 3.2 - Порівняння ймовірнісних підходів до оцінювання

Категорія	Нац. et al.	Papadimitriou	Zhang
Підхід	Логіка дозволяє довільні комбінації існуючих критеріїв конфіденційності	Виявлення витоків даних за допомогою ймовірності	Генерація запитів шуму для маніпулювання стратегіями запису шкідливих провайдерів послуг
Концепція реалізації	Розробка ймовірнісної гібридної логіки для специфікації вимог конфіденційності даних	Запропоновано стратегії розподілу даних (серед агентів), що покращують ймовірність виявлення витоків	HPNGS генерує запити шуму на основі історичної ймовірності
Галузь проблематики	Полегшує більш ефективний компроміс між захистом конфіденційності та корисністю даних	Не залежить від змін переданих даних (наприклад, водяні знаки)	Для ефективного захисту клієнтів кількість запитів шуму повинна залишатися мінімальною

Продовження таблиці 3.2

Категорія	Нац. et al.	Papadimitriou	Zhang
Переваги	Логіка є виразною та гнучкою для представлення багатьох існуючих критеріїв конфіденційності	Під час великого перекриття даних розподіл об’єктів може суттєво впливати на виявлення винних агентів	Підхід може значно зменшити кількість запитів шуму у порівнянні з випадковими запитами, більш ніж на 90%
Недоліки	Можна розробити більш надійний підхід, оскільки концепція є недостатньо виразною	Відкриває багато можливостей для вирішення різних проблем, які вимагають розширення цього підходу	Середовище враховує індивідуальні шкідливі провайдери, але не працює в паралельному режимі

Крім того, можна враховувати вагу секрету, щоб вимірювати серйозність його розкриття. Таким чином, визначається як функція ваги для кожної особи та секрету. Потім можна поєднати вагу з існуючими критеріями конфіденційності, щоб отримати нові моделі захисту конфіденційності. Це може сприяти більш ефективному балансу між захистом конфіденційності та корисністю даних. Логічна мова забезпечує уніфікований фреймворк для задоволення потреб специфікації як нових, так і існуючих критеріїв (таблиця 3.2) для порівняння різних подібних підходів, запропонованих раніше.

3.1.6 Кластеризації даних зі збереженням корисності

У цій роботі запропоновано алгоритм анонімізації, заснований на кластеризації та стійкий до атак на основі схожості та ймовірнісного виведення. Анонімізовані дані розподіляються на розподіленій файловій системі Hadoop. Цей метод досягає кращого балансу між конфіденційністю та корисністю даних. У цій роботі корисність даних вимірюється за допомогою точності та F-міри щодо різних класифікаторів.

Автори запропонували алгоритм кластеризації для досягнення анонімізованих кластерів, кожен з яких має рівномірний розподіл чутливих значень. Алгоритм кластеризації визначає найкращого сусіда для кожного кластера та додає один екземпляр за раз до існуючого кластера. Автори модифікували алгоритм, щоб подолати перекос у розподілі чутливих значень у результуючих кластерах, використовуючи техніку K-найближчих сусідів (KNN). Алгоритм KNN-(G,S) кластеризації визначає KN найближчих сусідів для кожної групи чутливих значень за допомогою наступного рівняння та додає KN записів до кластерів одночасно.

$$KN = \frac{|\ Di\ |}{NOB} \tag{3.2}$$

де $|\ D_i\ |$ — кількість екземплярів у кожній підгрупі чутливих значень, а NOB — кількість кластерів. Ця формула розподіляє записи рівномірно між усіма кластерами. Таким чином, утворені кластери матимуть рівномірний розподіл чутливих значень у вихідному наборі даних. Вихідний набір даних DD сортується за значенням чутливого атрибуту SA*SA*. Після сортування вхідний набір даних ділиться на підгрупи $D_1, D_2, \ldots, D_n$. Кожна підгрупа міститиме однакові значення для S A. Усі інші підгрупи, крім $D_{\min}$ (тобто підгрупи з найменшою кількістю екземплярів), вважаються D_{rem}. На основі значень k та S алгоритм працює у двох випадках.

У випадку 1 значення k менше або дорівнює S ($k \leq S$), а у випадку 2 значення k більше за S ($k > S$). Автори стверджують, що цей алгоритм також зменшує складність порівняно з іншими подібними підходами.

Проте, проведено аналіз найгіршого випадку обчислювальної вартості алгоритму KNN-(G, S) кластеризації, який представлено за допомогою великого O-нотації. З іншого боку, найгірший випадок вартості зберігання запропонованого алгоритму задається як O(n). Вихідний набір даних ділиться на підгрупи та зберігається окремо, що призводить до додаткової вартості зберігання, яка показана як O(n).

Вартість зберігання кластерів, утворених у випадку 1 або 2 алгоритму, також задається як O(n). Загальний розмір усіх кластерів дорівнює розміру вихідного набору даних. Загальна вартість зберігання алгоритму задається як S(n)$S(n)$. Після спрощення вартість зберігання алгоритму KNN-(G,S) кластеризації становить $O(n)$.

Однією з найкращих особливостей цього підходу є те, що він легко подолає можливість ймовірнісного виведення атаки. Ця можливість порівнюється з іншими техніками, заснованими на t-близькості.

Існуючі техніки анонімізації даних не показують такого порівняння. Зокрема, більшість існуючих технік оцінюють продуктивність лише за допомогою традиційних метрик. На відміну від них, у запропонованому підході втрата інформації вимірюється за допомогою традиційних метрик, таких як глобальний показник впевненості, метрика неоднорідної ентропії, нормалізована втрата інформації, нормалізований показник впевненості, помилка запиту та сума квадратів помилок. Однак у запропонованому підході автори досягли успіху в анонімізації за рахунок заміни квазіідентифікаторів (QID) на основі центроїдів, що є обчислювально ефективнішим за придушення та менш витратним за узагальнення (таблиця 3.3).

Таблиця 3.3 - Порівняння підходів до оцінювання на основі анонімізації

Архітектурний дизайн	Nayani et al.	Sreenivasa Rao	Yang et al.
Підхід	Розробка алгоритму анонімізації, заснованого на кластеризації, стійкого до зтик на освоїсть та ймовірнісного виведення	Пропозиція дозаово безпечної схеми CP-ABSC для системи стійкого використання PFR у хмарі	Порядче рішення для збереження конфденційності при обміні даними у хмарному середовищі
Концепція реалізації	Анонімізовані дані розподіляється на розподіленій файловій системі Hadoop	Конструкція передбачає малий розмір шиброваного тексту та вимагає менше обчислювальних пар	Різні методи комбінуються для підтримки багатьох парадигм медичних даних із різними рівнями конфденційності
Галузь проблематики	Корисність даних виміряється за точністю та метрикою F щодо різних класифікаторів	Схема забезпечує точний контроль доступу, конфденційність, автентичність, перевірку підпису та загальну верифікацію	Підтримує багато парадигм доступу до даних із різними рівнями конфденційності

Продовження таблиці 3.3

Архітектурний дизайн	Nayani et al.	Sreenivasa Rao	Yang et al.
Переваги	Метод досягне кращого балансу між конфденційністю та корисністю	Схема експлуатує монотонні будеві функції та реалізує безпеку в стандартній моделі	Показує, що синергія технологій, які зберігають конфденційність, може забезпечити кращий баланс між використанням інформації та захистом конфденційності
Недоліки	Відсутні порівняльні результати для підтвердження переваг щодо інших методів	Відсутні недоліки	Потрібно визначити, як продуктивність впливає на доступність для багатьох клієнтів, які одночасно користуються хмарними послугами

Однак, підходи, засновані на вищій корисності даних, коли вони використовуються лише в приватних хмарах, зберігають конфіденційність на бажаному рівні, оскільки інфраструктура в таких випадках користується довірою. Тому для систем, що працюють у публічних або гібридних хмарах, використання достатньо сильної ідеї збереження конфіденційності є дуже важливим, що можна зобразити в таблиці (таблиця 3.4).

Таблиця 3.4 - Відображення корисності та складності інформації на основі характеру різних видів доступу до приватності

Категорія	Ймовірнісний та ранжувальний підхід з використанням автентифікованих структур даних	Криптографічна політика приватності з визначеним рівнем доступності даних	Анонімізація та захист даних з можливістю зв’язку та обмеженим доступом до даних
Доступ до приватності	Ймовірнісний та ранжувальний підхід з використанням автентифікованих структур даних	Криптографічна політика приватності з визначеним рівнем доступності даних	Анонімізація та захист даних з можливістю зв’язку та обмеженим доступом до даних
Корисність інформації	Висока	Низька	Середня
Складність	Низька	Висока	Висока
Підхід	Орієнтований на політику з низьким рівнем захисту приватності	Орієнтований на користувача з високим рівнем захисту приватності	Політика з обмеженим доступом до інформації та можливістю зв’язку
Переваги	Зменшення накладних витрат на комунікацію, полегшення реалізації та обчислень	Забезпечує захист на рівні записів, запобігає підслуховуванню в різних середовищах даних	Вирішує проблеми аномалій політик, зменшує накладні витрати на комунікацію та обчислення

Продовження таблиці 3.4

Категорія	Ймовірнісний та ранжувальний підхід з використанням автентифікованих структур даних	Криптографічна політика приватності з визначеним рівнем доступності даних	Анонімізація та захист даних з можливістю зв’язку та обмеженим доступом до даних
Недоліки	Схильний до несанкціонованого доступу	Теоретично можливий, використовує технології прямого індексування	Порівняно високий час обробки запитів через відхилення зайвих даних

Хмарні обчислення є перспективною та новітньою технологією для наступного покоління IT-додатків. Основною перешкодою для швидкого розвитку хмарних обчислень є питання безпеки та конфіденційності даних. Зменшення витрат на зберігання та обробку даних є обов’язковою вимогою для будь-якої організації, тоді як аналіз даних та інформації завжди залишається найважливішим завданням для прийняття рішень у всіх організаціях. Тому жодна організація не передаватиме свої дані або інформацію в хмару, доки не буде встановлено довіру між постачальниками хмарних послуг та споживачами. Дослідники запропонували ряд методів для захисту даних та досягнення найвищого рівня безпеки даних у хмарі. Однак існує ще багато пробілів, які потрібно заповнити, щоб зробити ці методи більш ефективними. Потрібно більше роботи у сфері хмарних обчислень, щоб зробити їх прийнятними для споживачів хмарних послуг. У цій роботі проведено огляд різних методів щодо безпеки та конфіденційності даних, зосереджуючись на зберіганні та використанні даних у хмарі, для захисту даних у середовищах хмарних обчислень з метою встановлення довіри між постачальниками хмарних послуг та споживачами.

4 РЕАЛІЗАЦІЯ ПРОГРАМИ

Програма, розроблена в рамках цього проекту, є високотехнологічним інструментом для забезпечення конфіденційності та аналізу даних. Основною метою цієї програми є продемонструвати два ключових аспекти — криптографічну безпеку через шифрування даних за допомогою алгоритму RSA та кластеризацію даних за допомогою алгоритму K-Means. Кожен з цих етапів виконується на високому рівні з урахуванням ефективності та безпеки.

4.1 RSA Шифрування

Перше, що варто зазначити, це важливість правильної генерації ключів RSA. Алгоритм RSA широко використовується у сучасних системах безпеки для забезпечення конфіденційності переданих даних. Основним принципом роботи RSA є використання двох ключів — публічного та приватного. Публічний ключ використовується для шифрування, тоді як приватний ключ необхідний для дешифрування (Рисунок 4.1).

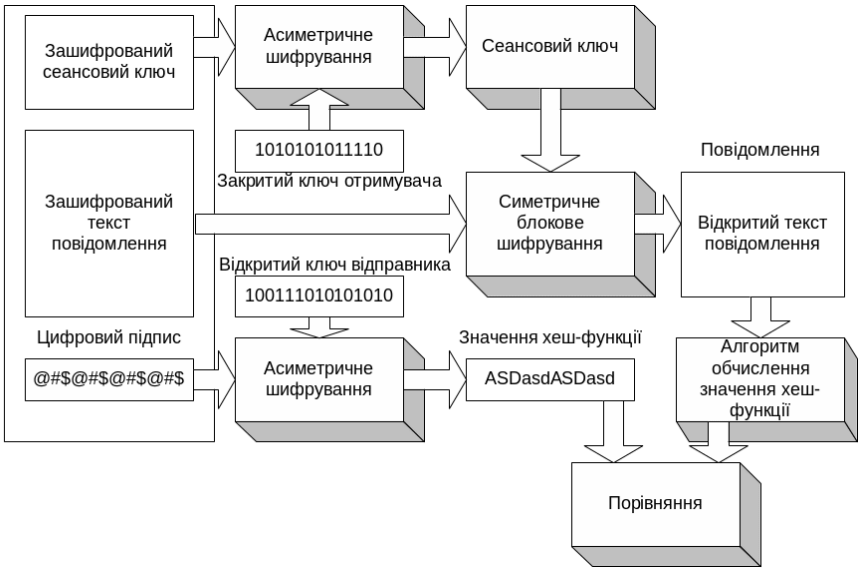

Рисунок 4.1 – Алгоритм шифрування/дешифрування rsa

Для реалізації цього алгоритму в програмі спочатку генеруються два простих числа p та q, які є основою для обчислення модуля $n = p \times q$, де n є частиною як публічного, так і приватного ключа. Крім того, на основі цих чисел обчислюється функція Ейлера $\phi(n) = (p-1)(q-1)$, яка є важливою для обчислення відкритого ключа e*e* та приватного ключа d*d*.

Алгоритм RSA використовує принцип взаємно простих чисел, що є необхідною умовою для вибору значення відкритого ключа e. Вибір e*e* проводиться таким чином, щоб воно було взаємно простим з функцією Ейлера $\phi(n)$. Тільки тоді обчислюється приватний ключ d, який є оберненим елементом до e*e* за модулем $\phi(n)$.

Після генерації ключів програма готова до шифрування та дешифрування даних. Наприклад, для шифрування повідомлення "Hello, World!" кожен символ цього повідомлення перетворюється на число за допомогою його ASCII-коду, після чого це число підноситься до степеня відкритого ключа e*e* за модулем n*n*. Зашифроване повідомлення представляється як набір чисел, що потім можна передавати через незахищений канал. Тільки володіючий приватним ключем зможе розшифрувати це повідомлення, використовуючи алгоритм дешифрування, що передбачає піднесення чисел до степеня приватного ключа *d* за модулем n.

Цей підхід гарантує високу ступінь безпеки, оскільки відновити початкові дані без знання приватного ключа є надзвичайно складним завданням. Таким чином, RSA ефективно захищає дані під час їх передачі. Програма демонструє, як можна поєднувати криптографічні методи з аналізом даних для забезпечення конфіденційності та виявлення закономірностей. Використання RSA забезпечує високий рівень захисту даних, а алгоритм K-Means дозволяє проводити ефективну кластеризацію. Це робить програму корисним інструментом для роботи з конфіденційною інформацією в різних галузях, таких як фінанси, медицина та наукові

дослідження.

4.2 Кластеризація за допомогою K-Means

Крім шифрування, програма також реалізує алгоритм K-Means для кластеризації даних. Алгоритм K-Means є одним з найпоширеніших методів для розподілу набору даних на класи або групи, де кожна точка в наборі даних належить до найближчого центроїда (середнього значення групи). Основна перевага цього алгоритму полягає в тому, що він ефективно обробляє великі набори даних, швидко розподіляючи їх на класи.

У рамках цього проекту використовуються випадкові дані — 100 точок, що генеруються в двовимірному просторі. Кількість кластерів, k, задається користувачем, у даному випадку вона становить 5. Алгоритм спочатку випадковим чином вибирає центри кластерів, а потім на кожній ітерації класифікує точки даних на основі відстані до центроїдів. Після цього центроїди оновлюються як середнє значення точок у відповідному кластері. Процес повторюється до тих пір, поки центроїди не перестануть змінюватися, що свідчить про досягнення стабільності кластеризації.

Цей алгоритм дозволяє знайти структуру в даних без попередньо заданих міток, що корисно для багатьох застосувань, таких як сегментація клієнтів у маркетингу або групування документів у тематичні категорії. Враховуючи, що цей метод є безнаглядним, він дуже корисний для великих наборів даних, де неможливо попередньо визначити, як саме будуть розподілені дані. Програма демонструє, як можна поєднувати криптографічні методи з алгоритмами машинного навчання для досягнення високого рівня конфіденційності та ефективного аналізу даних. Це робить її корисним інструментом для роботи з конфіденційною інформацією в різних сферах, забезпечуючи високий рівень захисту та ефективність обробки даних.

4.3 Загальний підхід та переваги

Обидва алгоритми — RSA та K-Means — використовують перевірені методи для забезпечення безпеки та аналізу даних. Програма показує, як ці методи можуть бути використані для досягнення двох важливих цілей:

- захист даних. Використання алгоритму RSA дозволяє шифрувати і дешифрувати дані, гарантуючи їх конфіденційність при передачі через незахищені канали зв'язку. Це особливо важливо для сучасних систем електронного банкінгу, онлайн-платежів та будь-яких інших областей, де передача чутливих даних є необхідною;
- аналіз даних. Алгоритм K-Means дозволяє проводити кластеризацію великих масивів даних, що корисно для різних сфер, таких як маркетингові дослідження, аналіз поведінки користувачів в Інтернеті та навіть медичні дослідження.

Таким чином, програма дозволяє вирішити два важливі завдання, кожне з яких є надзвичайно корисним у сучасному світі, де обробка і захист даних стають все більш критичними. Програма може бути використана для захисту та аналізу даних у хмарних середовищах, де конфіденційність та безпека є критично важливими. Вона дозволяє забезпечити захист даних під час їх передачі через незахищені канали, а також проводити аналіз даних без порушення їхньої конфіденційності. Це особливо важливо для організацій, які працюють з великими обсягами чутливої інформації, такою як медичні записи, фінансові звіти або особисті дані користувачів.

4.4 Детальний опис функцій у коді

Функція generate_prime() генерує випадкове просте число у діапазоні від 50 до 200. Для перевірки, чи є число простим, використовується функція is_prime(n), яка перевіряє, чи має число дільники, крім 1 і самого себе. Якщо число просте, воно повертається як pp або qq.

Функція $gcd(a,b)$ обчислює найбільший спільний дільник (НСД) двох чисел за допомогою алгоритму Евкліда. Ця функція використовується для перевірки, чи є e*e* взаємно простим з $\phi(n)$.

Функція modinv(a,m) обчислює обернений елемент до a*a* за модулем m*m* за допомогою розширеного алгоритму Евкліда. Ця функція використовується для обчислення приватного ключа d.

Функція encrypt(message,public_key) шифрує повідомлення, перетворюючи кожен символ у його ASCII-код і підносячи його до степеня e*e* за модулем n*n*. Результатом є набір чисел, які представляють зашифроване повідомлення.

Функція decrypt(encrypted_message, private_key) дешифрує повідомлення, підносячи кожне число до степеня d за модулем n*n* і перетворюючи його назад у символ. Результатом є оригінальне повідомлення.

Функція kmeans(data, k, max_iterations=100) виконує кластеризацію даних. Спочатку вибираються випадкові центроїди, після чого точки даних розподіляються на кластери на основі відстані до центроїдів. Після цього центроїди оновлюються як середнє значення точок у кластері. Процес повторюється до досягнення стабільності.

Функція euclidean_distance(a, b) обчислює евклідову відстань між двома точками, що використовується для визначення найближчого центроїда.

Функція mean(points) обчислює середнє значення для набору точок, що використовується для оновлення центроїдів.

4.5 Результати роботи програми

Програма успішно генерує ключі RSA, що є основою для забезпечення конфіденційності даних. Наприклад, були згенеровані два прості числа: p=83 та q=179. На їх основі обчислено модуль n=14857 та функцію

Ейлера $\phi(n)=14596$. Відкритий ключ $e=179$ був обраний таким чином, щоб бути взаємно простим з $\phi(n)$, а приватний ключ $d=1263$ обчислено як обернений елемент до e *e* за модулем $\phi(n)$. Ця пара ключів дозволяє забезпечити високий рівень захисту даних.

Для демонстрації роботи алгоритму RSA було зашифровано повідомлення "Hello, World!". Кожен символ повідомлення перетворено у його ASCII-код, після чого цей код піднесено до степеня $e=179$ за модулем $n=14857$. Результатом шифрування став набір чисел: [8306, 7798, 7805, 7805, 4228, 7562, 13636, 13154, 4228, 1188, 7805, 10661, 13995]. Цей зашифрований набір може бути переданий через незахищені канали зв’язку без ризику витоку інформації.

Дешифрування повідомлення відбувається за допомогою приватного ключа $d=1263$. Кожне зашифроване число підноситься до степеня d за модулем n*n*, після чого перетворюється назад у символ. У результаті отримано оригінальне повідомлення "Hello, World!", що підтверджує коректність роботи алгоритму RSA. Цей підхід гарантує високу ступінь безпеки, оскільки без знання приватного ключа розшифрування є практично неможливим.

Програма також успішно реалізує алгоритм K-Means для кластеризації даних. У рамках демонстрації було згенеровано 100 випадкових точок у двовимірному просторі, які потім розподілено на 5 кластерів. Процес кластеризації включав ітеративне оновлення центроїдів до досягнення стабільності. Наприклад, після 7 ітерацій алгоритм завершив роботу, розподіливши точки на кластери: Кластер 1 містить 27 точок, Кластер 2 — 21 точку, Кластер 3 — 17 точок, Кластер 4 — 18 точок, а Кластер 5 — 17 точок.

Цей результат демонструє ефективність алгоритму K-Means для виявлення структури в даних (рисунок). Кожен кластер представляє групу точок, які знаходяться близько одна до одної, що дозволяє виявляти закономірності в даних без необхідності попередньої мітки. Це особливо корисно для аналізу великих наборів даних, де неможливо заздалегідь

визначити, як саме будуть розподілені дані.

Програма демонструє високу ефективність у вирішенні поставлених завдань. Шифрування RSA забезпечує надійний захист даних під час їх передачі через незахищені канали, що є критично важливим для сучасних систем електронного банкінгу, онлайн-платежів та інших сфер, де передаються чутливі дані. Алгоритм K-Means дозволяє ефективно аналізувати дані, виявляючи закономірності та групи, що може бути використано для маркетингових досліджень, аналізу поведінки користувачів або медичних досліджень.

Крім того, програма є універсальним інструментом, який може бути легко адаптований для різних типів даних та інтегрований з хмарними сервісами. Це робить її корисним інструментом для бізнесу, досліджень та навчання. Наприклад, у фінансовій сфері програма може бути використана для захисту транзакцій або аналізу фінансових даних. У медичній сфері — для захисту медичних записів та кластеризації пацієнтів за діагнозами. У наукових дослідженнях — для аналізу експериментальних даних.

Таким чином, результати роботи програми підтверджують її ефективність у забезпеченні конфіденційності даних та їх аналізу. Вона є важливим інструментом для сучасних організацій, які працюють з великими обсягами чутливої інформації, та може бути використана для створення нових рішень у галузі захисту даних. Програма демонструє, як можна поєднувати криптографічні методи з алгоритмами машинного навчання для досягнення високого рівня безпеки та ефективності, що робить її важливим кроком у розвитку сучасних інформаційних технологій.

ВИСНОВКИ

У процесі роботи над проектом, спрямованим на створення комплексу технологій для забезпечення безпеки хмарних баз даних, було розроблено програму, яка демонструє ефективне поєднання криптографічних методів для забезпечення конфіденційності даних та алгоритмів машинного навчання для їх аналізу. Основна мета програми полягала у демонстрації двох ключових аспектів: шифрування даних за допомогою алгоритму RSA та кластеризації даних за допомогою алгоритму K-Means. Кожен з цих компонентів був реалізований з урахуванням сучасних вимог до безпеки та ефективності, що робить програму корисним інструментом для роботи з конфіденційною інформацією.

Алгоритм RSA, який був реалізований у програмі, є одним із найнадійніших методів шифрування з відкритим ключем. Він забезпечує високий рівень захисту даних за рахунок використання двох великих простих чисел pp та qq, які є основою для генерації ключів. Публічний ключ (e,n) використовується для шифрування даних, тоді як приватний ключ (d,n) — для їх дешифрування. Цей підхід гарантує, що навіть у разі перехоплення зашифрованого повідомлення без знання приватного ключа розшифрування є практично неможливим. У програмі шифрування здійснюється шляхом перетворення кожного символу повідомлення у його ASCII-код, після чого цей код підноситься до степеня ee за модулем nn. Результатом є набір чисел, які можуть бути передані через незахищені канали зв’язку. Дешифрування відбувається за допомогою приватного ключа, де кожне число підноситься до степеня dd за модулем nn, після чого перетворюється назад у символ. Цей процес забезпечує високу ступінь конфіденційності даних під час їх передачі.

Окрім шифрування, програма включає реалізацію алгоритму K-Means для кластеризації даних. Цей алгоритм дозволяє розподіляти дані на групи

(кластери) на основі їхньої схожості. У програмі генерується набір випадкових точок у двовимірному просторі, які потім розподіляються на кластери. Кожен кластер представляє групу точок, які знаходяться близько одна до одної. Цей метод дозволяє виявляти закономірності в даних без необхідності розкривати їхній вміст, що є особливо корисним для аналізу великих наборів даних. Алгоритм K-Means працює шляхом ітеративного оновлення центроїдів (центрів кластерів) до тих пір, поки вони не перестануть змінюватися. Це дозволяє ефективно розподіляти дані на групи, що може бути використано для різних застосувань, таких як сегментація клієнтів у маркетингу або групування документів у тематичні категорії.

Програма демонструє, як можна поєднувати криптографічні методи з аналізом даних для забезпечення конфіденційності та виявлення закономірностей. Використання RSA забезпечує високий рівень захисту даних, а алгоритм K-Means дозволяє проводити ефективну кластеризацію. Це робить програму корисним інструментом для роботи з конфіденційною інформацією в різних галузях, таких як фінанси, медицина та наукові дослідження. Програма може бути використана для захисту та аналізу даних у хмарних середовищах, де конфіденційність та безпека є критично важливими. Вона дозволяє забезпечити захист даних під час їх передачі через незахищені канали, а також проводити аналіз даних без порушення їхньої конфіденційності. Це особливо важливо для організацій, які працюють з великими обсягами чутливої інформації, такою як медичні записи, фінансові звіти або особисті дані користувачів.

Однією з ключових переваг програми є її універсальність. Вона може бути легко адаптована для різних типів даних та інтегрована з хмарними сервісами, що робить її корисним інструментом для бізнесу та досліджень. Наприклад, у фінансовій сфері програма може бути використана для захисту транзакцій або аналізу фінансових даних. У медичній сфері — для захисту медичних записів та кластеризації пацієнтів за діагнозами. У наукових дослідженнях — для аналізу експериментальних даних.

Крім того, програма може бути використана як навчальний інструмент для вивчення криптографії та методів аналізу даних. Вона демонструє, як можна поєднувати різні технології для досягнення високого рівня безпеки та ефективності. Це особливо важливо в умовах, коли обробка та захист даних стають все більш критичними для сучасних систем.

У результаті роботи над проектом було створено ефективний комплекс технологій, який забезпечує безпеку та аналіз даних. Програма демонструє, як можна поєднувати криптографічні методи з алгоритмами машинного навчання для досягнення високого рівня конфіденційності та ефективного аналізу даних. Це робить її корисним інструментом для роботи з конфіденційною інформацією в різних сферах, забезпечуючи високий рівень захисту та ефективність обробки даних. Програма є важливим кроком у напрямку створення надійних систем для захисту даних у хмарних середовищах, що є критично важливим для сучасних організацій.

Таким чином, програма не лише демонструє ефективність використання криптографічних методів та алгоритмів машинного навчання, але й відкриває нові можливості для їх застосування в реальних умовах. Вона є важливим інструментом для забезпечення безпеки та конфіденційності даних у сучасному світі, де обробка інформації стає все більш складною та вимогливою. Програма може бути використана для створення нових рішень у галузі захисту даних, що робить її важливим елементом у розвитку сучасних інформаційних технологій. [10]

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. ucloud [Електронний ресурс] – Режим доступу : www/ URL: https://ucloud.ua/hmarni-tehnologiyi-shho-cze-take/ – 25.10.2024 р. – Загол. з екрану.

2. EDIN [Електронний ресурс] – Режим доступу : www/ URL: https://edin.ua/shho-take-xmarni-texnologi%D1%97-i-navishho-voni-potribni/– 25.10.2024 р. – Загол. з екрану.

3. Microsoft [Електронний ресурс] – Режим доступу : www/ URL: https://www.microsoft.com/uk-ua/security/business/security-101/what-is-cloud-security - 25.10.2024 р. – Загол. з екрану.

4. conf [Електронний ресурс] – Режим доступу : www/ URL: https://conf.ztu.edu.ua/wp-content/uploads/2016/06/3.pdf - 25.10.2024 р. – Загол. з екрану.

5. Softico [Електронний ресурс] – Режим доступу : www/ URL: https://softico.ua/uk/news/top-hmarnih-zagroz-z-yakimi-neobhidno-borotisya-malim-ta-serednim-pidpriyemstvam/ - 25.10.2024 р. – Загол. з екрану.

6. dropbox [Електронний ресурс] – Режим доступу : www/ URL: https://experience.dropbox.com/uk-ua/resources/what-is-the-cloud - 25.10.2024 р. – Загол. з екрану.

7. Котяшичев И. А. Защита информации в «Облачных технологиях» как предмет национальной безопасности / И. А. Котяшичев, Е. А. Бырылова // Молодой ученый. — 2015. — №6.4. — С. 30-34..

8. The NIST Definition of Cloud Computing (англ.). [Електронний ресурс]: Режим доступа: http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800- 145.pdf

9. Документ України щодо обробки інформації в системах хмарних обчислень [Електронний ресурс] - Режим доступа : http://w1.c1.rada.gov.ua/pls/zweb2/webproc4_1?pf3511=58527

10. Попов А.О. Архітектура нульової довіри // Інформаційні технології та безпека : XXIV Міжнародна науково-практична конференція ІТБ-2024. - 19 грудня 2024. –с.107-110.