МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Західноукраїнський національний університет
Факультет комп'ютерних інформаційних технологій
Кафедра кібербезпеки

МЕЛЬНИК Павло

Метод шифрування даних в хмарних сервісах / Data Encryption Method in Cloud Service

спеціальність: 125 – Кібербезпека
освітньо-професійна програма – Кібербезпека

Кваліфікаційна робота

Виконав студент групи КБм -22
П. Мельник

Науковий керівник
к.т.н., доцент Н.Г. Яцків

Кваліфікаційну роботу допущено
до захисту:

«____» _____ ___2023 р.

Завідувач кафедри
_____ В.В.Яцків

ТЕРНОПІЛЬ – 2023

Факультет комп'ютерних інформаційних технологій
Кафедра кібербезпеки
Освітній ступінь «магістр»
спеціальність: 125 - Кібербезпека
освітньо-професійна програма –Кібербезпека

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ В.В.Яцків
«____» _____ 2023 року

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ
МЕЛЬНИКА Павла
(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи:
Метод шифрування даних в хмарних сервісах /
Data Encryption Method in Cloud Service
керівник роботи к.т.н., доцент Н.Г.Яцків
затверджені наказом по університету від «__» _____ 2022 року № _____
2. Строк подання студентом закінченої випускної кваліфікаційної роботи 1 грудня 2023 року.
3. Вихідні дані до кваліфікаційної роботи: завдання на випускну кваліфікаційну роботу студента, наукові статті, технічна література.
4. Основні питання, які потрібно розробити:
– провести аналіз категорій хмарних сховищ та їх інфраструктури;
– проаналізувати поширені технології шифрування даних;
– провести порівняльний аналіз алгоритмів шифрування даних в хмарних обчисленнях;
– провести порівняльний аналіз поширених хмарних сховищ;
– розгорнути приватне хмарне сховище та проаналізувати рівень шифрування даних в ньому.
5. Перелік графічного матеріалу у роботі:
– шифрування на основі ідентифікації;
– модель інтелектуальної системи обміну даних на основі хмарних обчислень;
– модель безпеки хмарних даних;
– архітектура шифрування з симетричним ключем;
– архітектура шифрування з асиметричним ключем;
– класифікація алгоритмів шифрування;
– підключення зовнішніх пристроїв;
– створення персонального ключа відновлення.

6. Консультанти розділів кваліфікаційної роботи

	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання 8 грудня 2022 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строки виконання етапів кваліфікаційної роботи	Примітка
1	Аналіз архітектури хмарної безпеки	12.2022 р. – 03.2023 р.	
2	Алгоритми шифрування даних в хмарних обчисленнях	03.2023 р. – 05.2023 р.	
3	Впровадження персонального хмарного сховища NextCloud	05.2023 р. – 11.2023 р.	

Студент _____ Павло МЕЛЬНИК
(підпис)

Керівник роботи _____ к.т.н., доцент Наталя ЯЦКІВ
(підпис)

АНОТАЦІЯ

Кваліфікаційна робота на тему «**Метод шифрування даних в хмарних сервісах**» на здобуття освітнього ступеня «Магістр» зі спеціальності 125 «Кібербезпека» освітньо-професійної програми «Кібербезпека» написана обсягом 64 сторінки і містить 27 рисунків, 2 таблиці та 35 джерел за переліком посилань.

Метою кваліфікаційної роботи є дослідження принципів проведення хмарних обчислень та алгоритмів шифрування даних в них.

Методи досліджень. Для розв’язання поставлених задач у даній кваліфікаційній роботі використано: методи аналізу контролю та обмеження доступу, процеси аутентифікації.

Результати дослідження: проведено дослідження алгоритмів шифрування даних в хмарних обчисленнях та порівняльний аналіз поширених хмарних сховищ.

Результати роботи можуть успішно застосовуватися при розгортанні приватних хмарних сховищ.

КЛЮЧОВІ СЛОВА: КІБЕРБЕЗПЕКА, ХМАРНІ ОБЧИСЛЕННЯ, ХМАРНІ СХОВИЩА, ЗАХИСТ ДАНИХ, ШИФРУВАННЯ ДАНИХ, ДЕШИФРУВАННЯ ДАНИХ, КОНТРОЛЬ ДОСТУПУ.

ABSTRACT

The graduate work on the topic «Data Encryption Method in Cloud Service» for Master's degree on speciality 125 "Cybersecurity" is written on 64 pages and contains 27 illustrations, 2 tables and 35 references.

The aim of graduate work is to investigate the principles of cloud computing and data encryption algorithms in them.

Research methods. To solve the tasks set in this qualification work, the methods of analysis, control, and access restriction, as well as authentication processes, have been used.

Results of the study. An analysis of data encryption algorithms in cloud computing was carried out, as well as a comparative analysis of popular cloud storage solutions.

The results of the work can be successfully applied in the deployment of private cloud storage.

KEYWORDS: CYBERSECURITY, CLOUD COMPUTING, CLOUD STORAGE, DATA PROTECTION, DATA ENCRYPTION, DATA DECRYPTION, ACCESS CONTROL.

.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

CC – Хмарні обчислення.

CSP – Постачальники хмарних послуг.

IoT – Інтернет речей.

SAN – Storage Area Networks.

NAS – Network Attached Storage.

SaaS – Програмне забезпечення як послуга.

PaaS – Платформа як послуга.

IaaS – Інфраструктура як послуга.

PKI – Інфраструктурі відкритих ключів.

PKG – Генератор приватних ключів.

LWE – Навчання з помилками.

API – Інтерфейси прикладного програмування.

RSA – Rivest Shamir Adelman.

AES – Advanced Encryption Standard.

PB – Відкритий ключ.

PK – Закритий ключі.

RB – Велике випадкове число.

CSS – Системи хмарного зберігання.

S-AES – Спрощений розширений стандарт шифрування.

IDEA – Міжнародний алгоритм шифрування даних.

ВСТУП

Технологія хмарних обчислень (CC) набула широкої популярності завдяки своїй здатності надавати величезні ресурси окремим особам і організаціям, до яких можна отримати доступ через Інтернет у будь-який час і в будь-якій точці світу [1, 2]. Багато інформаційних і технологічних (IT) компаній перенесли свою діяльність у хмару, яка надає своїм користувачам багатофункціональний хмарний досвід, включаючи доступ до спільних ресурсів, що робить ресурси доступними, коли вони потрібні, за меншими витратами. Ці ресурси також можуть бути швидко надані та звільнені з мінімальними адміністративними зусиллями, а CC надає можливість спільно використовувати, керувати та зберігати дані, які фактично розміщені на віддалених серверах, а не за допомогою внутрішніх ресурсів чи персональних пристроїв [1]. Клієнти можуть використовувати хмарні сервіси різних програм, прийнявши CC, а не купуючи або встановлюючи програмне забезпечення на своїх комп'ютерах [3]. CC надає клієнтам віртуалізовані ресурси за допомогою різних технологій, таких як веб-сервіси, віртуалізація, програми та операційні системи [1]. Основні переваги CC можна підсумувати як зниження витрат, підвищення продуктивності, стабільність, масштабованість, легке управління та доступність [4, 5].

Незважаючи на зазначені вище переваги CC, це породило різноманітні проблеми та виклики. Безпека є однією з найбільших перешкод, які перешкоджають прийняттю CC серед користувачів [1, 6]. Це серйозне занепокоєння, яке необхідно враховувати, і виникають проблеми з безпекою даних, оскільки клієнтські дані та програмне забезпечення знаходяться на території постачальника [7]. Таким чином, постачальники хмарних послуг (CSP) повинні захищати дані, програми та хмарну інфраструктуру від внутрішніх і зовнішніх загроз. Безпека хмарної інформації залежить від впровадження відповідних заходів безпеки інформації та контрзаходів, що робить створення та керування безпечним хмарним середовищем складною

операцією. Захист даних користувачів від зловмисних атак і ненадійних серверів є надзвичайно важливим.

Мета і завдання дослідження. Метою кваліфікаційної роботи є дослідження принципів проведення хмарних обчислень та алгоритмів шифрування даних в них.

Досягнення визначеної мети передбачає вирішення таких завдань:

– провести аналіз категорій хмарних сховищ та їх інфраструктури;

– проаналізувати поширені технології шифрування даних;

– провести порівняльний аналіз алгоритмів шифрування даних в хмарних обчисленнях;

– провести порівняльний аналіз поширених хмарних сховищ;

– розгорнути приватне хмарне сховище та проаналізувати рівень шифрування даних в ньому.

Об’єкт дослідження – процеси шифрування та дешифрування даних в хмарних сервісах.

Предмет дослідження – алгоритми та методи шифрування даних в хмарних сервісах.

Методи досліджень. Для розв’язання поставлених задач у даній кваліфікаційній роботі використано: методи аналізу контролю та обмеження доступу, процеси аутентифікації.

Наукова новизна одержаних результатів. Проведено дослідження алгоритмів шифрування даних в хмарних обчисленнях та порівняльний аналіз поширених хмарних сховищ.

Практичне значення отриманих результатів. Розгорнуто приватне хмарне сховище на базі Nextcloud.

РОЗДІЛ 1. АНАЛІЗ АРХІТЕКТУРИ ХМАРНОЇ БЕЗПЕКИ

1.1. Аналіз хмарних середовищ

З розвитком Інтернету речей (IoT) кількість пристроїв сприйняття інформації, підключених до Інтернету, зростає, щоб реалізувати взаємозв'язок між людьми, пристроями та «речами».

Згідно з новим прогнозом IDC [1], у 2025 році буде 41,6 мільярда пристроїв або «речей» Інтернету речей, які генеруватимуть 79,4 зетабайта (ZB) даних. Мало того, люди все ще прагнуть покращити ефективність збору даних пристроїв в IoT [2]. Безпрецедентна кількість даних генерується та розміщується на платформі провайдера хмарних послуг [3]. Завдяки високій продуктивності, масштабованим і надійним хмарним центрам обробки даних багато додатків і служб розумного міста будуть розміщені в хмарі. Таким чином, мешканці розумного міста та постачальники послуг можуть покладатися на хмарні послуги для розміщення, створення та/або розгортання своїх служб і програм розумного міста [4]. Крім того, перевага оплати за використання спонукає більшість традиційних підприємств активно переносити дані в хмару.

Хмара – це не тільки місце призначення робочого навантаження, але й забезпечує ефективну практику роботи, що робить підприємства більш гнучкими та гнучкими. Це сприяло як цифровій трансформації підприємств, так і трансформації модернізації мережі [5]. У звіті ООН про цифрову економіку за 2019 рік підкреслюється, що цифрова економіка стає важливою рушійною силою економічного розвитку. Згідно з неповною статистикою, на цифрову економіку припадає від 4,5% до 15,5% світового ВВП [6]. Хмарні обчислення сприяють глибокій інтеграції Інтернету, великих даних, штучного інтелекту та реальної економіки, а також є основою прискорення побудови сучасної економічної системи. За даними Gartner Inc. [7], світовий ринок публічних хмарних послуг зросте на 17% у 2020 році, досягнувши 266,4 мільярдів доларів

США, порівняно з 227,8 мільярдів доларів США у 2019 році. Загалом, хмарні програми все ще є основними.

Хмарне сховище – це, по суті, система хмарних обчислень, яка дозволяє користувачам зберігати та обмінюватися даними в Інтернеті. Переваги хмарного сховища включають необмежений простір для зберігання даних, зручний, безпечний і ефективний доступ до файлів і резервне копіювання за межами сайту, а також низьку вартість використання.

У практичних додатках хмарне сховище можна розділити на п’ять категорій, а саме: публічне хмарне сховище, персональне хмарне сховище, приватне хмарне сховище, гібридне хмарне сховище та спільнотне хмарне сховище.

У **загальнодоступній хмарі підприємства** передають бізнес зі зберігання даних постачальникам хмарних сховищ (наприклад, AWS і Alibaba Cloud) без необхідності розгортання інфраструктури та обслуговування серверів. Доступ до даних має лише авторизований користувач.

Переваги **публічної хмари**, такі як гнучкість, масштабованість і економія коштів, приваблюють велику кількість малих і середніх підприємств.

Персональна хмара, також відома як мобільне хмарне сховище, по суті є гілкою загальнодоступної хмари, але відрізняється від публічної хмари тим, що вона надає послуги загальнодоступного хмарного сховища для окремих користувачів.

У **приватній хмарі підприємствам** необхідно розгорнути інфраструктуру хмарних сховищ і залучити професійний персонал для керування та обслуговування серверів. Це гарантує, що приватна хмара має більш високий рівень безпеки, ніж публічна хмара, і контроль над даними знаходиться в руках самого підприємства. Але вартість різко зростає. Ця модель зберігання більше підходить для великих підприємств з великою кількістю дорогих і конфіденційних даних.

Гібридна хмара — це поєднання публічної хмари та приватної хмари, яка успадковує всі переваги обох. Підприємства можуть зберігати дорогі та

конфіденційні дані у приватній хмарі, а інші дані – у публічній хмарі. Привабливість цієї моделі зберігання продовжує зростати.

Як новий режим хмарного зберігання в останні роки хмара спільноти дуже підходить для медичної та фінансової промисловості. Хмара спільноти надає хмарні послуги для кількох компаній у певній спільноті. Зазвичай ці підприємства мають однакові проблеми або потребують спільної роботи над деякими проектами. Побудова інфраструктури та керування сервером можуть спільно здійснюватися членами спільноти Cloud або передаватися третій стороні.

З точки зору архітектури зберігання, основні хмарні платформи зазвичай пропонують три широкі класи сховища: блочне сховище, сховище файлів і сховище об’єктів [4].

- Хмарне блочне сховище, яке поважає Storage Area Networks (SAN), по суті, забезпечує віртуалізовану мережу Storage Area Network із забезпеченням керування логічним томом через спрощений інтерфейс веб-служб.
- 2) Зберігання файлів, яке також називають сховищем на рівні файлу або сховищем на основі файлів, зазвичай асоціюється з технологією мережевого сховища (NAS) [4].

Завдяки файловій системі файлове сховище керує даними спільного використання та доступом до даних, що зберігаються в ньому, більш гнучко, ніж блокове сховище. Масові дані створюють низку проблем для підприємств, таких як розширення сховища, спільне використання даних, ефективна передача, вартість і безпека даних, коли зберігання даних досягає рівня PB, обмеження NAS і SAN безпосередньо призводить до збільшення вартості обслуговування обладнання. в більш пізній період. Вони не можуть повністю задовольнити вимоги підприємства щодо надійності, доступності, безпеки та інших показників масового зберігання даних у цьому сховищі об’єкта.

Хмарне сховище базується на інфраструктурі віртуалізації та схоже на хмарні обчислення з точки зору доступних інтерфейсів, масштабованості та ресурсів вимірювання.

Воно складається з чотирьох рівнів [11], які можна подати таким чином:

1) Рівень зберігання, основна частина хмарного сховища, складається з пристроїв зберігання та уніфікованого керування пристроями зберігання даних.

2) Первинний рівень керування є основною частиною хмарного сховища, а також найскладнішою частиною хмарного сховища.

3) Рівень інтерфейсу програми є найбільш гнучкою частиною хмарного сховища.

4) Останнім є рівень доступу.

З цієї точки зору хмарне сховище надає послуги доступу до даних, включаючи зберігання даних, обчислення даних, автентифікацію та контроль доступу. Через особливості хмарного сховища в цьому процесі неминуче виникають проблеми з безпекою даних і конфіденційністю.

Вимоги безпеки даних у хмарних сховищах в основному відображені в наступних аспектах [8]:

• Конфіденційність даних стосується запобігання активним атакам неавторизованих сторін на дані користувачів і забезпечення повної відповідності інформації, отриманої одержувачем даних, з інформацією, надісланою відправником. Це означає, що лише уповноважені особи мають право доступу та отримання даних. Уявіть свій банківський рахунок. Ви, звичайно, повинні мати до них доступ, і працівники банку, які допомагають вам з транзакцією, повинні мати до них доступ, але ніхто інший не повинен. Після доступу інших осіб конфіденційність даних порушується, що є незворотнім.

• Цілісність даних — це надійність даних, тобто дані не можуть бути довільно підроблені та замінені. Наприклад, якщо ви робите покупки онлайн на Amazon, хтось може змінити товари у вашому кошику без вашого дозволу. Відсутність цілісності даних може створити серйозні проблеми з безпекою.

• Доступність даних підкреслює, що до них можна отримати звичайний доступ у будь-який час, а саме: користувач може отримувати доступ, завантажувати або вносити деякі зміни в дані в хмарі, як тільки їм це буде потрібно.

• Точний контроль доступу.

• Безпечний обмін даними в динамічній групі.

• Стійкий до протікання.

• Повне видалення даних. Коли користувачі більше не користуються хмарним сховищем, вони можуть повністю видалити дані, передані на хмарний сервер, і підтвердити, що дані були повністю знищені, замість того, щоб бути ошуканими зловмисними постачальниками хмарних послуг.

• Захист конфіденційності. Хоча користувачі насолоджуються зручністю хмарного сховища, постачальники хмарного сховища зберігають їхню конфіденційну інформацію, таку як особисті дані, місцезнаходження та конфіденційні дані для підприємства. Механізми захисту конфіденційності використовуються, щоб гарантувати, що ці дані залишаються секретними від цікавих противників і зловмисних співробітників постачальників хмарних послуг.

З подальшою централізацією даних і збільшенням їх обсягу захист даних у хмарному сховищі стає проблематичним. Таким чином, питання про те, як гарантувати, що користувачі та їхні інформаційні ресурси не будуть розкриті, ще довгий час буде головною проблемою постачальників хмарних послуг і науковців.

Однак існуючі методи інформаційної безпеки більше не відповідають вимогам інформаційної безпеки в епоху великих даних, і загрози безпеці поступово стануть вузьким місцем, що обмежуватиме розвиток технології великих даних. Насправді безпека зберігання даних включає статичну безпеку даних і динамічну безпеку даних у хмарному сховищі.

Статична безпека даних – це забезпечення безпеки статичних даних у хмарній системі зберігання, тоді як безпека динамічного зберігання — це

забезпечення цілісності та конфіденційності під час передачі даних. Дані передаються через IP-мережу в хмарному сховищі, тому загрози безпеці в традиційній мережі також існують у хмарній системі зберігання, такі як знищення даних, крадіжка даних, підробка даних, відмова в обслуговуванні тощо, що впливає на безпечне зберігання даних.

У хмарній системі зберігання дані користувачів можуть бути розподілені між декількома серверами, і кожен сервер може використовуватися кількома користувачами, що призводить до зростання ризику небажаного несанкціонованого доступу. Складні алгоритми шифрування не є дружніми до користувачів з обмеженими ресурсами, тому переконатися, що вони можуть працювати на власних пристроях, є практичною проблемою. Крім того, має бути висока ймовірність того, що пристрої користувачів будуть піддані атаці побічного каналу.

Отже, безпека даних і збереження конфіденційності в системі хмарного зберігання в основному стикаються з такими проблемами:

- точне керування доступом до даних;
- зловмисні постачальники хмарних послуг можуть повертати неправильні дані;
- результати аудиту доброчесності;
- атака бічного каналу;
- зловмисні постачальники хмарних послуг не дотримуються запитів клієнтів щодо повного видалення даних у хмара;
- збереження конфіденційності.

Хоча хмарне сховище розвивалося протягом багатьох років, воно все ще дуже важливо в Інтернеті речей, розумному місті та цифровій економіці. Безпека даних і захист конфіденційності в хмарних сховищах все ще мають велике значення.

1.2. Принципи та моделі хмарних обчислень

СС – це метафора для опису Інтернету як місця, де обчислювальна техніка була попередньо встановлена та доступна як послуга, де дані, програми, операційні системи, сховище та потужність обробки доступні в Інтернеті та готові до спільного використання між клієнтами [2]. СС відноситься до набору центрів обробки даних, які підключаються до Інтернету, щоб пропонувати свої послуги, і ці центри обробки даних базуються на віртуалізації своєї інфраструктури [10]. СС технологічно базується на інфраструктурі, програмній платформі, операційній системі, розробці хмарних програм, управлінні базами даних, програмному забезпеченні для керування системою та програмами, Інтернеті та мережі [2]. Постачальники послуг СС – це компанії, які надають своїм клієнтам ресурси та послуги СС, які використовуються динамічно за запитом клієнта відповідно до конкретної бізнес-моделі. На рисунку 1.1 показано зв'язок між найпоширенішими постачальниками послуг СС, такими як Amazon, Google, Microsoft і IBM.

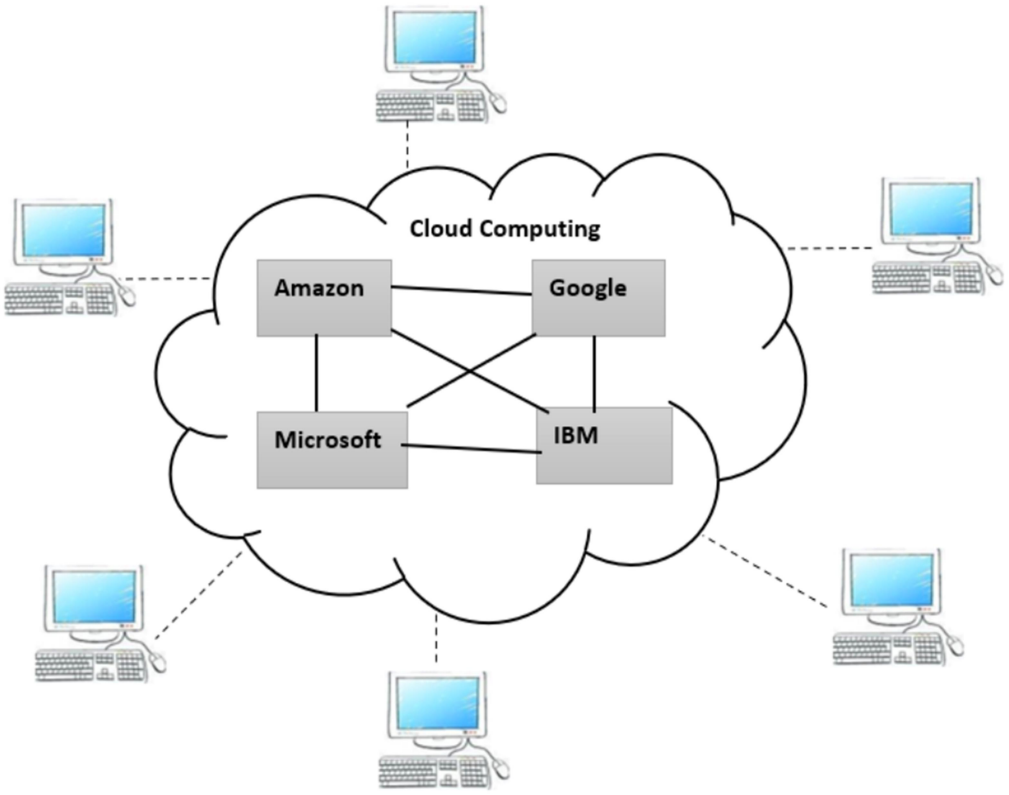

Рисунок 1.1 – Хмарні обчислення

СС класифікується на три типи [9]: приватна, публічна та гібридна хмара. Приватні хмари управляються та контролюються лише для окремої організації, а активи не використовуються іншими клієнтами, що вказує на те, що вони захищені від доступу неавторизованих користувачів. Публічні хмари доступні для широкої громадськості та організацій. Активи розподіляються між кожним із клієнтів. Клієнти платять власнику хмари залежно від наданої послуги та активів, які вони використовують. CSP керують фізичною інфраструктурою, яка розташована подалі від клієнтів. Гібридні хмари є сумішшю двох вищевказаних типів (публічного та приватного) [10].

СС надає три ключові послуги, а саме:

- програмне забезпечення як послуга (SaaS);
- платформа як послуга (PaaS);
- інфраструктура як послуга (IaaS) [11].

IaaS відноситься до апаратної інфраструктури CSP, яка включає мережі, сховище, пам’ять, процесори та низку інших обчислювальних ресурсів. Ресурси надаються як віртуалізовані системи, до яких можна отримати доступ через Інтернет. Основні ресурси знаходяться під контролем CSP [1].

PaaS забезпечує інтегроване середовище розробки, проміжне програмне забезпечення, операційні системи та ресурси рівня платформи через стороннього постачальника, який надає апаратні та програмні засоби користувачам через Інтернет. PaaS не надає клієнтам контроль над основною хмарною інфраструктурою, а лише над програмами, які переміщуються в хмару.

SaaS дозволяє споживачам використовувати програми як послугу через Інтернет. Користувачі можуть просто використовувати Інтернет для доступу до нього, а не купувати, встановлювати та підтримувати програмне забезпечення. Клієнти платять за використання, а не за право власності на програмне забезпечення.

Система СС розділена на дві частини: передню та задню частину, які спілкуються один з одним через мережу, зазвичай Інтернет. Передня частина -

це сторона, яку бачать хмарні клієнти. Клієнти зазвичай не бачать внутрішнього розділу, який включає мережеве підключення, хмарні сервери та їхні програми.На рисунку 1.2 показано категорії хмарних сервісів і архітектуру.

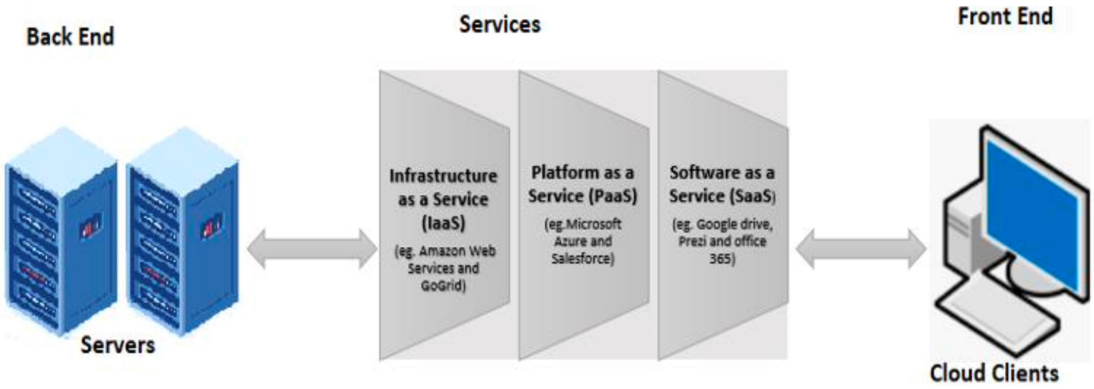

Рисунок 1.2 – Сервісна модель CC

Хмарні середовища задовольняють багатьом характеристикам, які [5]:

− самообслуговування на вимогу – хмарні служби (такі як мережеве сховище, доступ до мережі та безперервний моніторинг часу роботи сервера) не потребують жодних менеджерів. Самі клієнти можуть надавати, контролювати та маніпулювати обчислювальними ресурсами та IT-послугами за потреби.

− Об’єднання ресурсів – CSP може розподіляти витрати та ресурси CC (такі як сервери, сховище, база даних, програми, мережі та служби) серед великого пулу користувачів, що дозволяє користувачам, підключеним до хмари, використовувати дані одночасно та спільно використовувати хмару послуги відповідно до їхніх вимог.

− Широкий доступ до мережі – користувач може отримати доступ до ресурсів CC через мережу з будь-якої точки світу за допомогою підключення до Інтернету та пристрою (наприклад, смартфона, комп’ютера та КПК).

− Швидка еластичність – обчислювальні послуги та ресурси можна швидко та гнучко збільшувати чи зменшувати за потреби.

− Економія – CC зменшує величезні витрати на IT для своїх користувачів.

На рисунку 1.3 наведено характеристики CC.

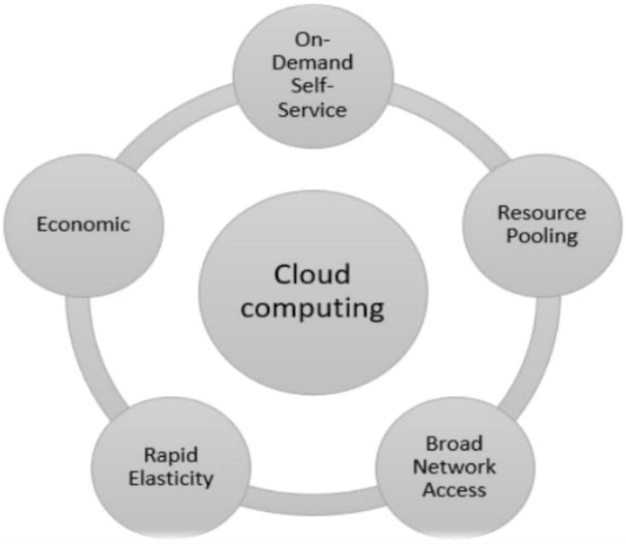

Рисунок 1.3 – Характеристики СС

Користувач платить за використані послуги без необхідності інвестувати в обчислювальну інфраструктуру, необхідну для роботи та підтримки ресурсів. На додаток до безкоштовного надання деяких послуг не потрібно сплачувати покриття або додаткові збори.

1.3. Технологія шифрування даних

Коли дані передаються в хмару, їх безпека вразлива. Шифрування є ефективним методом захисту даних. Суть шифрування даних полягає в перетворенні вихідного файлу відкритого тексту або даних у рядок нечитабельного коду за допомогою деяких алгоритмів, який зазвичай називають зашифрованим текстом. Навіть якщо хтось перехопить спотворений код, він/вона не зможе використати спотворений код для отримання оригінального вмісту, що ефективно захищає конфіденційність даних і запобігає фальсифікації даних. Користувачі, які мають право доступу, можуть розшифрувати файл за допомогою відповідного закритого ключа, а потім оновити та змінити зашифрований текст. Шифрування поділяється на симетричне шифрування та асиметричне шифрування. Симетричне шифрування використовує секретний ключ для шифрування та дешифрування даних. Однак перед використанням симетричного шифрування користувачам

необхідно визначити консенсусний ключ, що дуже незручно для спільного використання файлів кількома користувачами. Для порівняння, асиметричне шифрування, також відоме як шифрування з відкритим ключем, є більш зручним. Шифрування з відкритим ключем містить пару ключів. Відкритий ключ, який можна розкрити іншим для шифрування файлів, тоді як закритий ключ використовується для розшифрування тексту.

1.3.1. Шифрування на основі ідентифікації

У традиційній PKI (інфраструктурі відкритих ключів), щоб підтвердити, що ідентифікаційна інформація узгоджується з відкритим ключем, який використовується для шифрування, відправнику потрібно автентифікувати ідентифікаційну інформацію одержувача через довірений сторонній центр сертифікації (CA) перед шифруванням файлу відкритим ключем. Цей процес може призвести до значного збільшення робочого навантаження відправника, якщо він хоче поділитися даними з кількома одержувачами.

Щоб вирішити цю проблему, у 1984 році Шамір [8] запропонував концепцію криптографії на основі ідентифікації. Ідея полягає в тому, щоб пов’язати ідентифікаційну інформацію користувача з відкритим ключем, щоб не було необхідності перевіряти сертифікат одержувача перед шифруванням.

У 2001 році Боне та Франклін [12] офіційно дали визначення та модель безпеки шифрування на основі ідентифікації IBE та застосували білінійну карту для побудови безпечної схеми IBE у своїй основоположній статті. У такій системі Аліса є відправником, який хоче надіслати Бобу зашифроване повідомлення. Генератор приватних ключів (PKG), довірена третя сторона, потрібен для генерації відповідного відкритого та закритого ключів. По-перше, щоб зашифрувати повідомлення, Аліса використовує унікальну ідентифікаційну інформацію одержувача (електронна адреса Боба: Bob@g.com), щоб згенерувати відкритий ключ із PKG. Потім Аліса надсилає зашифроване повідомлення Бобу. Одержувач Боб зв’язується з PKG і проходить автентифікацію, щоб отримати відповідний закритий ключ. На рисунок 1.4 показано, як працює шифрування на основі ідентифікації.

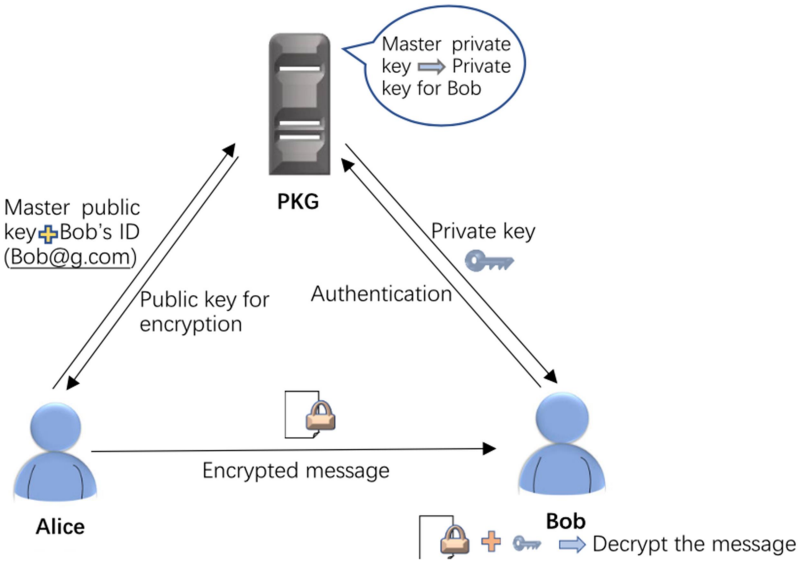

Рисунок 1.4 – Шифрування на основі ідентифікації

Алгоритм відкликаного IBE зазвичай приймає загальнодоступний параметр PP, ідентифікатор користувача, список відкликань RL, час відкликання t і стан st як вхідні дані, а оновлений список відкликань – як вихідні дані.

1.3.2. Постквантове шифрування

В останні роки зі швидким розвитком блокчейну, Інтернету речей і квантових обчислень увага світу до безпеки та конфіденційності даних зросла до безпрецедентного рівня, що висуває дедалі вищі вимоги до безпеки та захисту конфіденційності даних. Зараз безпека криптографії з відкритим ключем залежить від деяких математичних проблем (таких як проблема дискретного логарифмування та розкладання великих цілих чисел на множники), які важко розв’язати на традиційних комп’ютерах і класичних алгоритмах. У 1994 році запропонований короткий алгоритм безпосередньо загрожував RSA та пов’язаним алгоритмам. Останнім часом дослідження та розробки квантового комп’ютера стали центром уваги багатьох комерційних компаній. Хоча неясно, коли буде реалізований практичний квантовий комп’ютер, деякі квантові комп’ютери вже розроблені, наприклад компанія Honeywell нещодавно оголосила про створення 64-розрядного комп’ютера.

Постквантова криптографія – це нове покоління криптографії, яке може протистояти атаці квантового комп’ютера на існуючу криптографію. Нижче наведено поточні дослідження та існуючі відкриті питання щодо основних алгоритмів постквантового шифрування.

1) Механізмом автентифікації алгоритму підпису на основі хешування є хеш-дерево Меркеля, безпека якого залежить від стійкості хеш-функції до колізій. Хеш-дерево Меркеля використовується для аудиту цілісності, видалення даних [9]. Через використання деревовидної структури в схемі побудови на основі хешів наразі існує лише конструкція цифрового підпису та дуже мало шифрування з відкритим ключем системи.

2) Алгоритм на основі решітки може реалізувати криптографічну конструкцію, таку як шифрування, цифровий підпис, атрибутивне шифрування та гомоморфне шифрування, безпека яких залежить від складності вирішення проблем у решітці. За однакової безпеки алгоритм на основі решітки має менший розмір відкритого ключа, вищу швидкість обчислень і більш високий рівень безпеки порівняно з алгоритмом на основі хешу. Останнім часом стрімко розвивається конструкція решітчастої криптографії на основі LWE (навчання з помилками) [14] і RLWE (ring-LWE) [10]. Наприклад, зазначається, що дослідження Wei та ін. щодо відкликаного накопичувача IBE [9] базується на білінійному сполученні. Їх схема має хорошу продуктивність, але не може протистояти квантової атаки. Відкличне сховище на основі решітки все ще потребує подальшого дослідження.

1.4. Життєвий цикл безпеки хмарних даних

Адміністрування безпеки хмарних даних — це багатоетапний процес, який часто описують як життєвий цикл. Глибоке розуміння життєвого циклу безпеки даних також слугує дорожньою картою для проактивного захисту даних.

Життєвий цикл безпеки даних складається з ключових етапів:

Класифікація та виявлення даних: розпізнавайте та класифікуйте свої дані відповідно до їх конфіденційності. Знання характеру даних є невід'ємною частиною адміністрування належного контролю безпеки.

Шифрування та керування доступом: застосовуйте такі захисні заходи, як шифрування та псевдонімізація. Поєднайте це з керуванням доступом, щоб лише авторизований персонал міг отримати доступ до конфіденційних даних.

Виявлення вторгнень, запобігання та оцінка ризиків. Використовуйте розширені інструменти аналізу загроз і безпеки для швидкого виявлення загроз і проактивного запобігання. Крім того, проводите звичайну оцінку ризиків для ефективного прогнозування та попередження потенційних загроз.

Реагування на інциденти, криміналістика та відновлення: готуйтеся до сценаріїв порушень не лише теоретично, а й практично. Реалізуйте плани швидкого реагування, проводите судово-медичний аналіз після інциденту та забезпечуйте швидке обслуговування та відновлення даних для підтримки безперервності бізнесу.

Моніторинг відповідності, звітування та безперервний аудит: регулярні перевірки та аудити забезпечують дотримання правил і визначають потенційні сфери для вдосконалення. Регулярно контролюйте відповідність і повідомляйте про результати для забезпечення прозорості.

Постійне вдосконалення та виведення з експлуатації/утилізація даних: повторюваний крок для постійного вдосконалення вашої безпеки. Це також забезпечує безпечне виведення з експлуатації та практику утилізації даних, коли дані або системи більше не потрібні.

1.5. Проблеми хмарних обчислень

Клієнти послуг СС стикаються з наступними проблемами [16 – 18]:

− Відповідність нормативним вимогам: хоча клієнтські дані зберігаються у постачальника послуг, клієнти несуть головну відповідальність за захист і цілісність своєї власної інформації.

− Ізоляція даних: під час використання хмари дані часто передаються іншим клієнтам. Хоча шифрування є корисним, воно не є панацеєю від усього.

− Відновлення: хоча клієнти не знають про місцезнаходження їхніх даних, CSP повинен пояснити, що станеться з їхніми даними та послугою у разі аварії.

− Розташування даних: коли використовується СС, дані не є точно там, де вони розміщені. Клієнти не знають про місце, де зберігатимуться їхні дані.

− Підключення до мережі: у хмарній системі повільне підключення до мережі спричиняє вузькі місця під час передачі даних між концентраторами даних і всередині центру обробки даних, а збої в Інтернеті можуть призвести до величезних збитків для бізнесу.

− Операції вводу/виводу не значно покращують продуктивність у віртуальному середовищі порівняно зі спільною пам’яттю та часом центрального процесора. Більшість високопродуктивних обчислювальних програм потребує всіх посилань, які забезпечують одночасну роботу програми.

− Коли виникають помилки в хмарній системі, яка має широку базу користувачів, їх можна виправити лише у робочому середовищі в режимі реального часу. Вирішити такі проблеми може бути важко.

− Хмарні сервіси мають бути доступними завжди. Сервери повинні протистояти розподіленим атакам на відмову в обслуговуванні та відключенням електроенергії.

– Захист заходів конфіденційності та конфіденційності конфіденційних даних клієнтів є головним завданням СС, оскільки вони знаходяться під обслуговуванням і наглядом третьої сторони.

– Зловживання хмарними службами зі зловмисною метою.

– Захищайте та зберігайте дані від втрати, злому та крадіжки.

– Безпека мережі, рівень зв’язку, зберігання великих даних, зберігання та обчислювальна продуктивність, технічне обслуговування, захист веб-додатків периферійних обчислень, конфіденційність, надійність, цілісність

інформації, доступ до даних, автентифікація та витік даних – це додаткові проблеми, які потребують вирішення.

Безпека хмарного центру обробки даних здебільшого ідентична безпеці не хмарного центру обробки даних [9]. Необхідно захищати СС від будь-яких загроз. Деякі питання конфіденційності та безпеки, які вважаються важливими для СС [19]:

1. Зловмисні інсайдери: зловмисний інсайдер – це особа, яка має дозвіл на доступ до мережі та даних організації та використовує ці повноваження таким чином, що порушує конфіденційність і цілісність інформації організації та інформаційних систем. Більшість організацій усвідомлюють цю небезпеку, оскільки її важко виявити та вона має значний вплив на організацію.

2. Викрадення облікового запису або служби: ця загроза виникає через шахрайство та недоліки програмного забезпечення. У цьому випадку зловмисник може отримати доступ до конфіденційних регіонів у хмарі, де він може викрасти дозволи та конфіденційні дані.

3. Уразливості гіпервізора. Гіпервізор є найважливішою частиною програмного забезпечення віртуалізації. Гіпервізори мають очевидні вразливі місця в безпеці, а засоби правового захисту все ще обмежені та часто є власністю.

4. Незахищені інтерфейси та інтерфейси прикладного програмування (API): якщо використовується поганий набір інтерфейсів та API, організації можуть зіткнутися із загрозами безпеці, такими як невідомий доступ, повторне використання паролів, передача вмісту або аутентифікація відкритого тексту та негнучка керування доступом або недійсні авторизації.

5. Кібератаки: хакерство та кібератаки на мережі останнім часом дедалі більше стають серйозною загрозою.

РОЗДІЛ 2. АЛГОРИТМИ ШИФРУВАННЯ ДАНИХ В ХМАРНИХ ОБЧИСЛЕННЯХ

2.1. Архітектура безпеки даних в хмарних обчисленнях

Як зазначалося раніше хмарні обчислення забезпечують платформу для розміщення всього, що потребує ресурсів, через Інтернет, а парадигми хмарних обчислень є привабливими, оскільки організації можуть уникнути витрат на встановлення та обслуговування IT-інфраструктури, наймаючи ресурси, використовуючи просту політику оплати та використання для того, що вони хочуть використовувати та коли [20]. Крім того, постачальники хмарних обчислень можуть отримати вигоду від значної економії, надаючи однакові послуги широкому колу клієнтів, особливо зацікавленим сторонам розумних систем.

Загальна архітектура безпечного вивантаження даних із пристроїв розумної системи в систему хмарних обчислень зображена на рисунку 2.1.

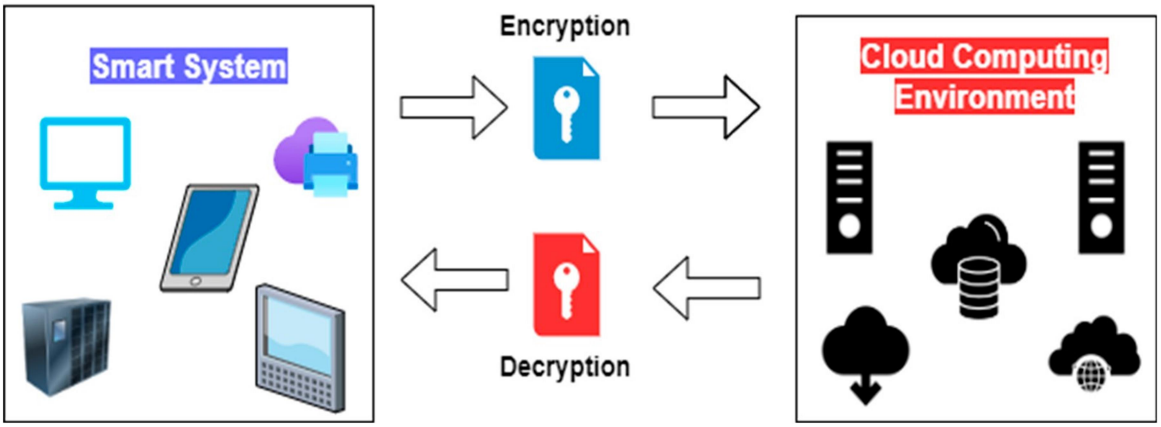

Рисунок 2.1 – Модель інтелектуальної системи обміну даних на основі хмарних обчислень

Одними з найбільш значних бар'єрів і перешкод на шляху швидкого зростання та впровадження хмарних обчислень у всій галузі є проблеми безпеки даних і конфіденційності. У наш цифровий день і епоху кількість

даних, що генеруються розумними системними пристроями, щодня зростає, що призводить до попиту на більше зберігання даних і швидшу обробку. Найголовнішою метою кожної організації є певним чином зменшити витрати на зберігання та обробку даних, зберігаючи той самий рівень аналізу даних та інформації. Однак конфіденційність даних має першорядне значення, оскільки дані переміщуються між розумною системою та ресурсами хмарних обчислень. Компанії повинні мати рівень довіри до моделі, щоб прийняти хмарні обчислення. Щоб досягти вищого рівня довіри та забезпечити безпечну передачу їхніх даних, алгоритми шифрування є рішенням для захисту конфіденційних даних. Дослідники запропонували багато алгоритмів для вирішення проблем довіри та досягнення високого рівня безпеки хмарних даних. Узагальнена модель безпеки хмарних даних наведена на рисунку 2.2.

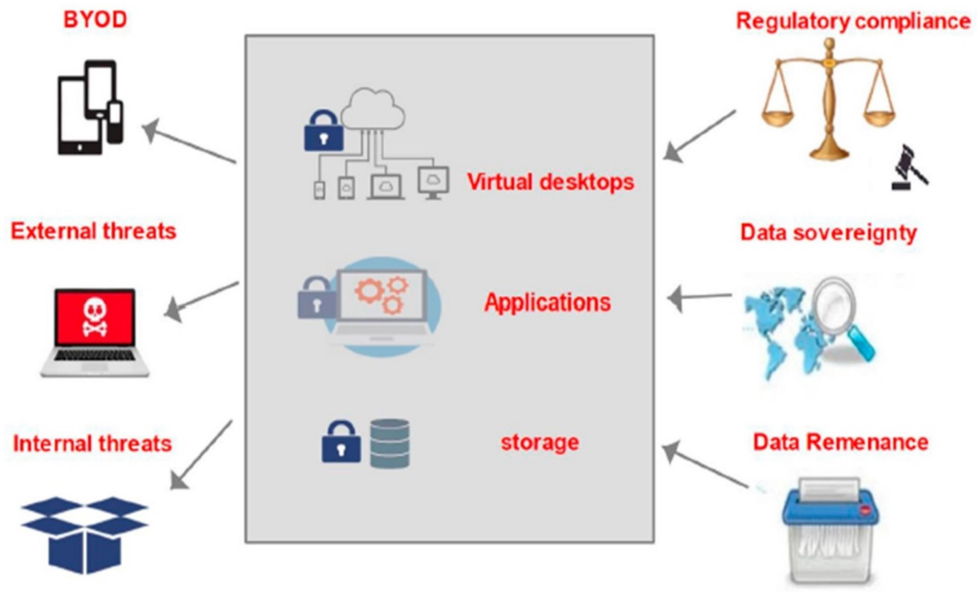

Рисунок 2.2 – Модель безпеки хмарних даних

Хмарні обчислення використовують спільні інфраструктурні ресурси для виконання аналізу даних та забезпечення вищого рівня обслуговування без витрат на додаткове технічне обслуговування та управління [21]. Будь-які дані, що передаються, мають бути захищені від зловмисних перехоплень і порушень конфіденційності даних.

Існують способи керування цією передачею з точки зору архітектури, але всі такі методи залишають дані у вигляді звичайного тексту, який може зрозуміти будь-який зловмисник. Процес шифрування має вирішальне значення в домені хмарних обчислень, оскільки дані передаються через Інтернет на платформу хмарних обчислень з локальних серверів і, у багатьох випадках, між ресурсами хмарних обчислень.

2.2. Порівняльний аналіз алгоритмів шифрування даних

Оскільки вся ця передача відбувається через Інтернет, надсилання даних у вигляді звичайного тексту становить значний ризик для безпеки та конфіденційності даних. Є багато способів перехопити ці дані в мережах, які вимагають додаткових зусиль для захисту даних. Шифрування значною мірою вирішує цю проблему, гарантуючи, що дані неможливо розшифрувати під час передачі.

Шифрування перетворює дані в іншу форму, яку можуть розшифрувати лише користувачі з відповідними ключами або іншими механізмами доступу [22].

Зашифровані дані зазвичай називають зашифрованим текстом, а розшифровані – звичайним текстом. Шифрування можна розділити на симетричне шифрування, шифрування з відкритим ключем і асиметричне шифрування, також відоме як шифрування з закритим ключем [23].

Коли шифрування та дешифрування виконуються з використанням одного ключа для обох функцій, це називається шифруванням із симетричним ключем (рис. 2.3), тоді як для асиметричного шифрування генерується пара відкритий-приватний ключ, де відкритий ключ шифрує, а закритий ключ розшифровує дані.

Одна з переваг асиметричного шифрування полягає в тому, що розподіл ключів можна контролювати краще та безпечніше.

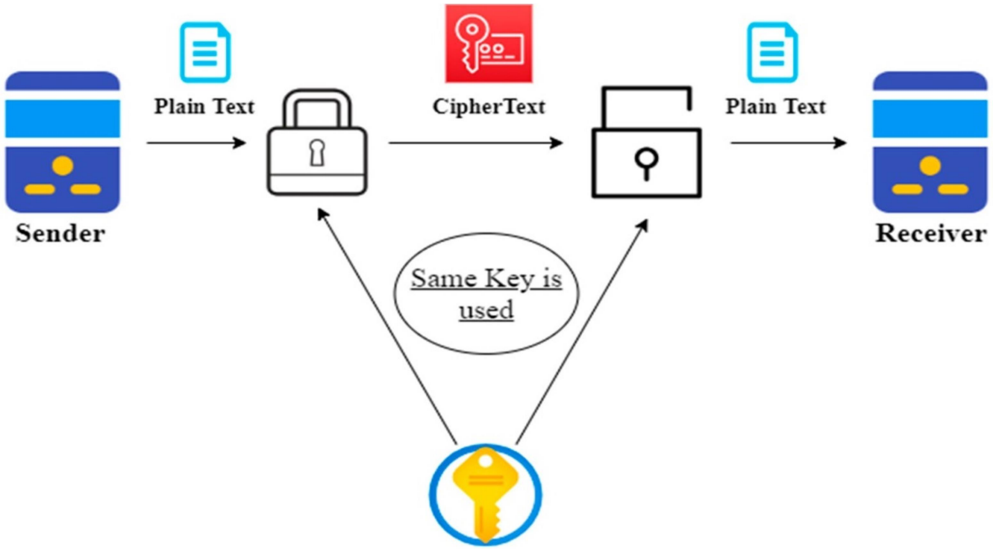

Рисунок 2.3 – Архітектура шифрування з симетричним ключем

Наприклад, якщо хтось отримує доступ до відкритого ключа, він марний і не може розшифрувати дані (рис. 2.4).

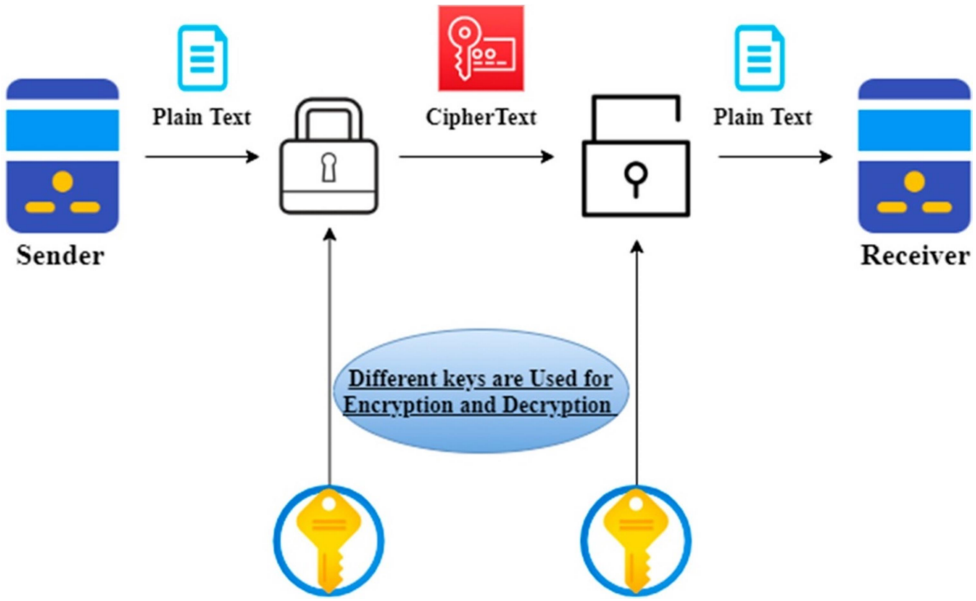

Рисунок 2.4 – Архітектура шифрування з асиметричним ключем

Тому, вкрай важливо визначити правильний метод розповсюдження ключів, щоб гарантувати, що вони не опинилися в руках шахраїв/пристроїв.

На рисунку 2.5 представлено загальну класифікацію алгоритмів шифрування [23].

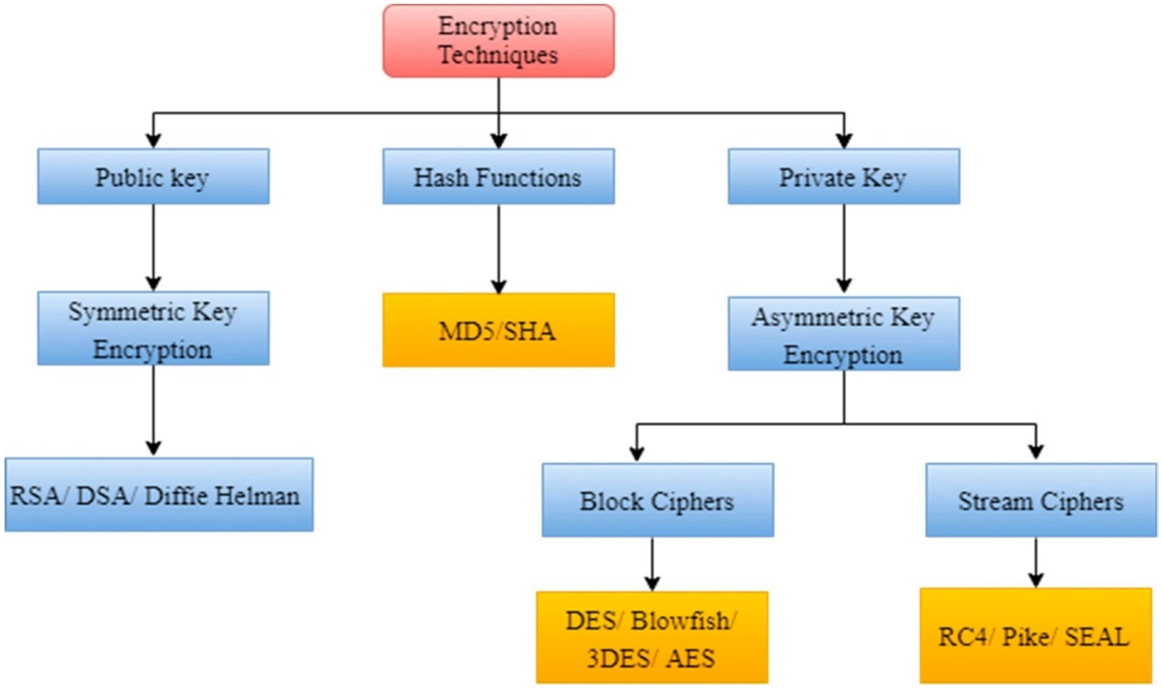

Рисунок 2.5 – Класифікація алгоритмів шифрування

Алгоритми шифрування відіграють важливу роль у безпеці хмарних даних. Здійснивши порівняння алгоритмів, що представлені на рисунку 2.5, а саме: Міжнародний алгоритм шифрування даних (IDEA), алгоритми AES, RSA, Blowfish і DES можна визначити найкращий алгоритм безпеки. Результати проведеного порівняння наведено в таблиці 2.1.

Таблиця 2.1 показує, що RSA є асиметричним алгоритмом, а IDEA, AES, Blowfish і DES є симетричними алгоритмами. RSA та IDEA менш безпечні, ніж AES, Blowfish і DES.

В даному дослідженні алгоритм AES займає найменшу кількість часу для шифрування хмарної інформації, алгоритм Blowfish вимагає найменшого обсягу пам’яті, а алгоритм AES можна використовувати для шифрування величезних обсягів даних.

AES є швидшим за інші алгоритми та є найкращим алгоритмом з точки зору параметрів автентифікації. RSA споживає найбільше пам’яті та потребує максимального часу шифрування.

Таблиця 2.1

Результати оцінки алгоритмів IDEA, AES, RSA, Blowfish і DES

Параметри	Алгоритм шифрування				
	IDEA	AES	RSA	Blowfish	DES
Платформа	Хмара	Хмара	Хмара	Хмара	Хмара
Тип шифру платформи	Симетричний	Симетричний	Асиметричний	Симетри-чний	Симетри-чний
Рівень безпеки	Безпечно тільки для клієнта	Безпечно тільки для клієнта	Безпечно тільки для клієнта	Безпечно для клієнта та провай-дера	Безпечно для клієнта та провай-дера
Можливість шифрування даних	Шифрування невеликих обсягів даних	Шифрування невеликих обсягів даних	Шифрування невеликої кількості даних	Менше, ніж AES	Менше, ніж AES
Аутентифікація	Менша ніш AES	Сильна	Сильна	Ідентичний AES	Менше, ніж AES
Використання пам'яті	Вимагає великого обсягу пам'яті	Потрібна максимальна кількість пам'яті	Потрібна максимальна кількість пам'яті	Вимагає мінімум пам'яті	Більше ніж AES
Час шифрування	потребує максимального часу	потрібен максимальний час	потрібен максимальний час	Більше ніж AES	Більше ніж AES

Шифрування даних — це спосіб дозволити дані, відокремлюючи їх від пристрою, на якому вони зберігаються. Адміністратори можуть зберігати та надсилати дані через незахищені канали. Шифрування даних забезпечує безпеку конфіденційної інформації та інтелектуальної власності. Дані захищені незалежно від їх передачі, оскільки в них вбудовано шифрування.

Багато організацій дотримуються суворих правил і принципів конфіденційності. Шифрування прокладає шлях, оскільки дані можуть переглядати лише одержувач, який володіє ключем для їх розшифровки.

Шифрування даних може бути досить дорогим, оскільки системи, які підтримують його в актуальному стані, повинні мати відповідну потужність і вдосконалення.

2.3. Аналіз алгоритмів шифрування даних в хмарних обчисленнях

Вищезазначені властивості алгоритмів шифрування також застосовують і при шифруванні даних що передаються на хмару. Проте, зазвичай для реалізації хмарних обчислень застосовують поєднання кількох алгоритмів в одне ціле.

2.3.1. Blowfish зі стисканням файлу

Хмарні платформи дозволяють користувачам використовувати спільні ресурси, не вкладаючи багато коштів і не піклуючись про обслуговування сервера. Безпека даних є першочерговою проблемою впровадження технології хмарних обчислень. Гровер та ін. [22] запропонував Blowfish із механізмом стисненого файлу для захисту даних перед їх збереженням у ресурсах хмарного сховища за допомогою методів шифрування та зменшення простору для зберігання. Файл спочатку стискається, а потім шифрується за допомогою алгоритму Blowfish, як показано на рисунку 2.6.

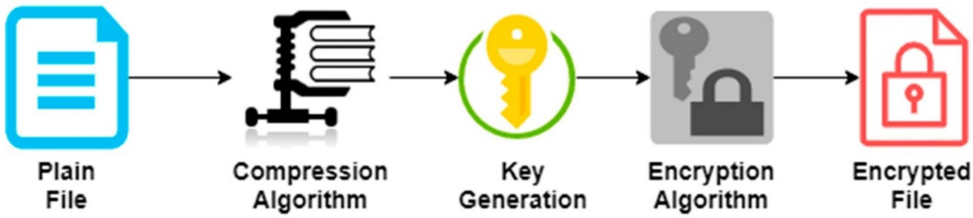

Рисунок 2.6 - Шифрування Blowfish

Запропонований метод вирішує проблему зменшення часу та простору шифрування шляхом стиснення файлу даних. Часову ефективність запропонованого підходу перевірено за допомогою трьох різних розмірів файлів. Потім результати порівнюються шляхом обчислення часу шифрування для не стиснутих і стиснених файлів.

Використовуючи цей підхід, якщо файл стискається перед застосуванням стратегії шифрування, можна зменшити час шифрування та

простір, необхідний для збереження цього файлу. Однак для більших розмірів файлів його продуктивність не обговорюється. Зі збільшенням кількості користувачів керування ключами стане складною проблемою.

Підхід до шифрування Blowfish використовується для захисту даних на стороні клієнта, що економить час, який витрачається на стороні сервера. Слабкість цього підходу полягає в тому, що зі збільшенням розміру файлу потрібно більше часу для стиснення та шифрування, а отже, продуктивність знижується.

2.3.2. Поєднання RSA з AES

Ханезаї та ін. [23] використовували дещо інші методи шифрування; вони використовували комбінацію Rivest Shamir Adelman (RSA) і Advanced Encryption Standard (AES) для шифрування. За допомогою методу подвійного шифрування користувачі обмінюються більш безпечними даними. RSA використовується для підвищення складності зашифрованого тексту, а AES використовується для швидкого пошуку даних. Використання RSA для передачі файлів захищено завдяки генерації асиметричних ключів, що займає багато часу.

Хмарний сервіс генерує відкритий ключ (PB), закритий ключ (PK), ідентифікатор файлу та велике випадкове число (RB). Спочатку користувач запитує PB з хмари. Хмарна система надає користувачеві PB та ідентифікатор файлу. Потім користувач надсилає файл, зашифрований за допомогою RSA, у хмару. Коли користувач запитує певний файл із хмарного сервісу, він надсилає запит на сервер із відкритим ключем.

Хмарна служба знаходить запитуваний файл у системі хмарного зберігання (CSS). Потім цей файл шифрується за допомогою AES. RB, який є секретним ключем симетричного алгоритму, шифрується за допомогою відкритого ключа. Потім CSS надсилає RB і запитуваний файл користувачеві. Симетричний алгоритм використовується через проблему розподілу ключів, але як керувати цими ключами не обговорюється.

Сильна сторона цього підходу полягає в тому, що він використовує гібридний підхід із використанням методів шифрування RSA та AES для забезпечення безпеки даних. Використання RSA підвищує рівень складності злому даних, тоді як AES скорочує час, необхідний для передачі файлів між користувачем і хмарним сховищем даних.

Основним недоліком запропонованого підходу є те, що якщо розмір файлу збільшується, кількість згенерованих ключів також збільшується, що породжує проблему керування ключами. Іншим недоліком є те, що час шифрування та дешифрування також є накладним для великих розмірів файлів.

2.3.3. Розширений стандарт шифрування

З розвитком технології хмарних обчислень постачальники хмарних послуг повинні вирішувати питання конфіденційності даних і безпеки. У [24] використовують підхід безпеки даних AES, який застосовується для шифрування даних перед надсиланням їх у хмару.

AES краще працює в програмному та апаратному середовищах як на 8-розрядних, так і на 64-розрядних платформах. AES потребує менше пам’яті, що робить його придатним для середовищ із меншим простором. Ключі AES легко налаштувати та підтримують будь-який блок, а розмір ключів, кратних 32, має бути більшим за 128. Дані надсилаються до постачальника хмарних послуг після того, як вони зашифровані користувачем за допомогою алгоритму AES. Користувач сам керує даними та ключем, що забезпечує цілісність даних. AES забезпечує менше споживання пам’яті та менше часу на обчислення, ніж інші методи шифрування. Щоб зберегти ключі, цей підхід передбачає встановлення окремого фізичного сервера на стороні користувача, що збільшує вартість обладнання.

Сильна сторона підходу AES полягає в тому, що шифрування та дешифрування даних виконує користувач, а не постачальник хмарних послуг.

2.3.4. Гомоморфне шифрування

Чжао та ін. [25] використовували техніку гомоморфного шифрування, що дозволяє користувачам виконувати обчислення із зашифрованими даними без їх дешифрування. Таким чином, у цій техніці користувач може працювати із зашифрованим текстом, не відкриваючи вихідні дані в хмарі, що забезпечує додатковий захист даних.

Результати гомоморфної техніки шифрування аналізуються за допомогою простого додавання та множення. Множення та додавання застосовуються до зашифрованого тексту, а їхні результати порівнюються з відкритим текстом. Обчислення зашифрованого тексту складніше, ніж звичайного тексту. Тому такі складні обчислення займуть дуже багато часу для невеликого набору даних. Користувач спочатку входить у хмару, вибираючи розділ зберігання на основі рівня безпеки даних. Якщо вибрано приватний розділ, то для шифрування використовується алгоритм AES, тоді як для публічного розділу використовується метод шифрування Blowfish. Для гібридного зберігання даних пропонуються дві моделі шифрування даних, які забезпечують безпеку в приватному або загальнодоступному розділі. Якщо потрібен низький рівень безпеки, можна вибрати спрощений розширений стандарт шифрування (S-AES), міжнародний алгоритм шифрування даних (IDEA) або техніку шифрування Blowfish. Спочатку використовується шифрування Blowfish для високого рівня безпеки, а потім використовується IDEA. Після того, як файл зашифровано, алгоритм безпечного хешування (SHA-1) використовується для створення коду цілісності. Цей код додається на початку зашифрованого файлу, і знову використовується SHA-1 для створення буквено-цифрового маркера з 16 цифр. Для отримання файлу користувач спочатку проходить автентифікацію, після чого маркер SHA-1 порівнюється з маркером, наданим під час отримання, щоб зберегти цілісність даних.

Сильна сторона підходу до гомоморфного шифрування полягає в тому, що він забезпечує безпеку в публічних, приватних і гібридних розділах зберігання за допомогою різних алгоритмів шифрування зі схемами перевірки

цілісності. Користувач вибирає розділ зберігання хмари відповідно до вимог безпеки. Приватний розділ забезпечує найвищу безпеку за допомогою алгоритму AES. У загальнодоступному розділі забезпечення безпеки обмежене. Цей розділ найкраще працює, якщо користувач бажає швидкого обчислення та меншого часу на шифрування та дешифрування.

2.4. Проблеми безпеки даних у хмарних обчисленнях та методи їх вирішення

Безпека даних у середовищі хмарних обчислень є складною проблемою, яка спонукає до дослідження нових механізмів для досягнення певного рівня довіри. Можна виділити кілька ключових аспектів, що впливають на довіру до хмарних технологій в цілому та хмарних обчислень зокрема.

Цілісність даних і конфіденційність. Незважаючи на те, що хмарні обчислення забезпечують ілюзію ресурсів за нижчою ціною, нижче управління ресурсами та високу доступність, вони сприйнятливі до загроз безпеці, оскільки кількість користувачів хмари зростає експоненціально, а область додатків, розміщених у хмарі, надзвичайно велика.

Ці речі створюють серйозніші загрози безпеці для покупців хмари. Пов’язана хмарна атака, що процвітає на об’єкти знань, може призвести до порушення знань і несанкціонованого доступу до інформації користувача. Через такі порушення цілісності хмарні знання втратили свою мультитенантну природу. Зокрема, постачальники SaaS також могли втратити свої технічні знання. Віртуалізація кількох фізичних ресурсів серед багатьох користувачів призводить до атак зловмисних інсайдерів постачальника хмарних послуг (CSP) та організації [26].

Засіб спільних ресурсів дозволяє зловмисникам здійснювати атаки на знання різних клієнтів і обробляти їхні знання, таким чином порушуючи конфіденційність і цілісність даних. Інший значний ризик полягає в тому, що CSP передає свої знання сторонньому сховищу.

Генерація ключів і керування криптографією хмарних обчислень не стандартизовані та відповідають вимогам. Ненормальне та небезпечне керування ключами заважає якісним алгоритмам криптографії добре працювати в загальних моделях хмарних обчислень. Така криптографія може також убезпечити від потенційних ризиків для хмарних обчислень.

Безпека приватної хмари. Проблеми безпеки охоплюють високу цінність впровадження та управління, необхідні навички та управління вразливими місцями. Завдяки цій моделі готовності реалізація безпеки зазвичай підтримується оцінкою ризику, і, таким чином, захисний капот не є всеосяжним.

Безпека публічної хмари. Безпека публічної хмари є складнішою, ніж приватна хмара, оскільки ресурси, здається, не завжди задіяні; однак це використовується багатьма хмарними покупцями. Це додає додатковий тягар гарантування доступу до всіх програм і знань у загальнодоступній хмарі, і, крім того, необхідно керувати безліччю зовнішніх впливів, таких як законодавчий захист і захист знань.

Гібридна хмарна безпека. У гібридній хмарі проблеми з безпекою порівняно високі, оскільки модель підготовки складна через неоднорідне налаштування, численні оркестровки та інструменти автоматизації. Це налаштування вимагає додаткових витрат із будь-яким недоглядом, що призводить до необхідного ризику.

Ризики безпеки. Організатори в освітньому секторі бажають використовувати хмарні сервіси, які, здається, не відрізняються радикально від тих сервісів, якими керують у їхніх центрах. Проблеми хмарних обчислень класифікуються за чотирма основними аспектами: мережа, керування доступом, хмарна інфраструктура та безпека знань. Це важливі сфери, які піддають ризику дані користувачів. Щоб зрозуміти успіх у вирішенні проблем безпеки та їх викликів у вищих навчальних закладах, потрібно проаналізувати різноманітні аспекти хмарних викликів, такі як загрози, ризики та моделі атак.

Безпека мережі. Середовище передачі, через яке користувач

підключається до хмарної інфраструктури, є вразливим до ризиків безпеки. Надання захищеного середовища запобігає витоку конфіденційних даних у середовищі передачі. CSP потребує захисту, щоб захистити знання від звичайних мережевих атак, таких як DoS [27], атака Man-in-the-Middle, підробка обробки інформації, перехоплення пакетів і сканування портів [28].

Ризики хмарної інфраструктури. Найбільш поширеною та важливою проблемою безпеки, з якою стикаються хмарні системи, може бути відсутність захисту віртуальної машини. Через те, що на еквівалентному ПК встановлено кілька віртуальних машин, користувач не може розмістити між ними апаратний пристрій захисту, наприклад брандмауер. Іншою проблемою є динамічна атмосфера, де віртуальні машини створюються, припиняються або переміщуються в інше місце механічно, що робить жахливим спостерігати за трафіком і перевіряти, чи наростає атака.

Інтероперабельність (здатність до взаємодії) і портативність. Компанії повинні мати можливість переходити до парадигм хмарних обчислень і з них і переходити до зовсім інших постачальників. Послуги хмарних обчислень повинні мати можливість безперебійно працювати з локальними IT. Згідно з опитуванням IDC [29] (з розміром вибірки 244), проведеним у 2008 році, сім IT-систем і додатків, які користувалися хмарними можливостями, були: додатки для управління IT (26,2%), кооперативні додатки (25,4%), персональні Додатки (25%), бізнес-додатки (23,4%), розробка та підготовка додатків (16,8%), серверні можливості (15,6%) і можливості зберігання (15,5%). Цей результат показує, що організації все ще мають міркування щодо безпеки та конфіденційності, переміщуючи свою інформацію в хмару. Опитування показує, що 31,5% організацій можуть перенести свої можливості зберігання даних у хмару за три роки. Однак цей діапазон залишається низьким порівняно з кооперативними програмами (46,3%) на той момент. Однією з головних причин браку є проблема сумісності. Портативність також є серйозною проблемою та вузьким місцем.

Вимірювання ефективності та вартість інформації. Компанії не

витрачають гроші на купівлю обладнання; однак за інформаційні заходи їм доводиться платити. це може бути значення кави для невеликих програм, але значно високе для програм, які потребують багато знань. Для того, щоб передати розширену та точну інформацію через мережу, потрібні відповідні інформаційні заходи. Через це деякі компаній все ще очікують зниження вартості перед переходом у хмару.

Науковці зосереджені на дослідженні хмарних обчислень та пошуку можливих рішень для подолання щоденних нових проблем безпеки завдяки чому можна виділити наступні кроки для захисту даних при хмарних обчисленнях:

- Автентифікація користувача. Користувач у хмарі має бути законним користувачем. Щоб перевірити, чи є будь-які невідповідні зміни в даних та інформації, цілісність є найкращим методом [30]. Для такого типу вирішення проблем використовується підхід цифрового підпису [31]. Підхід, запропонований у [32], є децентралізованим і надійним, у якому хмара служить для ідентифікації кінцевого користувача, не знаючи його інформації, яка зберігається в зашифрованому вигляді. Справжній користувач може лише розшифрувати цю інформацію.
- Конфіденційність. Щоб уникнути даних, необхідних для розробки безпечної системи зберігання даних у хмарі, конфіденційність може бути досягнута, якщо система розгорнула хороший алгоритм шифрування та ефективну систему керування ключами. Криптографія на основі атрибутів [33] є можливим рішенням з метою, щоб клієнт міг обмінюватися інформацією адаптованим і динамічним способом.
- Шифрування. Використання алгоритму шифрування є найкращим способом запобігання та захисту даних чи інформації в системі хмарних обчислень. Використання добре розробленого алгоритму забезпечує найкращу систему безпеки. Основна проблема використання алгоритму шифрування збільшує час обчислення. Якщо використовується подвійний алгоритм шифрування [34], він експоненціально збільшує час

обчислення. Інший метод, який називається повністю голоморфним шифруванням, може обчислювати результати обробки зашифрованої інформації, а не вихідних даних, що може створити потенційну таємницю інформації.

- Орієнтований на дані підхід до втрати даних. Якщо даними не керують належним чином, може виникнути проблема зберігання даних і несанкціонованого доступу [35]. Коли дані переміщуються в хмару, виправдано хвилюватися про їх безпеку. Втрата інформації з хмари через випадкове стирання, згубну зміну або демонстрацію природи, що припиняє роботу організації, що спеціалізується на хмарі, може бути жалюгідною для бізнесу. Як правило, DDoS-атака – це лише занепокоєння для більш серйозної небезпеки, наприклад, намагання взяти або стерти інформацію. Щоб запобігти втраті даних, розроблено підхід, орієнтований на дані.
- Управління ключами. «Погане шифрування — це погано, але погане керування ключами — ще гірше». Керування ключем є найважливішою проблемою безпеки в хмарних обчисленнях. Існує дуже складний метод зберігання зашифрованого ключа в хмарі. Вирішенням цієї проблеми є дворівневе шифрування використовуваних ключів [35]. Техніка подвійного шифрування ускладнює легке розшифрування секретної інформації.

Незважаючи на те, що хмарні обчислення забезпечують такі переваги, як спільне використання ресурсів і послуги на вимогу, які надаються без надто великих витрат на розбудову інфраструктури та купівлю ресурсів, для розвантаження даних, створених інтелектуальними системами, необхідно вирішити багато проблем безпеки. У літературі запропоновано багато алгоритмів шифрування та моделей конфіденційності даних для вирішення виникаючих проблем, але ця проблема все ще залишає прогалини для дослідників.

Переглядаючи різні алгоритми шифрування даних, можна зробити

висновок, що під час вибору алгоритму у залежності від бізнес-потреб організації необхідно досягти тонкого балансу між складністю та безпекою. Жоден алгоритм не можна назвати «найкращим» або «універсальним» — кожен алгоритм потрібно вивчати на предмет його переваг і недоліків. Складний алгоритм зазвичай вимагає більше часу для шифрування або дешифрування, тоді як алгоритм меншої складності може не підходити для дуже конфіденційних даних. Алгоритм Blowfish виділяється серед алгоритмів симетричного шифрування. Якщо є обмеження на потужність обробки та час, AES є найбезпечнішим із симетричних алгоритмів. Алгоритм RSA – це асиметричний алгоритм, який підходить для спільного використання конфіденційної інформації, оскільки його пара публічно-приватного ключа більш забезпечена.

Зараз тривають дослідження, спрямовані на пошук алгоритму шифрування, який добре масштабується зі збільшенням кількості даних, які з високою швидкістю генеруються інтелектуальними системами, і є ефективним у продуктивності. Безпека даних є основною перешкодою, через яку керівництво інтелектуальних систем не рішуче переносить свої дані в хмару. Ключі, які використовуються для шифрування та дешифрування даних, мають бути більш захищеними, щоб третя сторона не могла зламати дані автентифікації. Досягнувши цього, ми можемо захистити дані від підробки. Комбінуючи різні методи шифрування, можна досягти безпеки даних, навіть якщо це призведе до збільшення часу шифрування та дешифрування, і, отже, продуктивність буде знижена. Для максимальної пропускної здатності в майбутній роботі можна розглянути можливість паралельного шифрування даних.

РОЗДІЛ 3. ВПРОВАДЖЕННЯ ПЕРСОНАЛЬНОГО ХМАРНОГО СХОВИЩА NEXTCLOUD

Хмарні сховища забезпечують швидкий, безпечний та простий спосіб завантаження даних у хмару. На відміну від хмарних сховищ для компаній, послуги хмарного сховища для персонального використання зазвичай дешевші, хоча в них часто відсутні розширені функції, які можуть і не знадобитися.

Сучасні хмарні служби зберігання даних включають наскрізне шифрування, а також рішення для синхронізації файлів, такі як керування версіями та збереження файлів. Ці функції допомагають зберігати дані в безпеці, а також бачити, коли внесено зміни.

3.1. Порівняльний аналіз поширених хмарних сховищ

На даний час існують десятки хмарних сховищ, частина з яких надає послуги безкоштовно. У випадку платних сховищ або платних покращень наданих безкоштовних послуг список доступного контенту для користувачів значно зростає.

Одним із самих поширених ресурсі є Google Drive. Він являє собою широко відомий хмарний сервіс зберігання даних із діапазоном ємності від 100 ГБ до 30 ТБ . Він пропонує чистий, простий і легкий для розуміння веб-інтерфейс. Крім того, кожен власник облікового запису Google із самого початку отримає безкоштовні 15 ГБ пам’яті. Диск Google є ідеальним вибором, якщо ви шукаєте просте персональне хмарне рішення для зберігання. Якщо потрібно більше місця та краща безпека, завжди є можливість оновити обліковий запис до Google One і почати використовувати його розширені функції.

Наступним по популярності є хмарне сховище OneDrive від Microsoft, яке має численні плани, що дозволяють зберігати та отримувати доступ до файлів і фотографій з будь-якого пристрою. Воно інтегрується з іншими програмами

Microsoft 365, що робить його найкращим варіантом хмарного сховища для користувачів Microsoft. Найпростіший і безкоштовний план пропонує 5 ГБ пам’яті, тоді як OneDrive Standalone передбачає 100 ГБ пам’яті за 1,99 дол. США на місяць . Однак він включає лише хмарне сховище.

Якщо ви в основному працюєте з мультимедійними файлами, pCloud може бути вибраним постачальником хмарних сховищ. Він має вбудований відео та аудіоплеєр для сортування музичних файлів у списки відтворення. pCloud пропонує два індивідуальні плани на вибір: Premium 500 ГБ і Premium Plus 2 ТБ . На відміну від своїх конкурентів, pCloud окрім помісячної підписки пропонує опцію довічної, що дозволяє значно заощадити гроші.

Окрім універсальних рішень, є й такі, що доступні тільки для користувачів певних пристроїв. Прикладом такого сервісу є IDrive. Це надійне та гнучке хмарне рішення для зберігання та резервного копіювання, яке дозволяє завантажувати дані з усіх ваших пристроїв і зберігати їх в одному обліковому записі в хмарі. Він пропонує низку планів для особистих, ділових і корпоративних користувачів. IDrive працює на ряді пристроїв під керуванням Windows, macOS, Android та iOS. Він пропонує безперервну синхронізацію файлів для даних, що зберігаються на всіх ваших пристроях зберігання, включаючи мережеві диски. Існує функція відновлення за допомогою перетягування, яка дозволяє відновлювати важливі файли, які могли бути випадково видалені. IDrive автоматично зберігає до 30 попередніх версій усіх файлів, що зберігаються на його серверах, тому дуже легко скасувати будь-які зміни, якщо ви передумали. А 256-бітне шифрування AES (Advanced Encryption Standard) захищає ваші файли від неавторизованого доступу. IDrive постачається з безкоштовним базовим планом, який пропонує 5 ГБ онлайн-сховища, а коли воно закінчиться, ви зможете перейти на преміум-план за 79,50 доларів США (5 ТБ) або 99,50 доларів США (10 ТБ) на рік.

Також, для користувачів macOS, iOS і iPadOS Apple пропонує власне персональне хмарне сховище Apple iCloud. Завдяки тому, що iCloud розроблено та керується компанією Apple, він пропонує повну інтеграцію та надзвичайно

зручне використання на будь-якому пристрої компанії. iCloud не накладає жодних обмежень на розмір окремого файлу. Крім того, будь-які предмети, придбані в iTunes Store, від пісень до програм чи ігор, можна безкоштовно зберігати на його серверах, не враховуючи ліміт пам'яті. Для безпеки Apple пропонує 128-бітне шифрування AES і двофакторну аутентифікацію для всіх користувачів. iCloud автоматично створює резервні копії всіх даних із ваших пристроїв Apple на своїх серверах. Ви також можете використовувати спеціалізований сервіс iCloud Drive для зберігання певних файлів у хмарі.

iCloud інтегровано в iWork, пакет продуктивності, який, хоч і не такий обширний, як Google Workspace або Microsoft 365, все ж дуже добре працює на пристроях від Apple. Apple iCloud поставляється з 5 ГБ безкоштовного сховища, у подальшому ви можете перейти на один із платних планів від 0,99 доларів на місяць за 50 ГБ до 9,99 доларів на місяць за 2 ТБ.

І останнім представником хмарних сховищ, яке варто згадати є Nextcloud. Це програмне забезпечення з відкритим вихідним кодом, яке вперше було розроблено в 2016 році. Воно дозволяє користувачам запускати власні персональні хмарні сховища. Серверне програмне забезпечення Nextcloud можна безкоштовно встановити на Linux, а клієнтське програмне забезпечення можна безкоштовно встановити на комп'ютерах під керуванням Windows, OS X або Linux. Існують також програми для Android та iOS, які дозволяють мобільним користувачам також безпечно зберігати свої дані на Nextcloud. Nextcloud є розгалуженням проекту OwnCloud , створеного багатьма оригінальними членами команди OwnCloud. Ці два проекти мають багато подібності, але вони відрізняються інтерфейсом і ліцензійними угодами, особливо для версій Enterprise.

Хоча Nextcloud навіть не наближається до числа найпопулярніших хмарних служб зберігання даних, таких як Google Drive і Dropbox, їй вдалося знайти власну аудиторію. Nextcloud продемонстрував великі перспективи та можливості як автономний постачальник хмарних рішень, з потенціалом стати головним гравцем у бізнесі розміщення хмарних файлів у найближчі роки, коли

цикл його розробки розгортатиметься далі, а завдяки наскрізному шифруванню забезпечує потрібний рівень довіри користувачів до ресурсу.

Порівняльний аналіз шифрування в згаданих вище хмарних сховищ представлено в таблиці 3.1.

Таблиця 3.1

Порівняльний аналіз шифрування в хмарних сховищ

Назва сховища	Безкоштовний простір	Максимальний платний простір	Кількість підключених пристроїв	Шифрування даних
Google Drive	15 ГБ	30 ТБ	Без обмежень	Без наскрізного шифрування
One Drive	5 ГБ	6 ТБ	30	Без шифрування з нульовим знанням
pCloud	10 ГБ	10 ТБ	5	256-бітне шифрування AES
IDrive	5 ГБ	10 ТБ	Без обмежень	256-бітне шифрування AES
iCloud	5 ГБ	2 ТБ	10	128-бітне шифрування AES
Nextcloud	5 ГБ	30 ТБ	Без обмежень	Наскрізне шифрування

Як зазначалося раніше при користуванні хмарними сховищами із загальним доступом зазвичай виникає проблема довіри, про яку згадувалося раніше у роботі. Відсутність шифрування з нульовим знанням значно зменшує довіру користувачів до таких сервісів, оскільки їх використання загрожує конфіденційності даних. Технічні спеціалісти таких сервісів можуть розшифровувати файли, якщо їх змусять зробити це правоохоронні органи, що робить систему потенційно вразливою для хакерів.

3.2. Підготовка середовища для Nextcloud

Для оцінки ефективності шифрування даних при використанні хмарних сховищ було проведено інсталяцію та налаштування персонального хмарного сховища Nextcloud.

Для його установки достатньо операційної системи Debian 12 з 4 ГБ оперативної пам’яті та двома процесорами.

В процесі розгортання необхідно налаштувати наступні пакети:

- Сервер Debian із доступом root;
- Apache2;
- PHP 7 і вище;
- mySQL / MariaDB;
- Nextcloud.

Відповідно першим кроком буде установка **Apache2**, для чого спочатку запускаємо команду *sudo apt update* для отримання останніх оновлень, а згодом *sudo apt install apache2*, для інсталяції цього пакету (рис. 3.1).

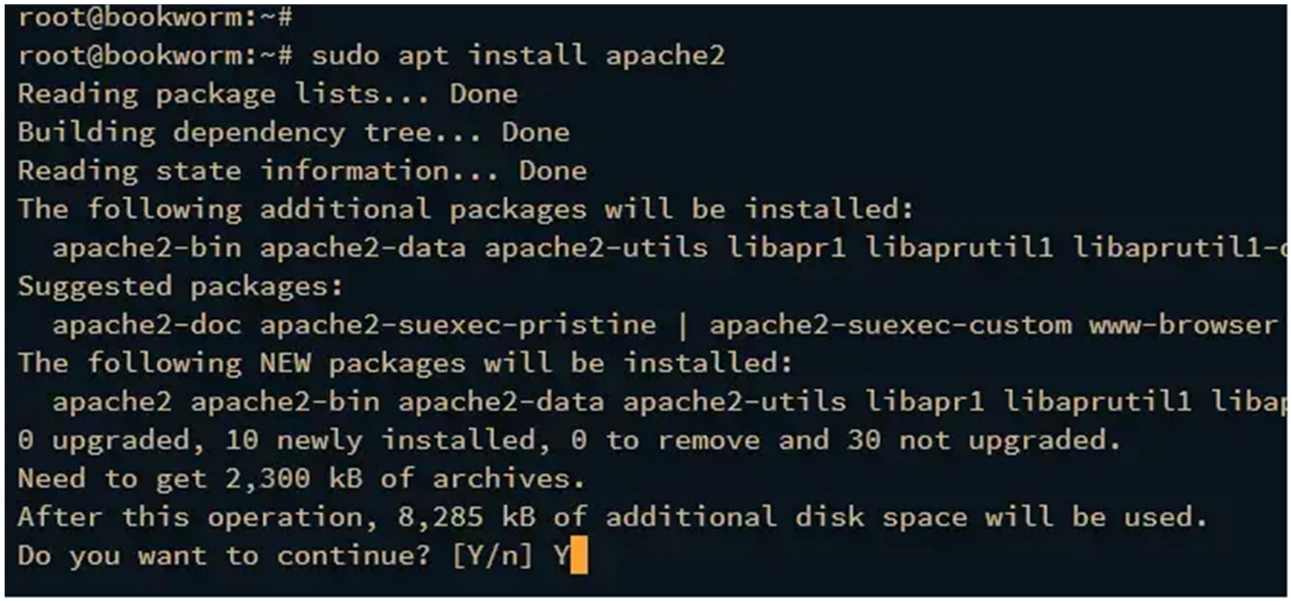

Рисунок 3.1 – Інсталяція Apache2

Після інсталяції необхідно перевірити статус служби **Apache2** (рис. 3.2).

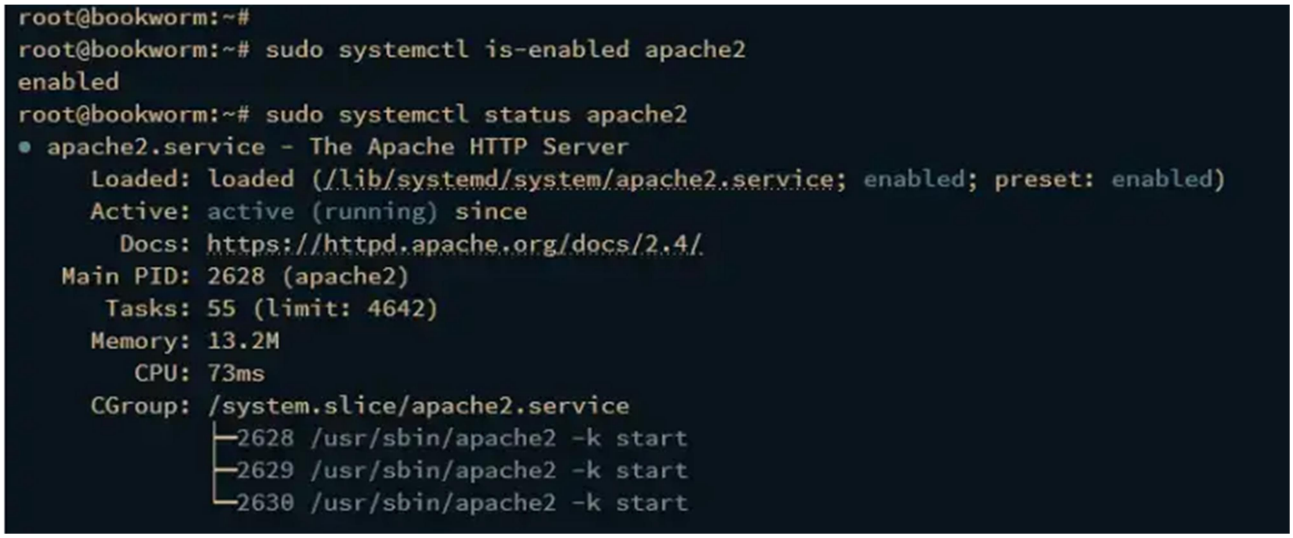

Рисунок 3.2 – Перевірка статусу Apache2

Статус **enabled** вказує, що служба Apache2 запускатиметься автоматично після запуску системи. А статус **active** підтверджує, що служба Apache2 вже запущена.

Наступним кроком буде відкриття портів OpenSSH, HTTP і HTTPS, а також інсталяція веб браузеру UFW (рис. 3.3).

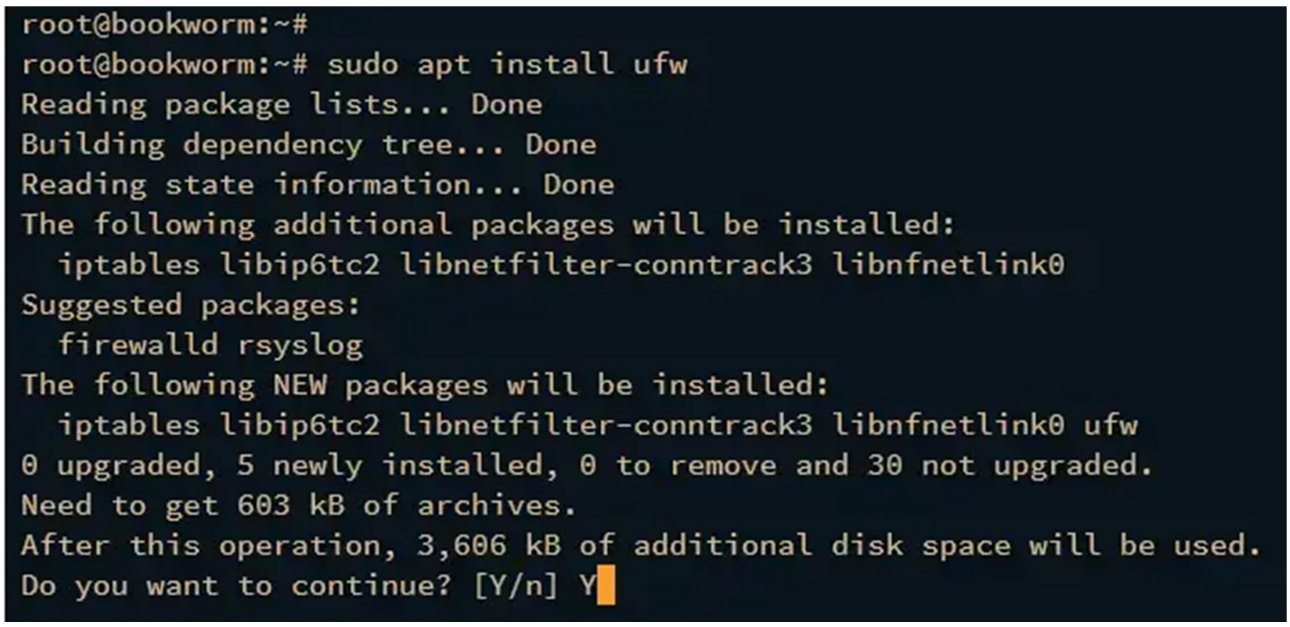

Рисунок 3.3 – Інсталяція UFW

Порт OpenSSH відкриваються завдяки службі *ufw* (рис. 3.4).

Рисунок 3.4 – Відкриття OpenSSH

Після виконання зазначених команд система прозвітує, що брандмауер активний і вмикатиметься під час запуску системи.

При своїй роботі сервер **Apache2** буде використовувати порти HTTP і HTTPS. Для їх ввімкнення необхідно виконати наступну послідовність команд:

sudo ufw app list

sudo ufw allow "WWW Full"

sudo ufw reload

Виконавши які за допомогою команди *sudo ufw status*, ми зможемо переконатися, що все виконано вірно (рис. 3.5).

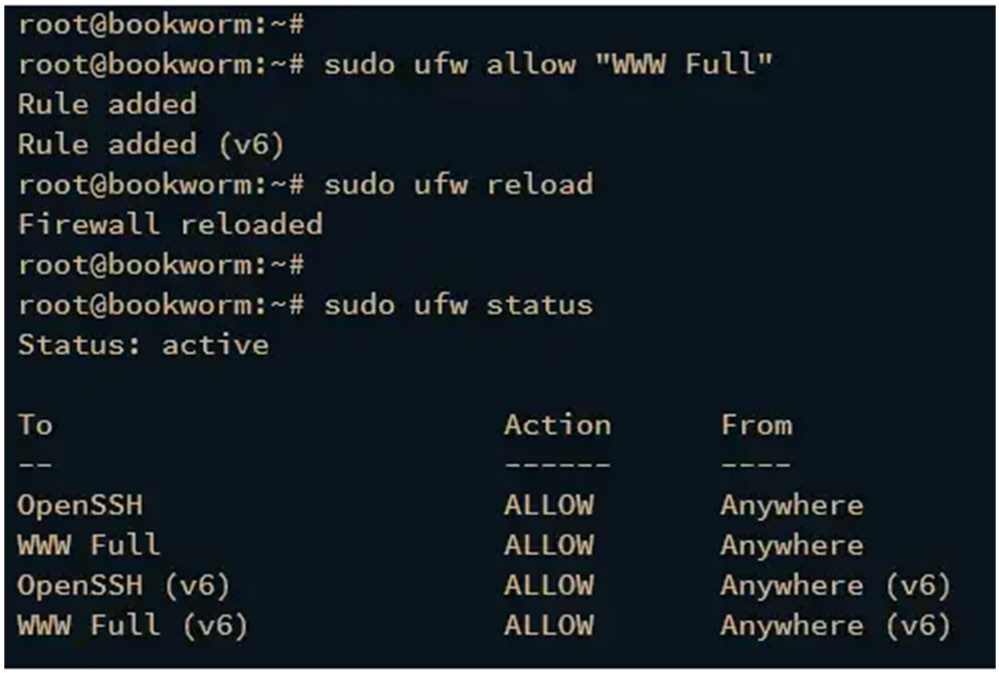

Рисунок 3.5 – Перевірка статусу портів

Як зазначалося спочатку, наступний крок полягає в установці PHP 7.0 і вище, проте у Debian 12 пакет PHP 8.2 знаходиться за замовчуванням, тобто версія PHP відповідає рекомендованій для встановлення Nextcloud. Для підтвердження цього можна перевірити поточну версію за допомогою команд:

php --version

php -m

Відповідно після їх виконання побачимо поточну версію PHP та модулі, що в неї входять (рис. 3.6).

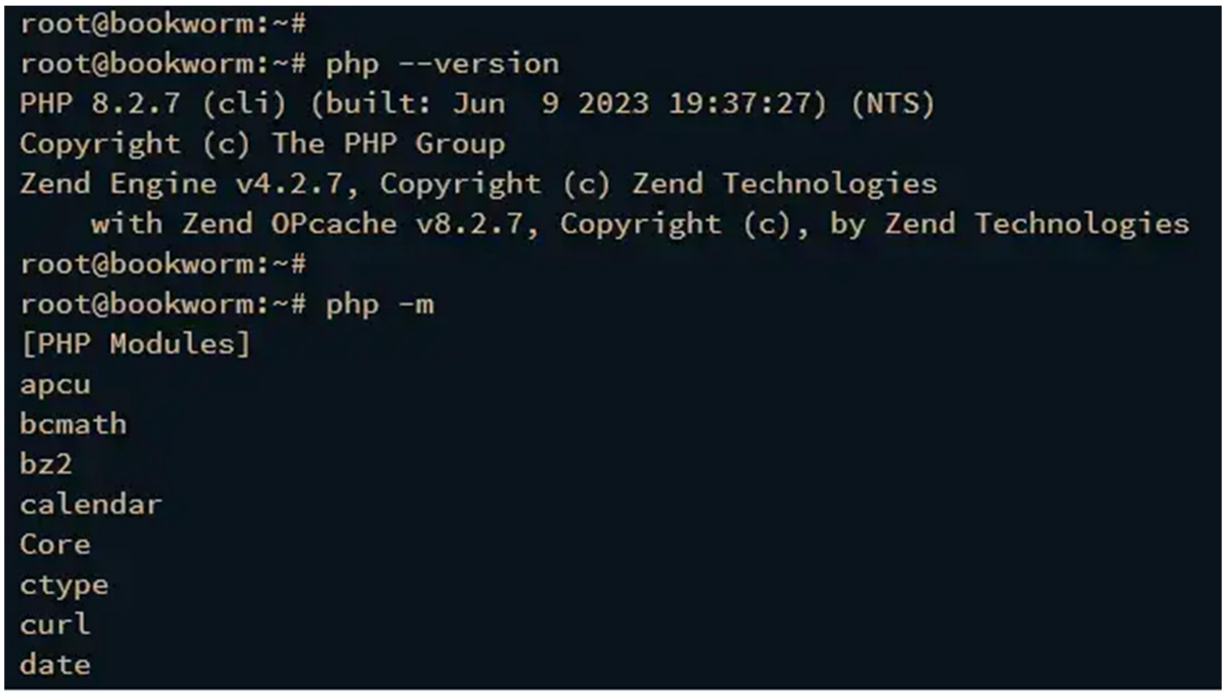

Рисунок 3.6 – Перевірка версії та модулів PHP

Для правильної роботи серверу **Apache2** необхідно налаштувати часовий пояс та значення деяких параметрів у файлі налаштувань. Для цього необхідно ввести наступну команду:

sudo nano /etc/php/8.2/apache2/php.ini

У результаті відкриється файл конфігурації, де необхідно розкоментувати параметр *date.timezone* і вибрати відповідний часовий пояс для PHP:

date.timezone = Europe/Amsterdam

Також необхідно збільшити значення за замовчуванням параметрів *memory_limit, upload_max_filesize, post_max_size і max_execution_time*.

memory_limit = 512M

upload_max_filesize = 500M

post_max_size = 600M

max_execution_time = 300

Увімкнути *file_uploads* та *allow_url_fopen*, змінивши значення за умовчанням на **On**.

file_uploads = On

allow_url_fopen = On

Також вимикаємо параметри *display_errors* та *output_buffering*, змінивши значення за умовчанням на **Off** .

display_errors = Off

output_buffering = Off

Для ввімкнення **PHP OPcache** розкоментовуємо параметр *zend_extension* і змінюємо значення на *opcache* .

zend_extension=opcache

А додавши до розділу *[opcache]* наступні рядки зробимо конфігурацію рекомендованою для Nextcloud.

opcache.enable = 1

opcache.interned_strings_buffer = 8

opcache.max_accelerated_files = 10000

opcache.memory_consumption = 128

opcache.save_comments = 1

opcache.revalidate_freq = 1

Після чого зберігаємо файл закривши редактор.

Нарешті, вводимо команду *systemctl*, щоб перезапустити службу **Apache2**. Кожного разу, коли конфігурація PHP змінюється необхідно перезапускати службу **Apache2**, щоб застосувати внесені зміни.

sudo systemctl restart apache2

Наступний крок полягає в розгортанні MariaDB Server, який використовується як база даних для Nextcloud, і налаштуванні паролю для користувача MariaDB за допомогою утиліти *mariadb-secure-installation*. Для установки самого серверу треба виконати команду *sudo apt install mariadb-server*, результат інсталяції якої можна перевірити (рис. 3.7).

Рисунок 3.7 – Перевірка версії та стану MariaDB

Тепер, коли сервер MariaDB запущено, слід його захистити, і це можна зробити за допомогою утиліти *mariadb-secure-installation*. Команда *mariadb-secure-installation* допоможе налаштувати пароль користувача MariaDB і автентифікацію, а також допоможе видалити стандартну перевірку бази даних анонімного користувача за замовчуванням.

Для цього виконуємо команду *mariadb-secure-installation*, щоб захистити сервер MariaDB. В процесі інсталяції буде поставлено кілька питань. Відповідаємо на них згідно наступних пунктів:

- Натискаємо ENTER, коли буде запропоновано ввести пароль користувача MariaDB.
- Вводимо n, коли вас запитають про метод автентифікації unix_socket.
- Вводимо Y, щоб встановити новий пароль для користувача root MariaDB. Потім вводимо новий пароль.

- Вводимо Y, щоб видалити стандартного анонімного користувача з MariaDB.
- Потім знову вводимо Y, щоб вимкнути віддалений вхід для користувача root MariaDB.
- Вводимо Y, щоб видалити стандартний тест бази даних із MariaDB.
- Нарешті, вводимо Y ще раз, щоб перезавантажити привілеї таблиці та застосувати зміни.

По закінченні інсталяції MariaDB буде успішно встановлено та налаштовано. У подальшому необхідно створити базу даних для Nextcloud, для чого заходимо в MariaDB через клієнт командою *sudo mariadb -u root -p*.

Увійшовши в MariaDB, виконуємо відповідні запити, щоб створити нову базу даних Mariadb та користувача для Nextcloud.

CREATE DATABASE nextcloud_db;

CREATE USER nextclouduser@localhost IDENTIFIED BY 'StrongPassword';

GRANT ALL PRIVILEGES ON nextcloud_db. TO nextclouduser@localhost;*

FLUSH PRIVILEGES;

Після виконання команд буде створено нову базу даних *nextcloud_db* і користувача *nextclouduser* із паролем *StrongPassword* (рис.3.8).

```
MariaDB [(none)]> CREATE DATABASE nextcloud_db;
EATEQuery OK, 1 row affected (0.001 sec)

MariaDB [(none)]> CREATE USER nextclouduser@localhost IDENTIFIED BY 'StrongPassword';
Query OK, 0 rows affected (0.002 sec)

MariaDB [(none)]> GRANT ALL PRIVILEGES ON nextcloud_db.* TO nextclouduser@localhost;
Query OK, 0 rows affected (0.002 sec)

MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.001 sec)
```

Рисунок 3.8 – Створення нової бази даних

А запустивши запит, *SHOW GRANTS FOR nextclouduser@localhost* переконаємося, що користувач *nextclouduser* може отримати доступ до бази даних *nextcloud_db* (рис. 3.9).

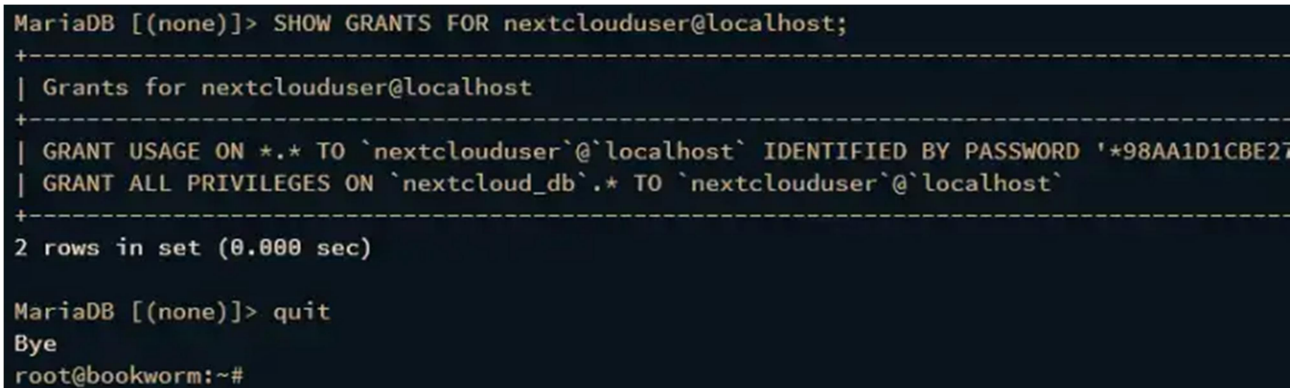

Рисунок 3.9 – Перевірка користувача *nextclouduser*

А цьому етапі проведено інсталяцію всіх необхідних складових для розгортання Nextcloud.

3.3. Розгортання персонального приватного сховища Nextcloud

Для перевірки доступності інсталювати Nextcloud необхідно запустити команду *sudo apt install curl unzip -y*, в результаті чого буде встановлено відповідний дистрибутив, та перейшовши в каталог */var/www* завантажити вихідний код Nextcloud відповідною командою (рис. 3.10).

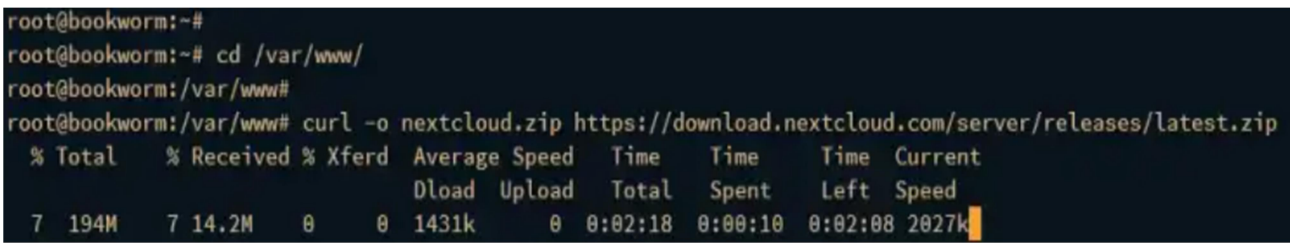

Рисунок 3.10 – Завантаження *nextclouduser*

Після чого розпаковуємо файл *nextcloud.zip* за допомогою команди *unzip*, а потім змінюємо право власності на каталог *nextcloud* на *www-data* наступними командами:

unzip nextcloud.zip

sudo chown -R www-data:www-data nextcloud

При цьому, кореневим каталогом для інсталяції **Nextcloud** є каталог */var/www/nextcloud*. А веб-сервер **Apache2** може отримати доступ до вихідного коду nextcloud через дані користувача *www.*

Наступним етапом буде завантаження вихідного коду Nextcloud. Необхідно створити нову конфігурацію **Apache2**, яка використовуватиметься для запуску **Nextcloud**, а доменне ім'я повинно співпадати з IP-адресою сервера Debian.

За допомогою команди *sudo nano /etc/apache2/sites-available/nextcloud.conf* створюємо нову конфігурацію хосту **Apache2.** Вносимо в нього наступний код:

```
<VirtualHost *:80>
   ServerName nextcloud.hwdomain.io
   DocumentRoot /var/www/nextcloud/
   # log files
   ErrorLog /var/log/apache2/files.hwdomain.io-error.log
   CustomLog /var/log/apache2/files.hwdomain.io-access.log combined
   <Directory /var/www/nextcloud/>
      Options +FollowSymlinks
      AllowOverride All
      <IfModule mod_dav.c>
         Dav off
      </IfModule>
      SetEnv HOME /var/www/nextcloud
      SetEnv HTTP_HOME /var/www/nextcloud
   </Directory>
</VirtualHost>
```

Після чого зберігаємо файл та виходимо з нього та вмикаємо конфігурацію за допомогою команди *a2ensite* (рис. 3. 11).

Сповіщення *Syntax OK* повідомить, що конфігурація введена правильно та веб-сервер **Apache2** працює, застосувавши команду systemctl, здійснимо перезавантаження поточної конфігурації, щоб зміни вступили в силу.

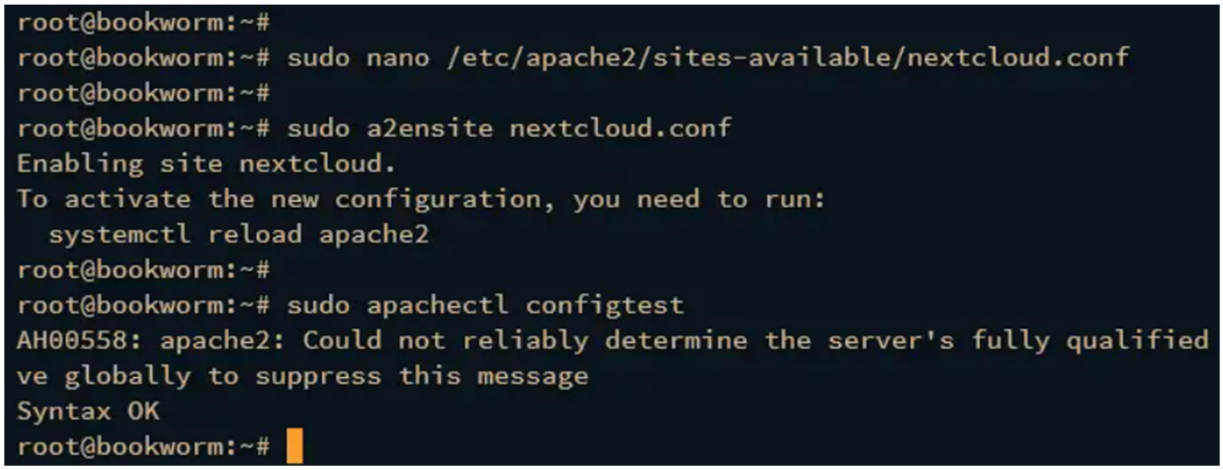

Рисунок 3.11 – Вмикання конфігурації

В результаті таких дій стане доступний вхід на веб-хост через протокол HTTP. Оскільки нам відомо, що він є не захищеним, то необхідно здійснити певну послідовність дій для налаштування HTTPS та сертифікатів SSL/TLS.

Для захисту веб-хоста пропонується застосувати інструмент Certbot, який дає змогу генерувати безкоштовні сертифікати SSL/TLS для кількох веб-серверів.

Необхідно встановити Certbot і плагін Certbot apache за допомогою команди *sudo apt install certbot python3-certbot-apache*. У результаті ми отримаємо сам бот та необхідні плагіни (рис. 3.12).

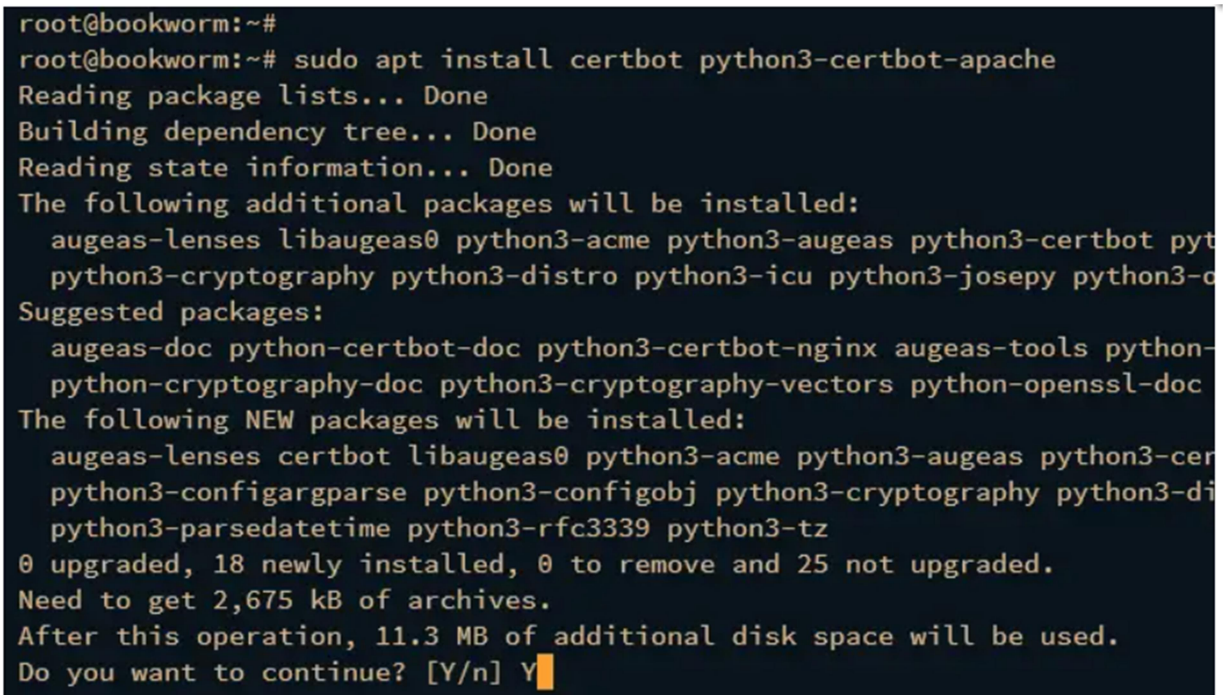

Рисунок 3.12 – Інсталяція Certbot

Продовживши роботу, виконуємо команду *sudo certbot --apache2 --agree-tos --redirect --hsts --staple-ocsp --email user@hwdomain.io -d nextcloud.hwdomain.io*, що дасть змогу в подальшому через доменне ім'я заходити на веб-інтерфейс Nextcloud застосовуючи протокол HTTPS та сертифікати SSL/TLS. Самі сертифікати будуть зберігатися у папці /etc/letsencrypt/live/*nextcloud.*

В подальшому необхідно здійснити вхід через веб-інтерфейс Nextcloud та налаштувати нового користувача адміністратора (рис. 3.13).

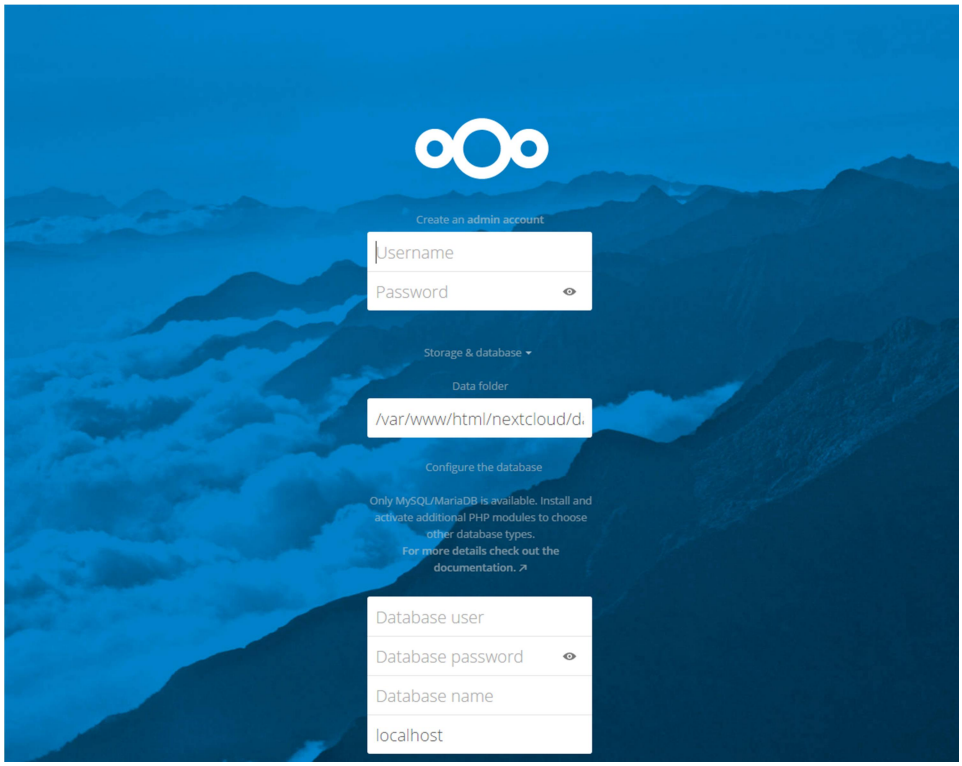

Рисунок 3.13 – Веб-інтерфейс Nextcloud

Окрім цього необхідно ввести логін та пароль користувача та назву бази даних, що була створена раніше, після чого отримаємо вікно входу в клієнт та пропозицію підключити синхронізацію зовнішніх пристроїв (рис. 3. 14).

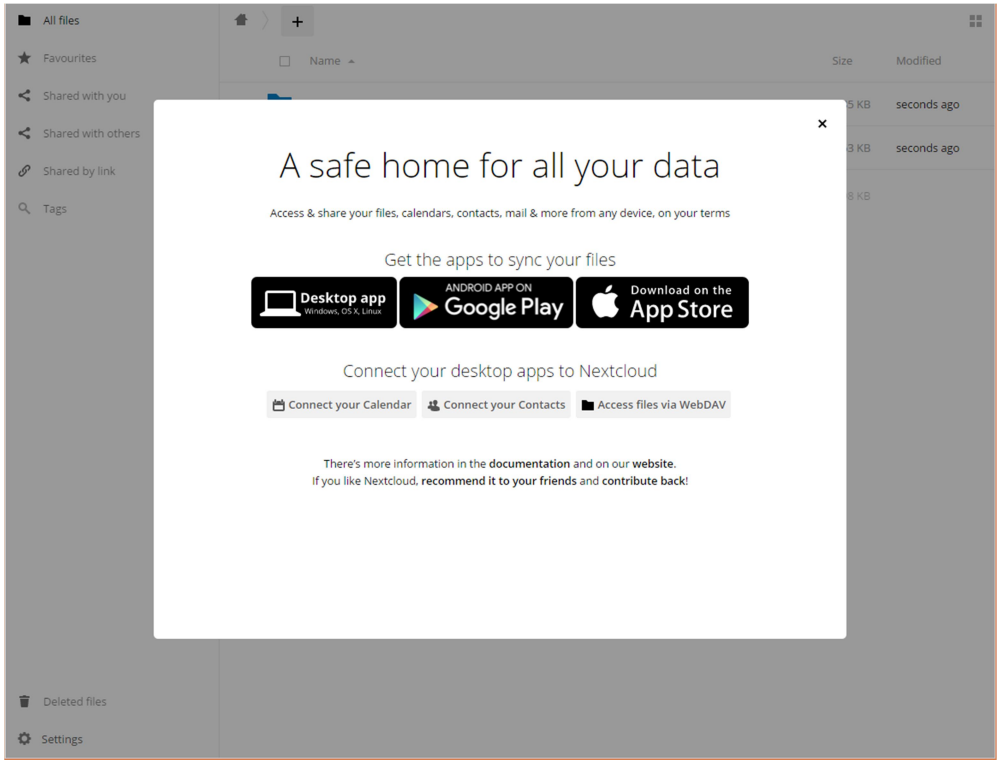

Рисунок 3.14 – Підключення зовнішніх пристроїв

Виконавши поточні налаштування та перейшовши з HTTP на HTTPS, наступний крок підвищення ефективності нашої системи полягатиме в увімкненні шифрування на боці серверу.

Це гарантуватиме, що якщо хтось отримає доступ до даних, розміщених на сервері, вміст файлів буде недоступним для читання (рис. 3.15).

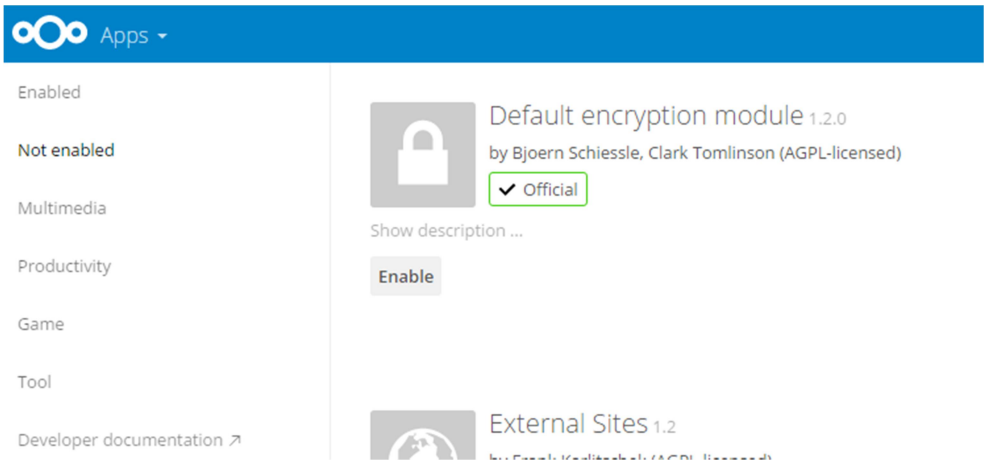

Рисунок 3.15 – Увімкнення модуля шифрування

Для увімкнення шифрування на босі серверу необхідно зайти у **Files** далі в **Apps**, а там вибрати **Not enabled** у лівому меню та під **Default encryption module** клацнути **Enable**. Далі необхідно перейти в меню адміністрування та увімкнути саме шифрування (рис. 3.16).

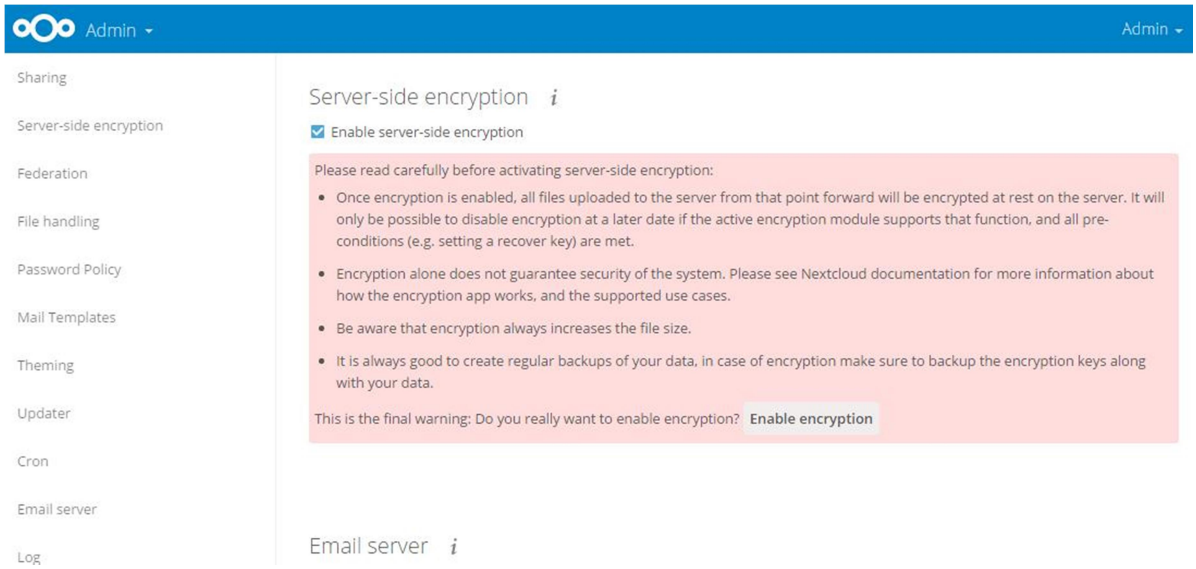

Рисунок 3.16 – Увімкнення шифрування на боці серверу

Для того, щоб зміни вступили в силу, необхідно повторно зайти в систему, після чого появиться можливість створити ключ відновлення (рис. 3.17).

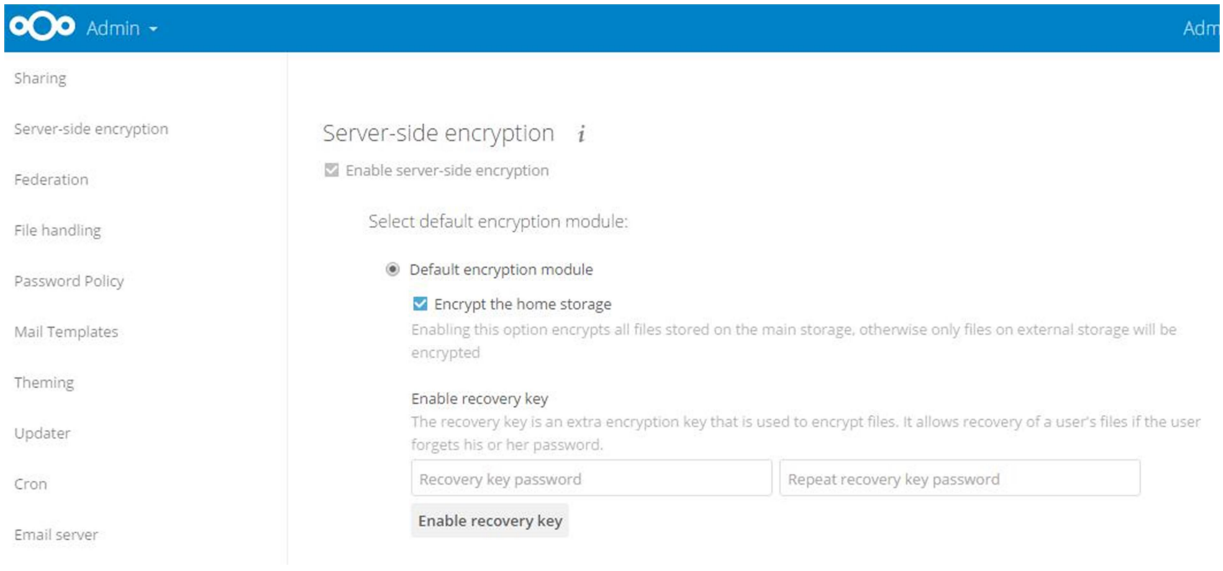

Рисунок 3.17 – Створення ключа відновлення

Проте, бувають випадки, коли один спільний ключ відновлення є недоцільним, наприклад коли є декілька користувачів кожен з яких має доступ до окремих конфіденційних даних. у такому випадку Nextcloud пропонує відновлювати дані по паролю, який буде унікальним для кожного з користувачів. Для увімкнення такої функції користувач клацнувши по ніку в правому кутку повинен вибрати опцію **Enabled** у вкладці **Personal** (рис. 3.18).

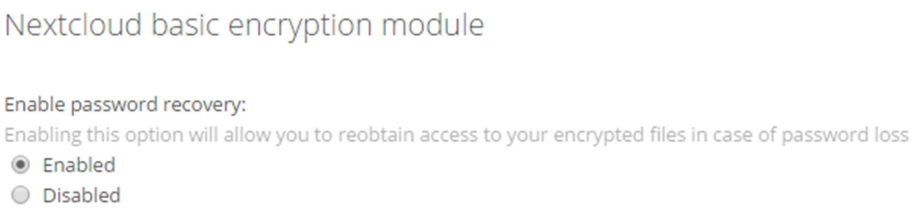

Рисунок 3.17 – Створення персонального ключа відновлення

Після виконання всіх цих налаштувань дані кожного з користувачів системи будуть захищені при зберіганні та передачі за допомогою сертифікатів SSL.

ВИСНОВКИ

Хмарні середовища є одним із останніх трендів у сфері IT і надає різноманітні переваги клієнтам. Хмарна інформаційна безпека є однією з головних проблем будь-якої організації, яка розглядає можливість переходу на хмару. Шифрування є одним із найбезпечніших рішень для блокування несанкціонованого доступу. У хмарних середовищах використовуються різні методи шифрування для захисту хмарних даних, що певною мірою сприяє зниженню рівня злому.

У результаті виконання кваліфікаційної роботи досягнуто наступних результатів:

1. Проведено аналіз проблем безпеки хмари, викликів впровадження хмари та алгоритмів шифрування, які використовуються в хмарних середовищах.

2. Було проведено аналіз літератури в галузі безпеки хмарних даних, у результаті якого проведено порівняння поширених алгоритми шифрування, а саме RSA, AES, DES, Blowfish і IDEA. Результати показують, що RSA та IDEA менш безпечні, ніж AES, Blowfish і DES, а алгоритм Blowfish вимагає найменшого обсягу пам’яті. Алгоритм AES можна використовувати для шифрування величезних обсягів даних. AES є швидшим за інші алгоритми та є найкращим алгоритмом з точки зору параметрів автентифікації. RSA споживає найбільше пам’яті та потребує максимального часу виконання.

3. Визначено ключові аспекти, що впливають на довіру до хмарних технологій в цілому та хмарних обчислень зокрема.

4. Проведено порівняльний аналіз поширених хмарних сховищ, у результаті якого обґрунтовано використання хмарних сховищ з такими методами захисту як захист з наскрізним шифруванням та з нульовим знанням.

5. Розгорнуто приватне хмарне середовище Nextcloud, що забезпечує наскрізне шифрування даних та відповідає заявленим вимогам довіри користувачів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Mondal, A., Paul, S., Goswami, R.T., Nath, S. (2020). Cloud computing security issues & challenges: A review. *In Proceedings of the 2020 International Conference on Computer Communication and Informatics*, 22–24 January, 1–5.
2. Narayan, S.; Gagné, M.; Safavi-Naini, R. (2010). Privacy preserving EHR system using attribute-based infrastructure. *In Proceedings of the 2010 ACM workshop on Cloud Computing Security Workshop*, Chicago, IL, USA, October, 47–52.
3. Grover, A.; Kaur, B. (2016). A framework for cloud data security. *In Proceedings of the Computing, Communication and Automation*, India, 1199–1203.
4. Khanezaei, N.; Hanapi, Z.M. (2014). A framework based on RSA and AES encryption algorithms for cloud computing services. *In Proceedings of the Systems, Process and Control (ICSPC)*, Malaysia, 12–14 December, 58–62.
5. Abha, S., Bhanali, M. (2013). Cloud computing security using AES algorithm. *Int. J. Comput. Appl.*, *67*, 19–23.
6. Zhao, F.; Li, C.; Liu, C.F. A cloud computing security solution based on fully homomorphic encryption. In Proceedings of the 16th International Conference on Advanced Communication Technology (ICACT), Pyeongchang, Korea, 16–19 February 2014; pp. 485–488.
7. Kaur, R.; Singh, R.P. (2014). Enhanced cloud computing security and integrity verification via novel encryption techniques. *In Proceedings of the International Conference on Advances in Computing, Communications and Informatics*, India, 24–27 September, 1227–1233.
8. Rani, S.; Gangal, A. (2012). Cloud security with encryption using hybrid algorithm and secured endpoints. *Int. J. Comput. Sci. Inf. Technol.*, *3*, 4302–4304.
9. Gai, K.; Qiu, M., Zhao, H., Xiong, J. (2016). Privacy-aware adaptive data encryption strategy of big data in cloud computing. *In Proceedings of the 2016*

IEEE 3rd International Conference on Cyber Security and Cloud Computing China, 273–278.

10. Mohamed, M.E.; Abdelkader, H.S.; El-Etriby, S. (2012). Enhanced data security model for cloud computing. *In Proceedings of the 8th International Conference on Informatics and Systems (INFOS)*, Egypt, 14–16 May, 12–17.
11. Kumar, D.A.; Dubey, A.K.; Namdev, M.; Shrivastava, S.S. (2012). Cloud-user security based on RSA and MD5 algorithm for resource attestation and sharing in java environment. *In Proceedings of the 2012 CSI Sixth International Conference on Software Engineering (CONSEG)*, India, 5–7 September, 1–8.
12. Uma, S.; Lakhani, K.; Mundra, M. Implementing digital signature with RSA encryption algorithm to enhance the data security of cloud in cloud computing. In Proceedings of the 1st International Conference on Parallel Distributed and Grid Computing (PDGC), Solan, India, 28–30 October 2010; pp. 211–216.
13. Li, Y.; Gai, K.; Qiu, L.; Zhao, H. Intelligent cryptography approach for secure distributed big data storage in cloud computing. *Inf. Sci.* 2017, *387*, 103–115.
14. Pradeep, K.V.; Vijayakumar, V.; Subramaniyaswamy, V. An efficient framework for sharing a file in a secure manner using asymmetric key distribution management in cloud environment. *J. Comput. Netw. Commun.* 2019, *2019*, 9852472.
15. Khan, S.S.; Tuteja, R.R. Data security in cloud computing using cryptographic algorithms. *Int. J. Innov. Res. Comput. Commun. Eng.* 2019, *7*, 1.
16. Manpreet, K.; Singh, R. Implementing encryption algorithms to enhance data security of cloud in cloud computing. *Int. J. Comput. Appl.* 2013, *70*, 18.
17. Poteya, M.; Dhoteb, A.; Sharmac, H. Homomorphic encryption for security of cloud data. *Procedia Comput. Sci.* 2016, *79*, 175–181.
18. Kartit, Z.; Azougaghe, A.; Kamal Idrissi, H.; Marraki, M.E.; Hedabou, M.; Belkasmi, M.; Kartit, A. Applying encryption algorithm for data security in cloud storage. *Adv. Ubiquitous Netw. Lect. Notes Electr. Eng.* 2015, *366*, 141–154.

19. Salama, D. Improving the security of cloud computing by building new hybrid cryptography algorithms. *Int. J. Electron. Inf. Eng.* 2018, *8*, 40–48.
20. Mondal, A.; Paul, S.; Goswami, R.T.; Nath, S. Cloud computing security issues & challenges: A review. In Proceedings of the 2020 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 22–24 January 2020; pp. 1–5.
21. Narayan, S.; Gagné, M.; Safavi-Naini, R. Privacy preserving EHR system using attribute-based infrastructure. In Proceedings of the 2010 ACM workshop on Cloud Computing Security Workshop, Chicago, IL, USA, 8 October 2010; pp. 47–52.
22. Grover, A.; Kaur, B. A framework for cloud data security. In Proceedings of the Computing, Communication and Automation (ICCCA), Greater Noida, India, 29–30 April 2016; pp. 1199–1203.
23. Khanezaei, N.; Hanapi, Z.M. A framework based on RSA and AES encryption algorithms for cloud computing services. In Proceedings of the Systems, Process and Control (ICSPC), Kuala Lumpur, Malaysia, 12–14 December 2014; pp. 58–62.
24. Abha, S.; Bhanali, M. Cloud computing security using AES algorithm. *Int. J. Comput. Appl.* 2013, *67*, 19–23.
25. Zhao, F.; Li, C.; Liu, C.F. A cloud computing security solution based on fully homomorphic encryption. In Proceedings of the 16th International Conference on Advanced Communication Technology (ICACT), Pyeongchang, Korea, 16–19 February 2014; pp. 485–488.
26. Gururaj, R.; Mohsin, I.; Farrukh, K. A comprehensive survey on security in cloud computing. *Procedia Comput. Sci.* 2017, *110*, 465–472.
27. Khalil, A.; Faiz, A.; Mohammad, H.; Hassen, F. Cloud computing security challenges in higher educational institutions—A survey. *Int. J. Comput. Appl.* 2017, *161*, 22–29.

28. Mahmud, R.; Srirama, S.N.; Ramamohanarao, K.; Buyya, R. Profit-aware application placement for integrated fog–cloud computing environments. *J. Parallel Distrib. Comput.* 2020, *135*, 177–190.
29. Hussain, S.; Ullah, S.S.; Uddin, M.; Iqbal, J.; Chen, C.-L. A Comprehensive Survey on Signcryption Security Mechanisms in Wireless Body Area Networks. *Sensors* 2022, *22*, 1072.
30. Uddin, M.; Khalique, A.; Jumani, A.K.; Ullah, S.S.; Hussain, S. Next-Generation Blockchain-Enabled Virtualized Cloud Security Solutions: Review and Open Challenges. *Electronics* 2021, *10*, 2493.
31. Yanes, A.R.; Martinez, P.; Ahmad, R. Towards automated aquaponics: A review on monitoring, IoT, and smart systems. *J. Clean. Prod.* 2020, *263*, 121571.
32. Ma, L.; Wang, X.; Wang, X.; Wang, L.; Shi, Y.; Huang, M. TCDA: Truthful combinatorial double auctions for mobile edge computing in industrial Internet of Things. *IEEE Trans. Mob. Comput.* 2021.
33. Munoz, A.; Mana, A.; González, J. Dynamic Security Properties Monitoring Architecture for Cloud Computing. In *Security Engineering for Cloud Computing*; IGI Global: Hershey, PA, USA, 2013; pp. 1–18.
34. Lopez, J.; Mana, A.; Munoz, A. A Secure and Auto-configurable Environment for Mobile Agents in Ubiquitous Computing Scenarios. In Proceedings of the Third International Conference on Ubiquitous Intelligence and Computing, Wuhan, China, 3–6 September 2006; Volume 4159, pp. 977–987.
35. Hussain, S.; Ullah, I.; Khattak, H.; Adnan, M.; Kumari, S.; Ullah, S.S.; Khan, M.A.; Khattak, S.J. A lightweight and formally secure certificate based signcryption with proxy re-encryption (CBSRE) for Internet of Things enabled smart grid. *IEEE Access* 2020, *8*, 93230–93248.

ГРОМАДСЬКЕ ОБЄДНАННЯ
«КІБЕРБЕЗПЕКА І АВТОМАТИЗАЦІЯ»

Матеріали
науково-практичного симпозіуму
«ЗАХИСТ ІНФОРМАЦІЇ»

30 листопада 2023
Тернопіль

У збірнику опубліковано матеріали науково-практичного симпозіуму «Захист інформації», Тернопіль, 2023. - 194 с.

Редактор коректор: Гуменний П.В.
Технічний редактор: Давлетова А.Я.

Адреса редакції:
Громадське об'єднання «Кібербезпека і автоматизація»
м. Тернопіль
Контактний телефон: (066)043-42-10
e-mail: enmkd.scs@gmail.com

УДК 004.056

МЕЛЬНИК П.

Західноукраїнський національний університет

АСПЕКТИ БЕЗПЕКИ ОБРОБКИ ДАНИХ У ХМАРНИХ СХОВИЩАХ

Вступ. З розвитком Інтернету речей (IoT) кількість пристроїв сприйняття інформації, підключених до Інтернету, зростає, щоб реалізувати взаємозв'язок між людьми, пристроями та «речами».

Згідно з новим прогнозом IDC [1], у 2025 році буде 41,6 мільярда пристроїв або «речей» Інтернету речей, які генеруватимуть 79,4 зетабайта (ZB) даних. Мало того, люди все ще прагнуть покращити ефективність збору даних пристроїв в IoT. Безпрецедентна кількість даних генерується та розміщується на платформі провайдера хмарних послуг. Таким чином, мешканці розумного міста та постачальники послуг можуть покладатися на хмарні послуги для розміщення, створення та/або розгортання своїх служб і програм розумного міста. Крім того, перевага оплати за використання спонукає більшість традиційних підприємств активно переносити дані в хмару.

Метою роботи є дослідження принципів та методів захисту даних у хмарних сховищах та при хмарних обчисленнях.

1. Дослідження принципів реалізації хмарних обчислень та їх типів

Хмара – це не тільки місце призначення робочого навантаження, але й забезпечує ефективну практику роботи, що робить підприємства більш гнучкими та гнучкими. Це сприяло як цифровій трансформації підприємств, так і трансформації модернізації мережі. У звіті ООН про цифрову економіку за 2019 рік підкреслюється, що цифрова економіка стає важливою рушійною силою економічного розвитку. Згідно з неповною статистикою, на цифрову економіку припадає від 4,5% до 15,5% світового ВВП [2]. Хмарні обчислення сприяють глибокій інтеграції Інтернету, великих даних, штучного інтелекту та реальної економіки, а також є основою прискорення побудови сучасної економічної системи.

Хмарне сховище – це, по суті, система хмарних обчислень, яка дозволяє користувачам зберігати та обмінюватися даними в Інтернеті. Переваги хмарного сховища включають необмежений простір для зберігання даних, зручний, безпечний і ефективний доступ до файлів і резервне копіювання за межами сайту, а також низьку вартість використання. Як новий режим хмарного зберігання в останні роки хмара спільноти дуже підходить для медичної та фінансової промисловості.

Хмара спільноти надає хмарні послуги для кількох компаній у певній спільноті. Зазвичай ці підприємства мають однакові проблеми або потребують спільної роботи над деякими проектами. Побудова інфраструктури та керування сервером можуть спільно здійснюватися членами спільноти Cloud або передаватися третій стороні.

З точки зору архітектури зберігання, основні хмарні платформи зазвичай пропонують три широкі класи сховища: блочне сховище, сховище файлів і сховище об'єктів.

- Хмарне блочне сховище, яке поважає Storage Area Networks (SAN), по

суті, забезпечує віртуалізовану мережу Storage Area Network із забезпеченням керування логічним томом через спрощений інтерфейс веб-служб.

– Зберігання файлів, яке також називають сховищем на рівні файлу або сховищем на основі файлів, зазвичай асоціюється з технологією мережевого сховища (NAS).

Завдяки файловій системі файлове сховище керує даними спільного використання та доступом до даних, що зберігаються в ньому, більш гнучко, ніж блокове сховище. Масові дані створюють низку проблем для підприємств, таких як розширення сховища, спільне використання даних, ефективна передача, вартість і безпека даних, коли зберігання даних досягає рівня PB, обмеження NAS і SAN безпосередньо призводить до збільшення вартості обслуговування обладнання. в більш пізній період. Вони не можуть повністю задовольнити вимоги підприємства щодо надійності, доступності, безпеки та інших показників масового зберігання даних у цьому сховищі об'єкта.

Хмарне сховище базується на інфраструктурі віртуалізації та схоже на хмарні обчислення з точки зору доступних інтерфейсів, масштабованості та ресурсів вимірювання.

Воно складається з чотирьох рівнів [3], які можна подати таким чином:

1) Рівень зберігання, основна частина хмарного сховища, складається з пристроїв зберігання та уніфікованого керування пристроями зберігання даних.

2) Первинний рівень керування є основною частиною хмарного сховища, а також найскладнішою частиною хмарного сховища.

3) Рівень інтерфейсу програми є найбільш гнучкою частиною хмарного сховища.

4) Останнім є рівень доступу.

З цієї точки зору хмарне сховище надає послуги доступу до даних, включаючи зберігання даних, обчислення даних, автентифікацію та контроль доступу. Через особливості хмарного сховища в цьому процесі неминуче виникають проблеми з безпекою даних і конфіденційністю.

Вимоги безпеки даних у хмарних сховищах в основному відображені в наступних аспектах [4]:

– Конфіденційність даних: конфіденційність даних стосується запобігання активним атакам неавторизованих сторін на дані користувачів і забезпечення повної відповідності інформації, отриманої одержувачем даних, з інформацією, надісланою відправником. Це означає, що лише уповноважені особи мають право доступу та отримання даних. Уявіть свій банківський рахунок. Ви, звичайно, повинні мати до них доступ, і працівники банку, які допомагають вам з транзакцією, повинні мати до них доступ, але ніхто інший не повинен. Після доступу інших осіб конфіденційність даних порушується, що є незворотнім.

– Цілісність даних: цілісність даних — це надійність даних, тобто дані не можуть бути довільно підроблені та замінені. Наприклад, якщо ви робите покупки онлайн на Amazon, хтось може змінити товари у вашому кошику без вашого дозволу. Відсутність цілісності даних може створити серйозні проблеми з безпекою.

– Доступність даних: доступність даних підкреслює, що до них можна

отримати звичайний доступ у будь-який час, а саме: користувач може отримувати доступ, завантажувати або вносити деякі зміни в дані в хмарі, як тільки їм це буде потрібно.

- Точний контроль доступу.
- Безпечний обмін даними в динамічній групі.
- Стійкий до протікання.
- Повне видалення даних: коли користувачі більше не користуються хмарним сховищем, вони можуть повністю видалити дані, передані на хмарний сервер, і підтвердити, що дані були повністю знищені, замість того, щоб бути ошуканими зловмисними постачальниками хмарних послуг.
- Захист конфіденційності: хоча користувачі насолоджуються зручністю хмарного сховища, постачальники хмарного сховища зберігають їхню конфіденційну інформацію, таку як особисті дані, місцезнаходження та конфіденційні дані для підприємства. Механізми захисту конфіденційності використовуються, щоб гарантувати, що ці дані залишаються секретними від цікавих противників і зловмисних співробітників постачальників хмарних послуг.

З подальшою централізацією даних і збільшенням їх обсягу захист даних у хмарному сховищі стає проблематичним. Таким чином, питання про те, як гарантувати, що користувачі та їхні інформаційні ресурси не будуть розкриті, ще довгий час буде головною проблемою постачальників хмарних послуг і науковців.

Висновки

Отже, безпека даних і збереження конфіденційності в системі хмарного зберігання в основному стикаються з такими проблемами:

- Точне керування доступом до даних.
- Зловмисні постачальники хмарних послуг можуть повертати неправильні дані.
- Результати аудиту доброчесності.
- Атака бічного каналу.
- Зловмисні постачальники хмарних послуг не дотримуються запитів клієнтів щодо повного видалення даних у хмара.
- Збереження конфіденційності.

Хоча хмарне сховище розвивалося протягом багатьох років, воно все ще дуже важливо в Інтернеті речей, розумному місті та цифровій економіці. Безпека даних і захист конфіденційності в хмарних сховищах все ще мають значення.

Перелік використаних джерел

1. Mondal, A., Paul, S., Goswami, R.T., Nath, S. (2020). Cloud computing security issues & challenges: A review. In Proceedings of the 2020 International Conference on Computer Communication and Informatics, 22–24 January, 1–5.

2. Zhao, F.; Li, C.; Liu, C.F. A cloud computing security solution based on fully homomorphic encryption. In Proceedings of the 16th International Conference on Advanced Communication Technology (ICACT), Pyeongchang, Korea, 16–19 February 2014; pp. 485–488.

3. Kumar, D.A.; Dubey, A.K.; Namdev, M.; Shrivastava, S.S. (2012). Cloud-user security based on RSA and MD5 algorithm for resource attestation and sharing in java environment. In Proceedings of the 2012 CSI Sixth International Conference on Software Engineering (CONSEG), India, 5–7 September, 1–8.

4. Rani, S.; Gangal, A. (2012). Cloud security with encryption using hybrid algorithm and secured endpoints. Int. J. Comput. Sci. Inf. Technol., 3, 4302–4304.

УДК 004.056.5

МОТРОНЮК.Н.Б.

Західноукраїнський національний університет

АРХІТЕКТУРА ТА АЛГОРИТМ РОБОТИ ТЕЛЕГРАМ-БОТА

Вступ. У сучасному цифровому світі питання кібербезпеки стає все більш актуальним та вимагає постійного вдосконалення заходів захисту. Одним із важливих аспектів цього питання є вчасне та ефективне виявлення можливих загроз та компрометаційних подій. У рамках цього дослідження розглядається питання розробки та впровадження телеграм-бота, спрямованого на автоматизований аналіз файлів для виявлення потенційно шкідливих об'єктів [1].

Архітектура та розробка такого бота є об'єктом ретельного аналізу, оскільки вони визначають ефективність та функціональні можливості системи. Передові технології програмування, інтеграція зі службами сканування, засоби оптимізації та заходи безпеки – усі ці аспекти відіграють ключову роль у створенні надійного та ефективного інструменту для аналізу файлового середовища.

Мета. Мета даної роботи полягає у створенні та аналізі архітектури та реалізації телеграм-бота, який забезпечить користувачів засобами безпеки при обміні та аналізі файлів, а також у вивченні можливостей оптимізації та захисту цього інструменту від кіберзагроз.

1. Огляд архітектури телеграм-бота

Архітектура телеграм-бота є ключовим елементом, що визначає його ефективність та можливості у виявленні компрометації файлів. Система складається з кількох важливих компонентів, які спільно працюють для досягнення основних цілей дослідження [2].

Користувацький інтерфейс. Користувачі взаємодіють з ботом через зручний інтерфейс чату в месенджері Telegram. Цей інтерфейс дозволяє надсилати файли для аналізу та отримувати інформацію щодо їхньої безпеки. Інтуїтивний та простий для використання інтерфейс робить взаємодію з ботом легкою та зрозумілою для широкого кола користувачів.

Модуль аналізу файлів. Цей ключовий компонент відповідає за проведення аналізу отриманих файлів з метою виявлення можливих загроз. Використовуючи сучасні методи сканування та аналізу хеш-сум, модуль генерує детальний звіт про

ЗАХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ВОДНОГО ГОСПОДАРСТВА ТА ПРИРОДОКОРИСТУВАННЯ
ГАЛИЦЬКИЙ ФАХОВИЙ КОЛЕДЖ ІМ. В. ЧОРНОВОЛА

КІБЕРБЕЗПЕКА ТА КОМП'ЮТЕРНО-ІНТЕГРОВАНІ ТЕХНОЛОГІЇ (КБКІТ – 2023)

науково-практична конференція
молодих вчених, аспірантів та студентів

29–31 серпня 2023
Тернопіль

Збірник матеріалів науково-практичної конференції молодих вчених, аспірантів та студентів «Кібербезпека та комп'ютерно-інтегровані технології» (КБКІТ - 2023), Тернопіль, 2023. - 234 с.

Редактор коректор: Гуменний П.В.
Технічний редактор: Давлетова А.Я.

Адреса редакції:
Західноукраїнський національний університет, кафедра кібербезпеки,
вул. Олени Теліги 8, м. Тернопіль 46003
Контактний телефон: (0352) 50-17-87
e-mail: kb.tneu@gmail.com

УДК 004.9

Мельник П.

Західноукраїнський національний університет

ПРИНЦИПИ ТА МОДЕЛІ ХМАРНИХ ОБЧИСЛЕНЬ

Вступ. Технологія хмарних обчислень (СС) набула широкої популярності завдяки своїй здатності надавати величезні ресурси окремим особам і організаціям, до яких можна отримати доступ через Інтернет у будь-який час і в будь-якій точці світу [1, 2]. Багато інформаційних і технологічних (IT) компаній перенесли свою діяльність у хмару, яка надає своїм користувачам багатофункціональний хмарний досвід, включаючи доступ до спільних ресурсів, що робить ресурси доступними, коли вони потрібні, за меншими витратами. Ці ресурси також можуть бути швидко надані та звільнені з мінімальними адміністративними зусиллями, а СС надає можливість спільно використовувати, керувати та зберігати дані, які фактично розміщені на віддалених серверах, а не за допомогою внутрішніх ресурсів чи персональних пристроїв [1]. Клієнти можуть використовувати хмарні сервіси різних програм, прийнявши СС, а не купуючи або встановлюючи програмне забезпечення на своїх комп'ютерах [3]. СС надає клієнтам віртуалізовані ресурси за допомогою різних технологій, таких як веб-сервіси, віртуалізація, програми та операційні системи [1]. Основні переваги СС можна підсумувати як зниження витрат, підвищення продуктивності, стабільність, масштабованість, легке управління та доступність [4, 5].

Метою роботи є дослідження принципів та моделей хмарних обчислень.

1. Дослідження принципів реалізації хмарних обчислень та їх типів

СС – це метафора для опису Інтернету як місця, де обчислювальна техніка була попередньо встановлена та доступна як послуга, де дані, програми, операційні системи, сховище та потужність обробки доступні в Інтернеті та готові до спільного використання між клієнтами [2]. СС відноситься до набору центрів обробки даних, які підключаються до Інтернету, щоб пропонувати свої послуги, і ці центри обробки даних базуються на віртуалізації своєї інфраструктури [6]. СС технологічно базується на інфраструктурі, програмній платформі, операційній системі, розробці хмарних програм, управлінні базами даних, програмному забезпеченні для керування системою та програмами, Інтернеті та мережі. Постачальники послуг СС – це компанії, які надають своїм клієнтам ресурси та послуги СС, які використовуються динамічно за запитом клієнта відповідно до конкретної бізнес-моделі. На рисунку 1 показано зв'язок між найпоширенішими постачальниками послуг СС, такими як Amazon, Google, Microsoft і IBM.

СС класифікується на три типи [7]: приватна, публічна та гібридна хмара. Приватні хмари управляються та контролюються лише для окремої організації, а активи не використовуються іншими клієнтами, що вказує на те, що вони захищені від доступу неавторизованих користувачів. Публічні хмари доступні для широкої громадськості та організацій. Активи розподіляються між кожним із клієнтів. Клієнти платять власнику хмари залежно від наданої послуги та активів, які вони використовують. CSP керують фізичною інфраструктурою, яка розташована подалі від клієнтів.

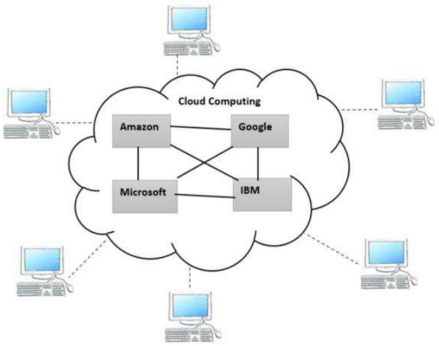

Рисунок 1 – Хмарні обчислення

Гібридні хмари є сумішшю двох вищевказаних типів (публічного та приватного) [8].

СС надає три ключові послуги, а саме:

- програмне забезпечення як послуга (SaaS);
- платформа як послуга (PaaS);
- інфраструктура як послуга (IaaS).

IaaS відноситься до апаратної інфраструктури CSP, яка включає мережі, сховище, пам'ять, процесори та низку інших обчислювальних ресурсів. Ресурси надаються як віртуалізовані системи, до яких можна отримати доступ через Інтернет. Основні ресурси знаходяться під контролем CSP [1].

PaaS забезпечує інтегроване середовище розробки, проміжне програмне забезпечення, операційні системи та ресурси рівня платформи через стороннього постачальника, який надає апаратні та програмні засоби користувачам через Інтернет. PaaS не надає клієнтам контроль над основною хмарною інфраструктурою, а лише над програмами, які переміщуються в хмару.

SaaS дозволяє споживачам використовувати програми як послугу через Інтернет. Користувачі можуть просто використовувати Інтернет для доступу до нього, а не купувати, встановлювати та підтримувати програмне забезпечення. Клієнти платять за використання, а не за право власності на програмне забезпечення.

Система СС розділена на дві частини: передню та задню частину, які спілкуються один з одним через мережу, зазвичай Інтернет. Передня частина - це сторона, яку бачать хмарні клієнти. Клієнти зазвичай не бачать внутрішнього розділу, який включає мережеве підключення, хмарні сервери та їхні програми.На рисунку 2 показано категорії хмарних сервісів і архітектуру.

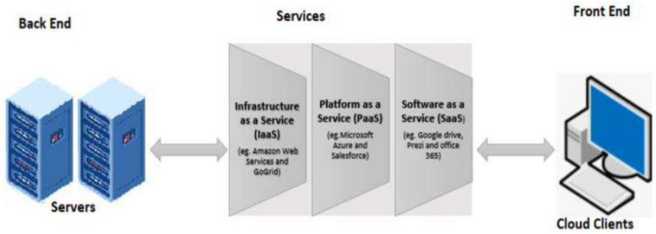

Рисунок 2 – Сервісна модель СС

Хмарні середовища задовольняють багатьом характеристикам, які [5]:

- Самообслуговування на вимогу: Хмарні служби (такі як мережеве сховище, доступ до мережі та безперервний моніторинг часу роботи сервера) не потребують жодних менеджерів. Самі клієнти можуть надавати, контролювати та маніпулювати обчислювальними ресурсами та IT-послугами за потреби.
- Об'єднання ресурсів: CSP може розподіляти витрати та ресурси CC (такі як сервери, сховище, база даних, програми, мережі та служби) серед великого пулу користувачів, що дозволяє користувачам, підключеним до хмари, використовувати дані одночасно та спільно використовувати хмару послуги відповідно до їхніх вимог.
- Широкий доступ до мережі: користувач може отримати доступ до ресурсів CC через мережу з будь-якої точки світу за допомогою підключення до Інтернету та пристрою (наприклад, смартфона, комп'ютера та КПК).
- Швидка еластичність: обчислювальні послуги та ресурси можна швидко та гнучко збільшувати чи зменшувати за потреби.
- Економія: CC зменшує величезні витрати на IT для своїх користувачів.

Перелік використаних джерел

1. Mondal, A., Paul, S., Goswami, R.T., Nath, S. (2020). Cloud computing security issues & challenges: A review. In Proceedings of the 2020 International Conference on Computer Communication and Informatics, 22–24 January, 1–5.

2. Narayan, S.; Gagné, M.; Safavi-Naini, R. (2010). Privacy preserving EHR system using attribute-based infrastructure. In Proceedings of the 2010 ACM workshop on Cloud Computing Security Workshop, Chicago, IL, USA, October, 47–52.

3. Grover, A.; Kaur, B. (2016). A framework for cloud data security. In Proceedings of the Computing, Communication and Automation, India, 1199–1203.

4. Khanezaei, N.; Hanapi, Z.M. (2014). A framework based on RSA and AES encryption algorithms for cloud computing services. In Proceedings of the Systems, Process and Control (ICSPC), Malaysia, 12–14 December, 58–62.

5. Abha, S., Bhanali, M. (2013). Cloud computing security using AES algorithm. Int. J. Comput. Appl., 67, 19–23.

6. Gai, K.; Qiu, M., Zhao, H., Xiong, J. (2016). Privacy-aware adaptive data encryption strategy of big data in cloud computing. In Proceedings of the 2016 IEEE 3rd International Conference on Cyber Security and Cloud Computing China, 273–278.

7. Mohamed, M.E.; Abdelkader, H.S.; El-Etriby, S. (2012). Enhanced data security model for cloud computing. In Proceedings of the 8th International Conference on Informatics and Systems (INFOS), Egypt, 14–16 May, 12–17.

8. Kumar, D.A.; Dubey, A.K.; Namdev, M.; Shrivastava, S.S. (2012). Cloud-user security based on RSA and MD5 algorithm for resource attestation and sharing in java environment. In Proceedings of the 2012 CSI Sixth International Conference on Software Engineering (CONSEG), India, 5–7 September, 1–8.