Анотація

Удосконалення методів захищеності хмарних сервісів// Кваліфікаційна робота освітнього рівня «Бакалавр» // Москіна Вікторія Ігорівна // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем і програмної інженерії, кафедра кібербезпеки, група СБс-42 // Тернопіль, 2021 // С. , рис. –10, табл. –12, кресл. – 0, додат.

Ключові слова: інформаційна безпека, хмарні обчислення, загрози безпеки інформації, база даних, методи розбиття

Метою даної кваліфікаційної роботи є отримання набору технологій для гарантування безпеки в одній із послуг які включає концепція хмарних обчислень, а саме хмарних базах даних. Об'єктом дослідження є захищеність інформації хмарних обчислень. Предметом дослідження є забезпечення належного рівня захисту інформації в хмарних базах даних. Методи дослідження: опрацювання літератури та інших інформаційних джерел за даною темою, аналіз існуючих методів та засобів захисту інформації в хмарних базах даних та їхніх характеристик. Результати роботи можуть бути використані для побудови системи захисту інформації, застосовної до хмарних баз даних.

ANNOTATION

Improvement of methods of protection of cloud services // Qualification work of educational level "Bachelor" // Moskina Victoria Igorivna // Ternopil National Technical University named after Ivan Pulyuy, Faculty of Computer Information Systems and Software Engineering, Department of Cyber Security, SBS-42 Group // Ternopil , 2021 // C., fig. -10, table. -12, chair. - 0, add.

Keywords: information security, cloud computing, security information threats, database, partitioning method

The purpose of this qualification work is to obtain a set of technologies for guaranteeing security in one of the services that includes the concept of cloud computing, namely cloud databases. The object of research is the security of cloud computing information. The subject of research is to provide an adequate level of protection of information in cloud databases. Research Methods: reviewing literature and other information sources on the topic, analyzing existing methods and tools for protecting information in cloud databases and their characteristics. The results of the work can be used to build an information security system applicable to cloud databases.

Зміст

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ,СКОРОЧЕНЬ І ТЕРМІНІВ

ЦОД - Центр обробки даних

ПЗ - Програмне забезпечення

IaaS - Infrastructure as a Service

PaaS - Platform as a Service

SaaS - Software as a Service

DBaaS - Database as a Service

EC2 - Elastic Compute Cloud

API - Application Programming Interface

SDL - Security Development Lifecycle

MAC - Mandatory access control

DAC - Discretionary access control

RBAC - Role based bccess control,

ABAC - Attribute based access control

ДТС - Довірена третя сторона

ОС – Операційна система

ВСТУП

В даний час розвивається новий інформаційний світ - він дозволяє взаємодіяти та вести бізнес безконтактно. Зараз коли майже всі знаходяться в Інтернеті є безліч пристроїв, і через це спостерігається вибухове зростання кількості підключень до мережі, що має суттєвий вплив на повсякденне життя. Разом зростає також «Інформаційний вибух» він приносить такі складності як:

- В бізнесі, з’являється багато конкурентів.
- IT-ресурси стали складнішими внаслідок розвитку цих технологій. Споживачі прискорили зростання обсягів інформації.
- Споживачі, що отримують послуги за допомогою інтернету мають великий вибір технологій, для інтернет розрахунків.

Актуальність роботи: Активний розвиток ринку хмарних послуг робить актуальним питання пов’язаних з безпекою інформації в сфері IT. Осатанім часом з’являються нові технології, для роботи з якими використовуються аспекти безпеки.

Мета роботи: Отримати набір технологій щоб гарантувати безпеку включаючи концепцію хмарних баз даних.

1 Хмарні сервіси

1.1. Загальний огляд

Під терміном хмарна обробка даних розуміють комп'ютерні ресурси та потужності в вигляді інтернет-сервісу. Тому ресурси надаються в чистому вигляді, і споживачеві буває неможливо знайти, ті комп'ютери що наразі обробляють його запити, і не знають якою ОС вони робляться.

Часом хмари можна порівняти з мейфреймами, перша відмінність мейфрейма від хмари в тому, що теоретична потужність практично не обмежується. Друга відмінність це те, що термінали для мейфреймів служили лише для взаємодії із споживачем. Хмарний термінал є досить потужним обчислювальним пристроєм який здатний безпосередньо керувати глобальною системою обчислювальних ресурсів.

Технології обробки даних виникли 90-х роках і набули популярності grid-обчислення. Спочатку напрямок розглядався можливістю використання у вільних ресурсах процесів, а також розвиток системи довільної оренди у обчислювальних потужностей.

Останнім часом обчислювальні хмари мають в своєму складі тисячі серверів, розміщених в ЦОД. Ці хмари забезпечують собою ресурси десятки тисяч застосунків, якими одночасно користують багато споживачів. Так що хмарні технології це інструмент досить зручний для тих підприємств які не можуть тримати власні ERP, CRM, бо вони вимагають придбання додаткових ресурсів.

В приватних споживачів отримують перевагу такі послуги, які надає компанія Google.

Хмарні технології стали популярності завдяки тому що: вони мають дуже різноманітну сферу застосування а також економлять не тільки на обслуговувані персоналу а також на інфраструктурі. Апаратне забезпечення є дуже спрощене при обробці даних та зберіганні інформації в віддалених центрах даних. Більша частина проблем перекладається провайдеру який надає свої послуги. Разом

такий підхід дає дозвіл стандартизувати ПЗ, незважаючи на те, що всі комп'ютери на підприємстві можуть мати різну ОС. Також хмарні технології дуже полегшують компаніям забезпечення до доступу даних споживачів, і також даних компанії, власних співробітників, які перебувають поза офісом, та мають доступ до інтернету.

Головним недоліком хмарним обчислень є повна залежність постачальна від послуг, іншими словами підприємство залежить від доступу до інтернету і провайдера сервісу. Хмарні обчислення не підходять підприємствам, які мають пряме відношення до державної і військової таємниці. Тому жодна комісія не може видати для цього сертифікат.

1.2 Моделі розгортання хмарних технологій

Моделі розгортання діляться на приватні, гібридні та загальнодоступні.

Гібридні хмари це поєднання приватних і загальнодоступних хмар. Зазвичай ці хмари створює підприємство, обов'язки управління виконуються підприємством та постачальником хмари, гібридні хмари надають послуги, яких половина відноситься до приватних, і половина до загальнодоступних. Переважно такий тип хмар використовує якщо організація має сезонний період активності. Іншими словами, коли внутрішня IT-інфраструктура не має можливості виконувати поточні завдання, тоді частина потужності перекидається на публічну хмару, а так само щоб надати доступ споживачам до ресурсів підприємства продовж чого публічна хмара і добре продумана гібридна хмара може обслуговувати по всім вимогам безпеки критично важливих процесів таких як отримання платежів. Головний недолік гібридної хмари це складність створення цих рішень та управління ними. Необхідно отримувати із різних джерел та організовувати, так якщо б це було єдине джерело між загальнодоступними та приватними компонентами, рішення може стати складнішим. Тому що відносно нові архітектурні концепції у сфері хмарних обчислень, з'являються практичні рекомендації та інструменти, так що її

поширення може затягнутися до того часу поки архітектура не буде краще вивчена.

Приватні хмари вони є внутрішніми хмарами інфраструктури та служби підприємства. Приватні хмари розташовані у межах корпоративної мережі. В організації хмарою керують зовнішні підрядники чи нею можна керувати самостійно, в приміщеннях замовника розміщується інфраструктура та вона може розміщуватися і у зовнішнього оператора. Найкращий варіант приватної хмари це хмара яка розгорнута на території організації в якій контролюють і обслуговують хмару співробітники компанії. Приватні хмари відрізняються від загальнодоступних тим, що підприємство саме займається підтримкою і установою хмари.

В приватних хмар є багато переваг а саме детальніший контроль над великою кількістю ресурсів хмари та забезпечує компанію багатьма доступними варіантами конфігурації. Тому приватні хмари більш захищені для інформації яку не можна довірити загальнодоступним хмарам.

Загальнодоступні хмари це хмарні послуги ті що надає постачальник. Вони розташовуються за межами корпоративної мережі. Споживачам цих хмар не дають можливість керувати даними та обслуговувати хмару, вся відповідальність належить власнику хмари. Постачальник хмари бере участь у обов’язків встановлення і обслуговування хмари. Споживачі мають платити тільки ті ресурси хмари що вони отримують. Абонентом хмари може бути споживач та компанія. Ці хмари за доступною ціною пропонують легкий спосіб розгортання web-сайтів чи бізнес-сайтів із можливостями масштабування, ті що у інших мають бути недоступні. Є такі приклади розгортання сайтів: сервіси-онлайн Amazon EC2 та Amazon Simple Storage Service, Salesforce.com, Google Apps, Microsoft Office Web. Публічні хмари надаються у вигляді стандартної конфігурації яка складається з найбільш поширених випадків її використання. Це означає що споживачеві дається небагато в порівняні з системами де споживач має змогу сам надати конфігурацію мережі. В цій мережі споживач слабо контролює інфраструктуру, так що процеси які вимагають сурових

заходів безпеки і відповідають нормативним вимогам, нечасто підходять для реалізації в цій хмарі.

Видами хмарних сервісів можуть володіти і розпоряджатися не тільки споживач і провайдер, так і вони разом. Так само відрізняються і права доступу споживачів ресурсів(Див. таблицю 1).

Таблиця 1 Управління та обслуговування різними видами хмарних ресурсів

Види хмари	Ким обслуговується інфраструктура	Власник інфраструктури	Місце знаходження інфраструктури	Доступ до хмари
Публічне	Зовнішнім провайдером	Зовнішній провайдер	У зовнішнього провайдера	У будь-якого споживача
Приватне /Суспільне	Споживачем або зовнішнім провайдером	Споживач або зовнішній провайдер	У зовнішнього провайдера або споживача	У авторизованого споживача
Гібридне	Споживачем і зовнішнім провайдером	Споживач і зовнішній провайдер	У зовнішнього провайдера і споживача	У авторизованих і у будь-який зовнішніх споживачів

Віртуалізація дає змогу краще структурувати, об’єднувати в пул також дає змогу надавати динамічно ресурси інфраструктури: десктопи та сервери, ємності щоб зберігати дані, сполучне ПЗ, мережеве обладнання. Для того щоб технічне середовище вважалося хмарним потрібно ще інші складові віртуальні

машини, ОС, ПЗ grid-обчислень, ПЗ для абстрагування ресурсів зберігання, засоби масштабування та кластеризації.

Терміном приватна хмара називають ресурси, які використовуються одною організацією, це означає, хмарні ресурси організації є повністю ізольовані у хмарі від інших. Головна помилка є у тому, що хмара має економити гроші. Економія може бути, та вона не має бути обов'язковим атрибутом. Приватна хмара дає дозвіл щоб ефективно перерозподіляти ресурси, для того щоб задовольнити корпоративні вимоги, та зменшити капітальні витрати для купівлі обладнання. Приватна хмара вимагає інвестицій для автоматизації, і економії яка не окупить своєї вартості. Тому, зменшення витрат не має бути головною перевагою для моделі. Головним стимулом тут є не економія, а швидкість виходу на ринок, та можливість швидко адаптуватися та динамічного масштабування відповіді товари попиту, вони дозволяють як найкраще підвищити швидкість для впровадження нових сервісів. Приватна хмара має це конфіденційність, і не має конкретне місце розташування. Деякі, наприклад, можуть розмішувати свої ЦОД у хостинг-провайдерів, та потрібно ізолювати провайдерів окремо це можливість віртуальна приватна мережа і за допомогою інших подібних технологій. Приватна хмара це не тільки сервіси інфраструктури. Серверна віртуалізація –використовується для приватних хмарних обчислень та є потужний двигун для них. Та приватна хмара не закінчується на інфраструктури та послузі IaaS. Розробки та тестування нового ПЗ PaaS має великий сенс, більший за надання віртуальних машин.

Останнім часом найшвидший сегмент хмарних обчислень - є IaaS. Цей сегмент надає низькорівневі ресурси для ЦОД у просту для розуміння форму, та не впливає на фундаментальний принцип роботи. Для створення нового застосунку, спочатку вони є призначені для хмари та надають найновіші послуги. Приватна хмара має можливість не бути приватною. З одного боку, приватна хмара надає такі переваги: ефективність, масштабованість і швидкість перебудови, а також захищає від потенційних і реальних загроз безпеки. Більшість сервісів приватних хмар, переходять в гібридні хмарні сервіси, вони

розширюють доступні можливості за рахунок використання загальнодоступних хмарних послуг та інших сторонніх ресурсів.

1.3 Основні властивості хмарних технологій

Національний Інститут USA NIST у своєму документі "The NIST Definition of Cloud Computing " подає визначення різним характеристикам хмар таким як: широкосмуговий доступ до мережі. Цей доступ можна отримати з використанням деяких типів пристроїв. Є кілька сильно поширених пристрої та мобільні телефони, тонкі споживачі. Контраст широкосмугового доступу до мережі є доступом для обчислювальних та мережевих ресурсів у епоху мейнфреймів.

Самообслуговування на вимогу це ключова характеристика для хмарних обчислень. Коли розглядати IT це є складний ланцюжок поставок у застосунках та кінцевим споживачем у кінці ланцюжка, надається здатність до самозабезпечення завдяки ресурсам у типових IT-середовищах це фундаментально порушує робочий процес та процеси, ті що розвивалися в функції корпоративних IT в останні кілька десятиліть. Вони включають у собі робочі процеси, які пов'язані з закупівлею систем зберігання, та серверів, мережевих вузлів, і ліцензій для програмного забезпечення. Відомий ефект у управлінні ланцюжком поставок - є неповна чи неточна інформація що призводить до занадто високої мінливості у виробничих витратах ця мінливість відноситься не тільки в виробниче середовище, та у безпосередньо в виділення IT ресурсів які знаходяться не у хмарному середовищі. Хмарні архітектури, створюються та проектуються із урахуванням необхідності в самозабезпеченні. Передумовою є використання складних програмних рамок, та пакетів для програмного забезпечення COTS, вони призначаються щоб управляти та автоматизувати у корпоративних робочих навантажень.

Об'єднання ресурсів в пули це фундаментальна передумова масштабованості у хмарі. Якщо об'єднаних обчислень мереж та сховищ постачальник послуг не об’єднані будуть надаватися всі послуги через декілька

ізольованих чи дискретних, незалежних ресурсів але з невеликою кількістю з'єднань але можливо це робити і без них. Інфраструктура розрахована щоб багато споживачів середовища, могли спільно використовують суміжні ресурси які знаходяться у спільні хмарі. У разі використання багатокористувацької оренди може спостерігатися неминуче збільшення у експлуатаційних витратах, але їх можна зменшити за допомогою певних конфігурацій в обладнанні і програмних рішень, як у профілі застосунків та серверів.

Вимірюваний сервіс має значення користування об'єднаних ресурсів він контролюється та повідомляє споживачу, при цьому забезпечує прозорість норм споживання та витрат. Моніторинг для використання цілей повернення платежів він давно є вимогою для багатьох зацікавлених сторін.

Остання виділена характеристика, в визначенні хмарних обчислень NIST, це швидка еластичність. Її вирішальне значення щоб скоротити витрати та час виходу на ринок є еластичні обчислення.

1.4 Моделі обслуговування хмарних технологій

Останнім часом виділяють основні моделі обслуговування хмарних технологій, вони називаються шарами хмари. Ці шари хмари це послуги інфраструктури, та послуги платформи, послуги застосунків і вони відображають будову не тільки хмарних технологіях, та і інформаційних технологіях у цілому. До послуг інфраструктури відноситься набір фізичних ресурсів, до яких входять сервери, мережеве обладнання і накопичувачі, які були пропоновані замовникам послуг, які надаються. Послуги інфраструктури надають обчислювальні потужності у міру необхідності в вирішені задач належного до оснащення ЦОД. Часто такі послуги підтримують інфраструктуру та велике число споживачів якщо рівняти із послугами застосунків. Приватним прикладом послуг інфраструктури є апаратне забезпечення. Споживач отримує обладнання, в якому може розгорнути власну інфраструктуру із використанням підходящого ПЗ. При цьому споживач не може керувати базовою інфраструктурою хмари, та йому надається контроль над ОС, системами

зберігання, та розгорнутими застосунками. В цьому випадку для захисту платформ та застосунків відповідає споживач, але провайдер хмари відповідає за організацію захисту інфраструктури.

Послуги платформи - модель обслуговування, у якій споживачеві надаються застосунки в вигляді набору послуг. Безпека як послуга надає можливість споживачам швидко відкривати продукт, для того щоб дозволити забезпечити безпечне використання у web-технологіях, та безпеку електронного листування, безпеку локальної системи. Цей сервіс дає дозвіл споживачам заощаджувати в розгортанні та підтримці власної системи безпеки. Модель PaaS - є IaaS разом з ОС та її інтерфейсом прикладного програмування. Споживач не має можливості керувати базовою інфраструктурою хмари, та має контроль над розгорнутими застосунками. Таким чином, споживач сам має подбати щоб забезпечити захист застосунків, які будуть розгорнуті в наданих платформах.

Застосунки можуть працювати не тільки у хмарі, а і в традиційних ЦОД підприємства. Щоб досягти масштабованості, необхідної у хмарі, послуги часто віртуалізуються, так і як у розглянутих раніше послугах інфраструктури. Переваги цих послуг. Плавне розгортання версій, це означає, що споживач немає відчувати зміни ПЗ в хмарі.

Застосунки як послуга надають доступ в застосунки і до сервісу, іншими словами застосунки провайдера запускаються у хмарі та надаються споживачам на їх вимогу як послуги. Тобто, споживач може отримувати доступ до ПЗ, яке є розгорнуте у віддалених серверах, це все робиться за допомогою Інтернету, причому всі запитання, оновлення і ліцензію на дане ПЗ регулює постачальник даної послуги. Оплата в цьому випадку здійснюється за використання ПЗ. Ця програма доступна за допомогою різних клієнтських пристроїв чи через інтерфейси тонких споживачів, як Web-браузер, чи web-пошта, чи інтерфейси програм, споживач не може керувати базовою інфраструктурою хмари, у тому числі мережами, і серверами, ОС на кінцевому. Ці послуги застосунків є знайомі повсякденного споживачеві. Найпопулярніший прикладом застосунків цього типу це поштові сервіси як GMail, та Yahoo Mail. Існують тисячі застосунків SaaS, та завдяки технології Web 2.0 число зростає із кожним днем. Серед служб

застосунків існує безліч застосунків, які націлені на корпоративне співтовариство. Переваги: В апаратному забезпечені та трудових ресурсів знижується капіталовкладення.

Моделі поділяються за засобами доступу і обслуговування вмістом їх можна побачити у таблиці 1.2.

Таблиця 1.2 - Моделі обслуговування за засобами доступу і управління

Моделі для обслуговування	Засоби доступу та управління	Зміст
SaaS	Web-браузер	Хмарні програми: соціальні мережі, та системи управління вмістом, застосунки для офісів, і інтелектуальна обробка даних.
PaaS	Середовища хмарної розробки	Хмарна платформа: мова для програмування, структуровані дані бібліотеки, та утиліти конфігурації.
IaaS	Система управління для віртуальної інфраструктури	Хмарна інфраструктура: сервера для обчислювання, і сховища даних..

Це ще не всі моделі якими обслуговуються хмарні технології. Існує багато інших таких як:

- Апаратне забезпечення як послуга - модель купівель, вона є аналогічна лізингу чи ліцензуванню, при цьому устаткування не належить споживачеві але належить постачальнику керованих послуг, та встановлюється на об'єкті замовника, і угода визначає відповідальність яку особи беруть на себе.
- Модель HaaS має можливість стати економічно ефективним способом але тільки для малого та середнього бізнесу, це дозволяє забезпечити

співробітників найсучаснішим обладнанням при цьому мінімізуючи всі витрати.

- DBaaS - є різновидом моделей PaaS. При його використані споживач отримує доступ до бази даних різного типу але тільки за запитом.

Споживач має право вибрати будь яку базу даних, вказавши лише версію, та загальну конфігурацію. Базу даних можна розмістити в ОС в віртуальній машині лише за запитом споживача..

Останнім часом провайдери хмарних послуг збільшили невелику кількість пропозицій DBaaS. Компанія IBM, надає доступ в масштабуванні та повністю керованої бази даних, вона це робить із допомого стандартній об'єктно-орієнтовані API.

DBaaS складається із компоненту управління базою даних, він керує всіма базовими екземплярами в базах даних через API. API доступний споживачеві через консоль управління, зазвичай web-застосунок, який споживач використовує для управління та налаштування баз даних а навіть для надання чи скасування доступу до цих баз даних.

Переваги та мінуси бази даних як сервісів або DBaaS:

Створення та уведення традиційної бази даних це є дуже дорогим та трудомістким процесом, управління ним є ускладненим, особливо для підприємств із обмеженими ресурсами і вимогами, вони використовують малі чи середні бази даних.

DBaaS має значення, що великі та малі підприємства мають можливість змінювати розміри у своїх базах даних відповідно до потреб компанії та бюджету на який вони розраховують.

DBaaS таж пропонує пакет послуг із управління даними, який пропонує компаніям автоматичне розгортання та керувати власними серверами і інфраструктурою бази даних, бо замість них це робить DBaaS. Бази даних керуються та розміщуються третьою стороною, доступом керує споживач Cloud across the Globe.

Є безліч факторів, що вимагають DBaaS, а не традиційну базу даних. Деякі із них описані нижче:

- Для управління дуже великим обсягом даних.
- Продуктивність.
- Щоб зробити системи міцнішими після аварії та щоб забезпечити функціонування системи під час аварій.

За прогнозами у Міжнародні інформаційні корпорації, розгортання застосунків у хмарі має збільшитись на 15,3% у порівнянні із аналогічним періодом інших років. В звіті також говориться, про те що використання хмарних обчислень дає 520% окупності у інвестиціях через наступні причини:

- Через те що рух до ринку є швидший на 70%. Через те що вже є база даних, і не має затримки в закупівлях.
- Витрати на інфраструктуру знижені у 70-80%. Непотрібно витрачатися на інфраструктуру, мережеві витрати та технічне обслуговування.

Переваги використання DBaaS:

- Висока масштабованість – дуже багато місця у сховищі даних.
- Економічність – одна із найбільших переваг, потрібно платити тільки за те, що ми використовуємо та вартість апаратного забезпечення та мережі.
- В компаніях яким важко в управлінні своїми даними, провайдер може надати не дуже дорогу альтернативу щоб можна було керувати всіма даними у своїх майданчиках.
- У DBaaS компанія має платить лише за дані які вона використовує, та час. Головною перевагою є те що є дуже багато місця у сховищі даних
- Всі витрати несе на собі постачальник, а бізнес інвестує це.
- Постачальник несе відповідальність за безпеку даних та безперебійність.

Мінуси використання DBaaS:

- Не надається доступ у базу даних. В екстрених ситуаціях споживач нічого не зможе зробити.
- Не має контролю у фізичні безпеці серверу. У разі відключення мережі вхід в систему буде не можливий до її відновлення.
- Споживач не може контролювати конфіденційність даних.

2 ЗАГРОЗИ ТА ВРАЗЛИВОСТІ ХМАРНИХ ОБЧИСЛЕНЬ

Хмарні обчислення надають послуги на вимогу, скорочуючи при цьому капіталовкладення у інфраструктуру та максимально збільшуючи використання всіх ресурсів які є у наявності. Хмарні технології надають мобільність застосунків у інфраструктурних сервісах. IT- індустрія в наш час є гнучкою, і спроможною та економічно ефективною вона ідеально підходить для розробки програм і хостингу застосунків подякуючи росту хмарних обчислень та забезпеченню доступу до обчислювальних ресурсів.

Для забезпечення конфіденційності, цілісності і доступності інформації використовують моделювання загроз в вигляді систематичного та всебічного аналізу загроз. Базову інформацію збирає моделювання загроз, воно є в вигляді сценаріїв використання та зовнішніх залежностей, припущень щодо внутрішньої і зовнішньої безпеки, її реалізації. Розробили такі методи моделювання загроз для оцінки і аналізу таких загроз та вразливостей:

- Моделювання Microsoft загроз є модель ефективного процесу яка включає в себе 5 логічних кроків.
- TAM - модель, була заснована спеціально для бізнес-цілей, які були досягнені в програмі. В програмі TAM інструменти використовуються щоб генерувати та класифікувати всі відомі загрози.
- Щоб визначити аналіз загроз потрібно скласти ефективний план для зниження ризиків.
- PNs була заснована за допомогою аналізу персональних мереж.
- Шляхом обробки обчислень парадигма стала широко поширеною. Тому, що споживач є більш захищеним і через це стикається із сферами безпеки. Ця модель є новим моделюванням він включає в себе проблеми комп’ютерного середовища.

В зв'язку з тим, що хмарні середовища є великими розподіленими обчисленнями, методики які перераховувались вище не включають у собі всі

проблеми, через це для вирішення проблем в комп'ютерній безпеці потрібно зробити нові підходи для моделювання загроз.

В дослідженні краще представлено методологічні моделі загроз які використовуються в розгортанні безпечного обчислювального середовища шляхом демонстрації загроз та вразливостей у хмарних обчисленнях та знаходження рішень безпеки. Методологія складається із наступних етапів.

Питання безпеки хмарних обчислень: Визначення властивостей та загроз:

- Загроза - шкода чи несанкціонований доступ, яка виникає у результаті вразливості для знищення активів організації, та діяльності чи системи інформації.
- Вразливість - це слабкість в інформаційній системі, що може бути використана чи викликана ресурсами загроз.

2.1 Загрози хмарних обчислень

Cloud Security Alliance випустив посібник для безпеки в критичних областях та хмарних обчисленнях. Найбільш значні загрози, класифікуються таким чином:

Втрата чи витік даних: Найсерйозніша та жахлива загроза для бізнесу та споживачів. Будь-яка дія може призвести до втрати даних у споживача.

Викрадення облікового запису чи послуги: Слабкість дає змогу зловмисникам вкрасти облікові дані, та доступ до важливих областей.

Небезпечний інтерфейс: Клієнти хмарних обчислень використовують інтерфейси. Технології автентифікації, захищають ресурси обчислювання від шкідливих атак API.

Відмова у обслуговуванні: коли у хмарних сервісах підвищують надійність.

Шкідливий інсайдер: Це коли систему було пошкоджено діловим партнером, і авторизованим співробітником чи адміністратором, головне щоб він мав доступ до мережі чи ресурсів компанії. Данна загроза небезпечна не тільки для конфіденційності а і цілісність та доступність ділової інформації.

Витік даних: загроза несанкціонованого доступу. Щоб знизити ризик цієї загрози слід використовувати шифрування даних.

Зловживання хмарними службами: Це коли споживач із кредитною карткою зареєструвався щоб отримати хмарні послуги. Інтеграція слабких способів виявлення шахрайства дозволяє зловмисникам використовувати далі із допомогою моделей PaaS та IaaS

Недостатня належна обачність: Найважливіші фактори для розвитку хмарних обчислень є зменшення витрат, покращення безпеки та доступ до пулу ресурсів.

Небезпечна міграція віртуальних машин: При переміщені віртуальних машин під час чого об'єднаних хмар та гібридних хмар, зловмисники мають можливість отримати доступ до даних та передати ці дані в ненадійний хост.

Існує загроза загальних вразливостей, через те, що базові компоненти, не можуть забезпечити сильну ізоляцію в моделями хмарних обчислень.

2.2 Вразливості хмарних обчислень

При переміщені критично важливих даних і застосунків організація на хмарні сервіси, потрібно врахувати основні характеристики такі як хмари, і відомі засоби контролю безпеки та сучасні хмарні пропозиції

- Перехват сесії : Під перехватом сесії розуміють відправку хакерами команди до веб-застосунку.
- Вихід за межу віртуальної машини: У цій вразливості зловмисник може запустити у віртуальній машині код, який дозволяє зламувати ОС та взаємодіяти безпосередньо із гіпервізором щоб отримати доступ у хостову ОС. Щоб запобігти цьому злочину система виявляє шкідливу активність, що знаходиться на рівні віртуальної машини.
- Застаріла криптографія: Коли шифрування ненадійне або його взагалі немає зловмисник може розшифровувати зашифровані дані. Для того щоб захистити цю систему, споживач повинен бути впевнений, в тому

чи насправді потрібні дані зашифровані, і використовувати правильне зберігання ключів та розробити хороший алгоритм.

- Несанкціонований доступ в інтерфейс управління: Інтерфейс управління надає доступ до споживачів хмар. Несанкціонований доступ дає можливість зловмисникам в отримані повного контролю над споживачами та застосунками.
- Інтернет-протокол: Якщо немає методів автентифікації, які не входять у базовий протокол, то зловмисники можуть впроваджувати у мережу свій шкідливий трафік.
- Відновлення даних: Хмарні обчислення дозволяють керувати ресурсами між різними споживачами. Ця характеристика призводить до того що дані можуть бути вкрадені та може статися витік даних.
- Виставлення рахунків: Лічильники хмарних обчислень, та послуги вимірювання, як зберігання обліковий запис споживача та обробка використовуються в оптимізації наданих послуг. Ці вразливості включають в себе не тільки обробку даних обліку та виставлення рахунків, а і витік рахунків.
- Замок постачальника: Споживачі хмари залежать від першого постачальника. Через те, що споживачі не мають можливості легко переходити між різними провайдерами.

2.3 Порівняння існуючих фреймворків моделювання загроз

Існуючі моделі розглядаються в таблиці 1, для аналізування та порівняння існуючих моделей.

Таблиця 1 - Порівняння існуючих моделей загроз

Характеристика моделі загроз	Моделювання загроз від Microsoft	TAM	PTA	Модель загроз для персональних мереж	Моделювання загроз у всебічній обчислювальній парадигмі	Модель Аміні-Джаміла
Виявлення та класифікація активів	✓		✓	✓	✓	✓
Встановлення ролей користувачів						✓
Ідентифікація сфер безпеки					✓	✓
Оцінка надійності					✓	✓
Сканування сфер безпеки						✓
Виявлення загрози	✓	✓		✓	✓	✓
Виявлення вразливості				✓	✓	✓
Здійснення контрзаходу					✓	✓
Оцінка та вимірювання загроз	✓	✓		✓	✓	✓
Оцінка та вимірювання вразливостей				✓		✓
Визначення нових активів, загроз, вразливих місць					✓	✓

2.4 Модель Аміні- Джаміла

Модель Аміні- Джаміла представлена на рисунку 1. Складається із чотирьох основних етапів, в якому кожен включає декілька підпунктів. Перший крок якої є виявлення тих активів які мають доступ до цих пунктів. Довіра, є термін надійності, що використовується щоб позначити безлічі надлишкових web-служб. Другий крок визначає чи здатний провайдер надавати вимоги споживачів у буквальному сенсі слова. Унікальні загрози і усунення виявляють шляхом розробки відповідних контрзаходів які продемонстровані в третьому етапі. Його мета - це виявлення нових загроз щоб підвищити безпеку. Це останній етап який представлений в системний рейтинг найбільш небезпечних та дієвих загроз, вразливостей.

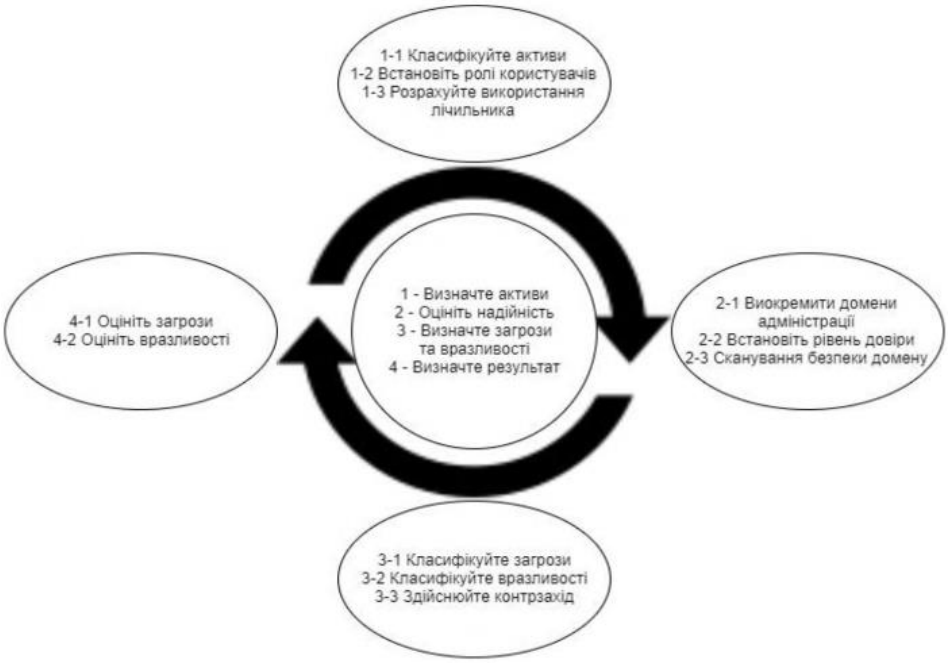

Рисунок 1 - Фреймворк загроз в Аміні-Джаміла

Актив IT - дані, програмного чи апаратного забезпечення, які належать компанії та використовуються щоб вести бізнес. Організації повинна бути впевнена в тому що ресурси не будуть розповсюджені і вони налаштовані щоб захиститися від загроз безпеки і вразливостей. Несанкціонований доступ здійснюються неофіційними споживачем у системі. Через це необхідно знати щоб блокувати атаки машини підприємств та місце розташування. На даний час організації використовують тільки ефективні інструменти управління активами щоб робити аналіз власних даних, але тальки програмного і апаратного забезпечення. Для виявлення і моніторингу активів споживачам запропоновано структуру в якій є декілька кроків (рисунок 1).

Оцінка надійності: Довіра - систем між двома організаціями, що покладаються тільна на свою надійність щоб забезпечити безпеку. Останнім часом щоб оцінити надійність шляхом забезпечення конфіденційності, цілісності, і доступності та достовірності довіра грає важливу роль. Щоб визначити і оцінити довіру необхідно розділити на адміністративні області, та виявити уразливості системи та встановити рівень довіри.

Визначення загроз та вразливостей: Організації виявляють велику цікавість до аутсорсингу своїх ресурсів в віртуальних доменах. Обсяг динних

має можливість бути пошкоджений загрозами із боку ресурсів. Для управління і оцінки ризиків потрібно розуміти та аналізувати загрози та вразливості. Ефективна класифікація безпеки потрібна щоб виявити загрози та вразливості в класах які засновані на передбачувані атак та розробки рішень про ефективні загрози у системі.

Визначення результату: Наявність оцінки систем вразливостей, та ранжування ступеня загроз щоб приймати рішення як краще робити щоб захистити систему від шкідників. Результатом цього є список чи база даних для профілів загроз та вразливостей, які складається із відсортованих ризиків в безпеки.

2.5 Вибір підсистеми контролю доступом

Політика що дозволяє обмежувати та забороняти доступ у систему називається контролем доступу. Крім цього, доступ має можливість відстежувати та реєструвати всі дані про спроби доступу до системи коли і хто це робив. Контроль доступу виявляє споживачів, що намагаються отримати доступ без відома споживача у його системи. Є багато моделей контролю доступу, що включають у собі найбільш поширені такі як: мандатне керування доступом, вибіркове керування доступом та керування доступом в основі ролей. Всі перераховані моделі є відомими моделями контролю доступу на основні ідентифікації. У моделях управління доступом всі споживачі та ресурси ідентифікуються із унікальних іменами. Ідентифікація здійснюється безпосередньо з допомогою ролей, які були призначені суб'єктами. Це ефективні моделі для контролю доступу у незмінних і розподілених системах.

Останнім часом великі розподілені системи розвиваються дуже швидко. До дих моделей відносяться мережеві обчислення та хмарні обчислення. У системах споживачі та постачальники ресурсів вони не належать в одну і ту саму галузі безпеки. Споживачі завжди ідентифікуються по їх атрибутам чи характеристиками. В таких випадках у традиційних моделях контролю доступу, вони засновані на ідентифікації, але вони не надто ефективні, через це доступ у

систему повинен здійснюватися на основі рішень, які засновані на певних атрибутах.

Є такі методи для контролю доступом: Перший спосіб, для забезпечення безпеки ресурсів та даних, - контроль доступу до ресурсів та самої системи. Контроль доступу дає дозвіл споживачам керувати, файлами та іншими ресурсами. Також контролює надання прав споживачу в доступ до файлів. В системах контролю доступу застосовують різноманітні кроки, такі як ідентифікація, автентифікація, і авторизація.

В кожному етапі інформаційних технологій дослідники та технологи усвідомили важливість для запобігання втручанню споживачів в роботи всіх у спільних системах. Було розроблено багато моделей для контролю доступу. Особистість споживача є основним показником, який дозволяє споживачам використовувати систему чи її ресурси. Цей підхід має назву Контроль доступу він заснований на основі ідентифікації. Разом з зростанням мереж та кількості споживачів у них встановили що IBAC є слабким щоб захищати велику кількість споживачів. Вдосконалені концепції контролю доступу, що включали групу/громадськість/власника. IBAC виявився проблематичним в розподілених системах. Управління доступом у системи та ресурсів стали важкими та вразливим до помилок. Також був представлений новий метод який був відомий як RBAC. Керування доступом споживача у систему на основі його ролі. Роль, призначена споживачу, і у основному вона заснована на принципі найменших привілеїв. Роль визначається із найменшою кількістю дозволів чи функцій, необхідна щоб виконання роботи. Дозволи можуть надавати чи видалятися, коли змінюють привілеї для певної ролі. Проблеми в RBAC є очевидними, коли вони були поширені в адміністративних доменах. З важко домовитися про привілеї асоціювання із різними ролями. В ABAC доступ надається тільки за атрибутами, які споживач може дати, такі як, дата його народження чи номер мобільного телефону. Дійти згоди дуже важко, особливо у численних установами та організаціями. Методи контролю доступу засновані на автентифікації споживача в сайті під час запитів. Такий метод називається методом керування доступом на основі автентифікації. Для зв’язку з доменами

потрібний тісний зв'язок. Крім цього, всі підходи ускладнюють призначення до підмножин прав в адміністратора. Завдяки цьому загальні схеми використання, мають змогу реалізовуватись шляхом скорочення функціональних можливостей чи порушення принципу найменших привілеїв.

Контроль доступу розширюється наступними функціями:

1) Делегування повноважень для визначення атрибутів
2) Децентралізація атрибутів та функцій
3) Інтерференція різних атрибутів

ABAC надає забезпечення політики конфіденційності повноважень. Ці повноваження дають змогу організації які зберігають свою автономність. Разом ABAC забезпечує автоматичні переговори щодо довіри, є можливість перевіряти їх коли це необхідно.

2.6 Вибір підсистеми аудиту

Для застосунків хмарні обчислення є перспективні у області інформаційних технологій, але персональні споживачі та підприємства мають вирішити деякі питання, що пов'язані з зберіганням даних та розгортанням застосунків. Найбільша перешкода на шляху впровадження є безпека даних, до неї входять питання як конфіденційність, і нормативно-правова відповідність, та правові запитання. Через це найважливіша ціль це підтримка безпеки та цілісності даних які зберігаються в хмарі. З самого початку потрібно усунути сторбованість споживачів за безпеку, для того щоб хмарне середовище стало надійним та споживачі і підприємства адаптуватися до його масштабів.

Основними проблемами включають в себе конфіденційність даних, та доступність даних, та їх розміщуються та безпечну передачу. Загрози такі як втрата даних, та зовнішні шкідливі атаки це основні проблеми безпеки у хмарі. У хмарній системи цілісність даних означає забезпечує інформації яка зберігається. А також дані не мають бути втрачені чи змінені. Цілісність інформації та точність даних підтримуються постачальниками. Щоб краще підтримувати конфіденційність даних потрібно використати стратегію для

контролю доступу. Проблеми з конфіденційністю даних вирішуються за допомогою підвищують надійність у хмарних обчисленнях.

Аудит даних вводиться у хмарні обчислення щоб забезпечити зберігання даних. Процес перевірки даних споживача, може бути здійснений за допомогою самого споживача і за допомогою сторонніх аудиторів це є визначення аудиту. Воно дає допомогу для підтримування цілісність даних, які зберігаються у хмарі. Роль верифікатора є розділена на дві рівні частини: перша це - приватний аудит, він означає що тільки споживач чи власник даних має право на перевірку цілісності даних, які зберігаються. Друга –надання публічному аудиту, що дозволяє всім, а не лише споживачу, робити запит в сервер та виконувати перевірку достовірності даних за допомогою TPA. TPA - є сутністю, що використовується тільки щоб діяти від імені споживача. Ця сутність володіє всіма необхідними знаннями, та можливостями, професійними навичками, що необхідні у виконання роботи із перевірки цілісності даних. Є одна важлива річ це те що TPA може ефективно перевіряти хмарне сховище даних і не давати запит до локальної копії даних.

У хмарному середовищу є три мережевих об'єкта(рис.2) – споживач і хмарний сервер та TPA. Споживач зберігає дані свої на сервері, який надається провайдером хмарних послуг CSP. TPA робить перевірку даних споживача, періодично перевіряючи цілісність даних коли того вимагає споживач і його повідомляють про всі зміни чи помилки, які було виявлено у даних споживача. На рисунку 2 показана архітектура в хмарному сховищі даних.

Рисунок 2 Хмарна архітектура зберігання даних

В таблиці порівнюються різні фактори, такі як використаний метод, із підтримка громадського аудиту, та збереження конфіденційності, динаміка даних та пакетний аудит. Таблиця також показує, чи є підтримується цілісність та конфіденційність даних, які зберігаються у хмарному сервері. В таблиці 2 видно, як для перевірки цілісності даних застосовуються різні методи. Та в кожному методі є пов'язані міх собою проблеми.

Таблиця 2 - Наукові роботи по критеріям

Наукові роботи	Використаний метод	Підтримує публічний аудит	Підтримується збереження конфіденційності	Підтримує динамічні дані	Підтримує пакетний аудит	Підтримує цілісність даних	Зберігає конфіденційність даних
1	2	3	4	5	6	7	8
Privacy Preserving Public auditing for data storage security in Cloud computing	HLA з випадковим маскуванням	+	+	-	-	+	-
Privacy Preserving public Auditing for Secure Cloud Storage	HLA з BLS підписом	+	+	-	+	+	-
Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing	HLA з BLS підписом та хеш деревом Меркель	+	+	+	+	+	-
Towards Secure and Dependable Storage Services in Cloud Computing	Гомоморфні токени + надмірний код	+	+	+	-	+	-
Secure and efficient privacy preserving public auditing scheme for cloud storage	HLA з BLS підписом	+	+	-	+	+	-
Cloud Server Storage using TPA	Хеш дерево Меркель	+	+	+	-	+	-
Privacy preserving & Public auditing service for Data storage in Cloud Computing	Хеш дерево Меркель + RSA	+	+	-	-	+	+
Privacy Preserving & Batch auditing in Secure Cloud data storage using AES	HLA з випадковим маскуванням + AES	+	+	-	+	+	+
Privacy preserving Public auditing in cloud using HMAC algorithm	Хеш-код аутентифікації повідомлень	+	+	-	-	+	+
Swapnali More auditing	AES з SHA-2 та RSA	+	+	+	+	+	+

Потрібно обрати ефективний протокол публічного аудиту, що дозволяє подолати обмеження, які накладаються іншими системами аудиту ми можемо побачити в таблиці 3.1 - це система аудиту Свапналі Мор.

Потрібно обрати ефективний протокол публічного аудиту, що дозволяє подолати обмеження, які накладаються іншими системами аудиту ми можемо побачити в таблиці 3.1 - це система аудиту Свапналі Мор.

2.7 Вибір підсистему криптографічного захисту БД

Зберігання і обробка конфіденційних даних у системі, що надана третьою стороною збільшує ризик несанкціонованого розголошення, з умовою що система скомпрометована зловмисником.

Одне із можливих рішень для такої проблеми - шифрування даних у машині користувача перед завантаженням даних на сервер, запити виконуються, отримуючи назад зашифровані дані із сервера, розшифровувати і виконувати запит можна у машині споживача. Однак для того щоб дати запит до бази даних та аналітичних навантажень потрібно передати набагато більше даних ніж необхідно, через те що велика частка бази даних зчитується як виконання запиту, і сам результат зазвичай не є великим набором даних або згорткою даних таких як, сумою вартості товарів.

MONOMI спирається на попередню роботу із виконання запитів у зашифрованих базах даних CryptDB, і вирішує деякі проблеми, при цьому не привносячи нових недоліків. Тому вважаю розглядання CryptDB не доцільним, через існування тієї ж за ідеологією, але кращою по реалізації системи.

MONOMI є підходом, заснованим на роздільному виконанні споживач, сервер. Використовуючи алгоритми для шифрування, що дозволяють виконувати операції в зашифрованих даних, такі як порівняння і групування. Роздільне виконання запиту дає дозвіл MONOMI виконувати невелику частину запиту до сервера. Для іншої частини запиту, що взагалі не можуть бути виконані в сервері MONOMI завантажує споживачу проміжні результати та виконує там остаточні обчислення. Схему MONIMI є на рисунку 3.

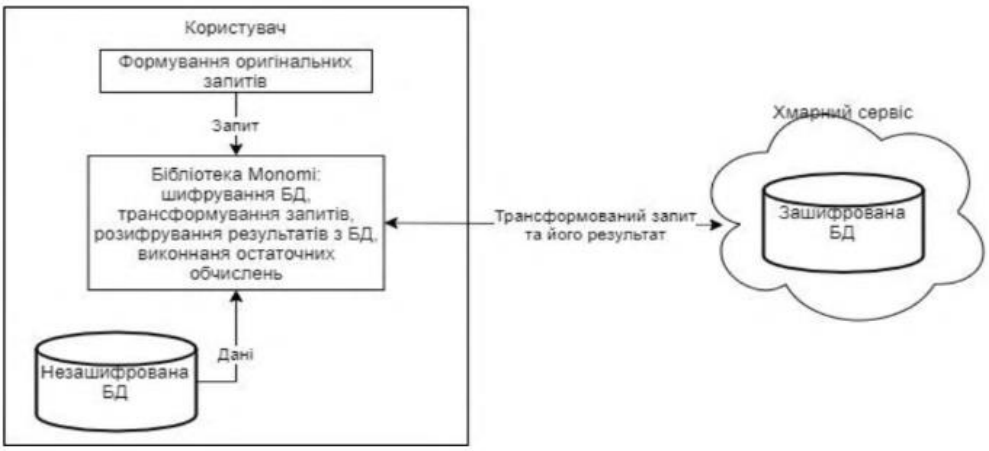

Рисунок 3 - Схема MONOMI

MONOMI порівнюється з системою Віктора Телло, це схематично зображено на рисунку 4

Рисунок 4 - Схематичне зображення системи Віктора Телло

В MONOMI велика частина обчислень виконується на стороні споживача, який заперечує суттєву перевагу використання хмарних баз даних і алгоритми, які використовуються в даній системі для виконання операцій над зашифрованими даними, і допускають витік даних.

В той час як в системі Віктора Телло використовується довірена третя сторона, а споживач не виконує, для виконання всіх проміжних обчислень, таких як: виконання проміжних обчислень, та розбиття і індексування бази даних у зашифрованому вигляді до хмарної бази даних, і не допускається витік цих даних.

Тому у сучасних хмарних базах даних повинна бути використовувана система Віктора Телло.

3 СИСТЕМА БЕЗПЕКИ ХМАРНИХ БАЗ ДАНИХ

3.1 Підсистема контролю доступом

Найбільш важливим механізмом безпеки у хмарних сервісах є контроль доступу, та традиційні моделі для контролю доступу які не застосовуються для хмарних сервісів. У результаті аналізу минулого розділу стало очевидно, що в сучасних системах хмарних обчислень необхідно використовувати керування доступом який оснований на атрибутах. Велика кількість ресурсів, і динамічних споживачів, конструкції динамічної та гнучкої - це деякі із важливих особливості які використовують хмарні сервіси. Крім цього, всі автономні домени у хмарній системі мають власну політику безпеки, і список контролю доступу, декларація що приймає рішення відносно авторизацію SAML такі як заява про політику XACML. Через це важливо мати дуже гнучку модель для керування доступом, для того щоб мати всі ці різноманітні політики у різних доменах. У ABAC рішення щодо надання доступу приймаються на основі атрибутів запитуючої сторони, та сервісу, ресурсів та середовища. Керування доступом складається із чотирьох елементів:

- Сторона що записує(Req): в хмарний сервіс надсилає запити, викликаючи дії в сервісі
- Сервіс($Serv$): програмне і апаратне забезпечення із мережевим та заздалегідь визначених операціях
- Ресурс(Res): з ресурсом здійсню дію хмарний сервіс
- Навколишнє середовище(Env): має інформацію, що статистика є корисна в прийнятті рішеннь відносно доступу. Інформація не пов'язане із будь-якою сутністю.

Атрибути для всіх сутностей, яка визначається особистістю та ознаками визначаються таким чином:

- $Attr(Req) = \{ReqAttr_i | i \in [1, I]\}$
- $Attr(Serv) = \{ServAttr_j | j \in [1, J]\}$

- $Attr(Res) = \{ResAttr_k | k \in [1, K]\}$
- $Attr(Env) = \{EnvAttr_l | l \in [1, L]\}$

Де I, J, K та L це цілі числа, що показують найбільшу кількість атрибутів для всіх сутностей

Система авторизації хмарних сервісів підтримує політику безпеки. Забезпечення для інтеграції в різних політиках щоб зробити масштабний контроль доступу всі політики інкапсулюється в незалежних одиницях. $ABAC$ як політика підтримує якість широкого набору для політик, вони визначаються так:

- Policy $= \{P_m \in [1, M], P_m\}$

Рішення чи надавати доступ приймається за дорогою цієї функції $df()$. $P_n df()$ це функція оцінки в рамках політики P_n , що визначається наступним чином:

- $P_n df\big(Attr(Req), Attr(Serv), Attr(Res), Attr(Env)\big) =$ дозволити або заборонити

Атрибутів мають можливість мати наступний вигляд:

- $RedAttr1 = Attribute(\text{GID} = \text{”admm”} = \text{”#####”})$
- $ServAttr1 =$ Attribute (Special Type $=$ ”Paas”, Service Name $=$ “\ platform creation)
- $Res\text{Attr}1 =$ Attribute(Computing $-$ ”Node 1 and Node 2”, networiking $=$ ”switch 1”)
- $Env\text{ttrl} =$ Attribute (Service Time $=$ ”current Time”, domain $=$ ”Cloud 1 . Clusterl and Cloud 2. Cluster 1”)

Характеристика для ABAC

- Ієрархічна структура політики, засновується на концепції абстракції і інкапсулювання
- Цей набір політик складається із різних стратегій, що потребують підтримки
- Всі політики мають свої власні алгоритми для прийняття рішень

- Не використовується уніфікований метод щоб описати кожну політику
- В політиках є ефективна підтримка для різних стратегій
- Модель статистик гнучка та розширювана

Рішення контролю за доступом статистики дуже важливий в комунальній системі. Але для великих розподілених систем, наприкла хмарна система, рішення відносно доспуту є гнучким та масштабованим. Саме через в аудиті використовують для керування в доступі основі атрибути

3.2 Підсистема аудиту

Вибрана система Свапналі Мор була розроблена щоб перевіряти хмарні дані на сторонні аудитори, і час від часу запити, без змоги получити всі дані чи створити додаткове навантаження на споживачів хмарних середовищ у online режимі, та на сервери хмари. Система не дає можливості відкривати дані стороннім аудиторам.

В схема є три основні елементи: власних данних, сховища хмарного сервера та сторонніх аудиторів. Власник чи споживач данних несе повну відповідальність в поділі файлів у блоки, для шифрування із використанням алгоритму AES, який генерує хеш-значення $SHA-2$ в кожномі файлі, конкатенацію хешів та в генерації сигнатури RSA. Хмарний сервер використовується лише для зберегання зашифрованих блоків файлів. Подякую цьому в файлах немає додаткових навантаженнь для обчислення верифікаційних доказів. Верифікаційним доказом розуміють генерацію хешів щоб зашифровати блоки, потрібно щоб конкатенація та генерація цифрового підпису збігалася. Щоб споживач чи власник даних питав данні в стороннього аудитора, той одразу питає зашифровані дані із хмарного сервіса. Після отримання цих даних аудитор генерує у хеш-значення в кожному блоку зашифрованих файлах. Використовується алгоритм SHA-2, і споживач. Пізніше конкретизується хеш-значення та генерується в сигнатуру RSA для файлу. У

процесі верифікації підпис, який генерується аудитором, та підпис, що зберігає аудитор, надає споживачеві дані. Якщо підписи збігаються оазом, це означає, що дані не є пошкодженими і не були заміненні сторонніми особами чи зловмисниками. Якщо підписи все таки незбігаються, значить що цілісність даних порушена чи замінена. Результати перевірки приходять власнику даних. На рисунку 5 є архітектура для схем аудиту.

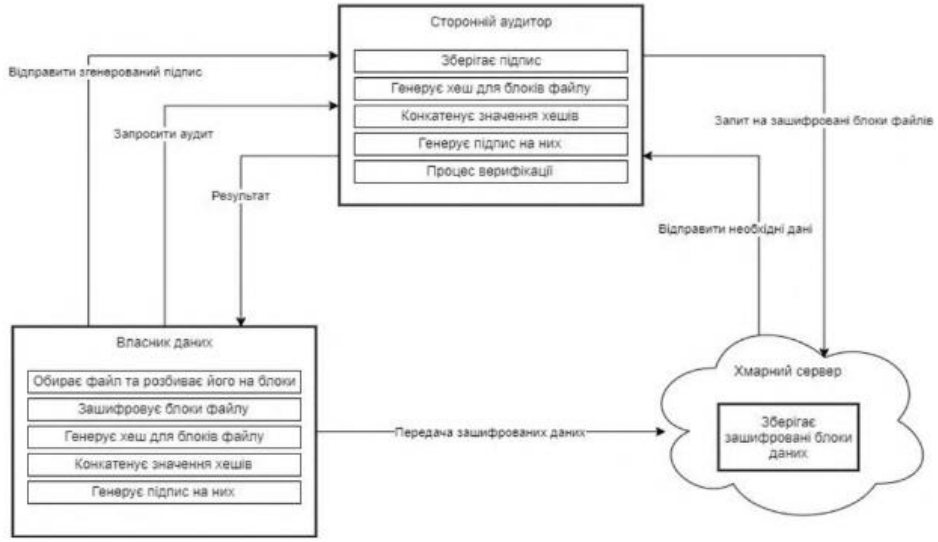

Рисунок 5 - Архітектура для схем аудиту

Власник даних виконує велику частину обов'язків, що пов'язані із даними. У схемі аудиту власник спочатку виконується вхід та регістрація у хмарномий сервер. Новий споживач має спочатку зареєструватися, і бути активним у системі. Після регістрації видається повідомлення про успішної реєстрації. Якщо він є споживачем системи, то виконується вхід в систему.

Після входу власник даних вибере файл, що потрібно зберегти в хмарному сервері. Для розбиття потрібного файлу у блоки, використовують алгоритм $File\ Splitter$. Алгоритм перевіряє, існує файл чи ні. Наприклад: розмір файлу є $23kb$, він розділяється на $20kb$ та $3kb$ розмір розбиття становить 20 кб. Потім використаєм стійкий алгоритм шифрування Advanced Encryption Standard для забезпечиння конфіденційності баз даних. Файл шифрує $128-$бітні блоки даних. Після шифрування $128-$бітних блоків хеш-значення генеруються окремо. Використовується алгоритм хешування $SHA-2$ який використовують для генерування. Після цього, хеші в кожному блоці конкатенуються в

цифровий підпис RSA. Потім цифрові підписи відправляється у сервіс стороннього аудитора, там вони використовують цифровий підпис щоб перевіряти цілісность даних. Власник даних може проводити перевірки цілісності даних в стороннього аудитора. На рисунку 6 показана робота власника даних у рамках цієї схеми аудиту.

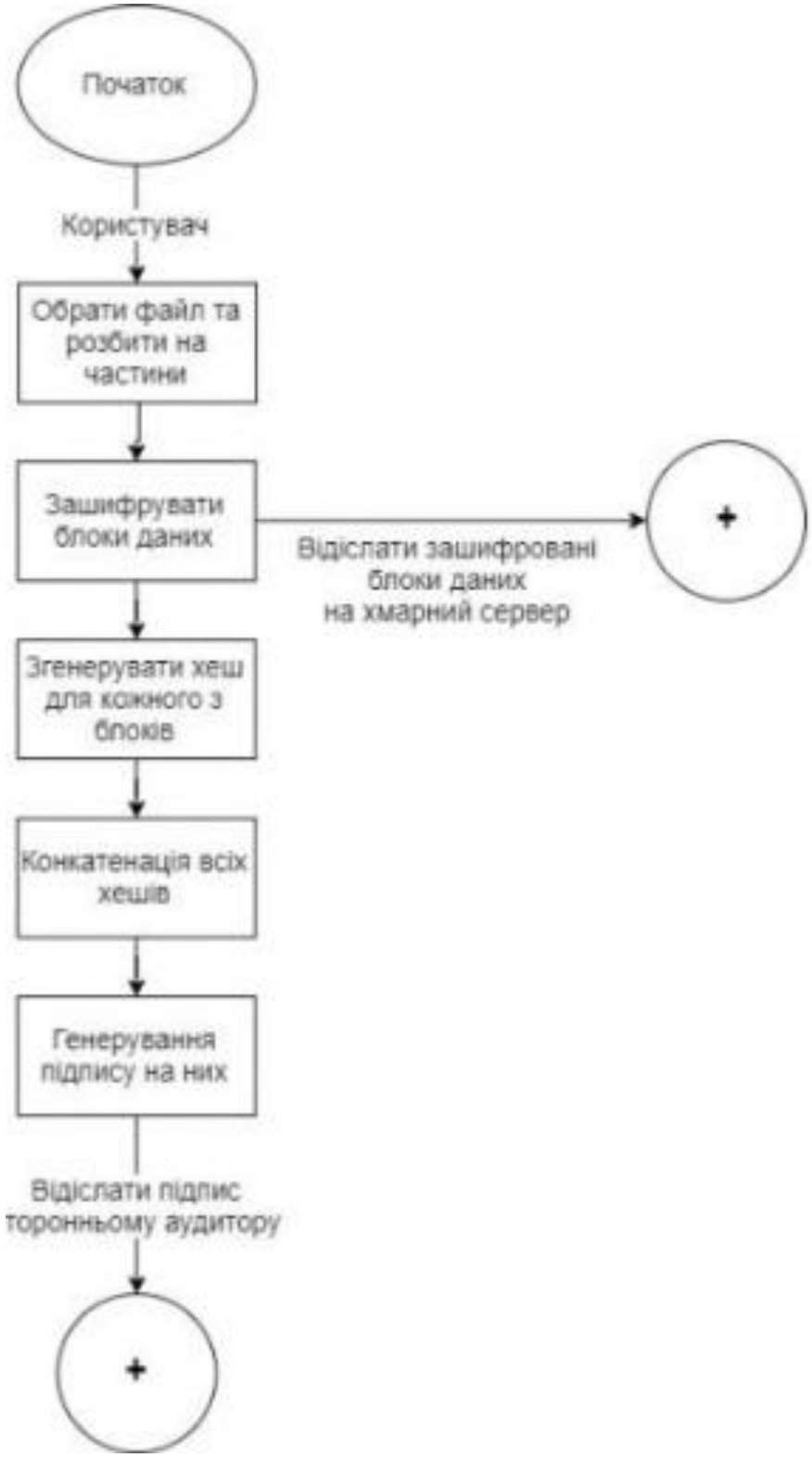

Рисунок 6 - Блок-схема власника даних

Для зберігання даних що зашифровуються використовується хмарне. У данні схемі для виконання завдання із перевірки даних використовують сторонний аудитор. Після отримання запиту від споживача на проведення аудиту данних про сторонній аудитор починає процес. А також дотримування алгоритму, що виконується власником даних, таких як, генерування хеша він може зашифрувати блоки даних, і конкатенація, генерування підпису в них. Пізніше, в процесі верифікації порівнюються два підписи. Коли підписи збігаються, то це означає цілісність даних непорушена. В іншому вападку цілісність порушена. Аудитор надає відповідні результати власнику даних. В рисунку 7 надається приклад робота стороннього аудитора.

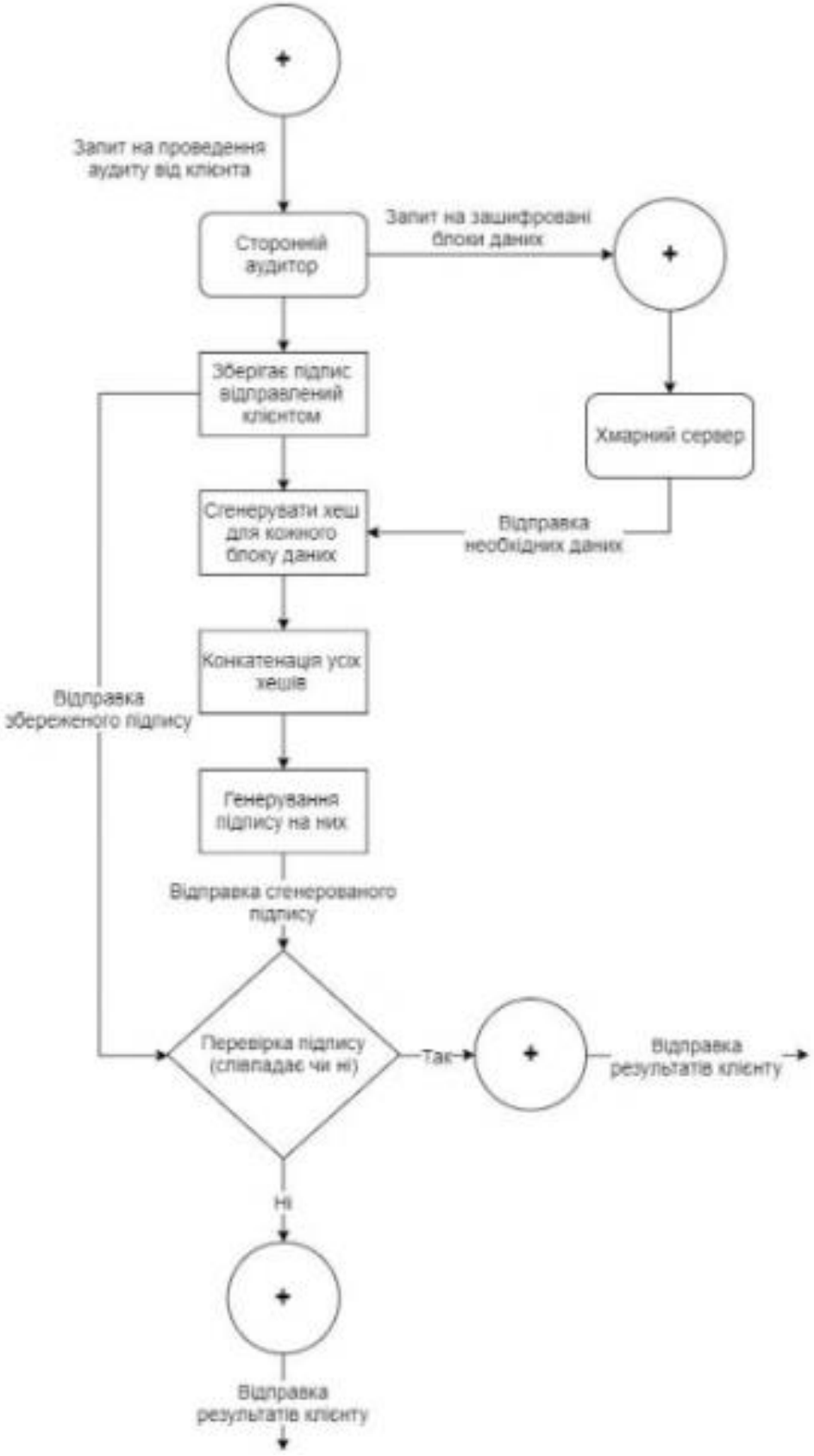

Рисунок 7 - Технологічна схема роботи для стороннього аудитора

3.3 Підсистема криптографічного захисту бази даних

Віктор Фуентес Телло запропонував рішення, що допоже захистити дані із методів шифрування та розбиття на секції.

3.3.1 Довірена третя сторона

Компаніям, даються різні можливості для послуг зберігання, обробки а також зв'язку щоб знизити витрати використовують технологічні інфраструктуру. Багато авторів розробили методи з допомогою ДТС, у яких ДТС виконує функцію управління діяльністю та зв'язку між сторонами. Тут дані контролюються ДТС, і споживачі повинні запитувати доступ щоб оьримати даних та послуг.

3.3.2 Шифрування даних

Споживачі та компанії забезпечують зберігання данних в хмарних серверах. Для їх збереження використовують шифрування даних перед відправкою у хмарний сервіс. Якщо споживач запитує будь-яку інформацію яка є в зашифровані таблиці, для цього він обов’язково повинен запросити всю таблицю. Але призводить до завеликого навантаження на споживачів. Щоб зменшити навантаження додається інформація індексування. Індексні дані використовуються системою управління щоб виборати ті дані які необхідних.

3.3.3 Система Віктора Телло

В системі є три основні сегменти: споживацький сервіс довірена третя сторона та хмарний сервіс. По перше оригінальні дані відправляються ДТС із споживацької машини щоб зберегти корпоративну базу даних у хмарному середовищі зберігання. Організація та ДТС разом використовують один закритий ключ щоб шифрувати та розшифровувати файли для обміну ними. ДТС виконує такі функції: методи розбиття і шифрування записів та відправка

зашифрованих записів у хмарні служби зберігання даних. На рисунку 8 показана схема, яка ілюструє систему.

Рисунок 8 - Модель Віктора Телло

ДТС керує запитами споживачів, які надсилають цей запис. Для цього ДТС аналізує та модифікує запит в залежності від того який методу розбиття, ми використаємо для отримання індексу. Наступний кроком є хмарний сервіс що зберігає дані які надсилається новий запит щоб отримати записи, які відповідають умові запиту. ДТС отримує ці записи та розшифровує оригінальні кортежі щоб застосувати умови запиту, шифрує результати, використовуючи при цьому спільний ключ між ДТС та споживачем. Щоб захисти зв'язкок між ДТС і хмарою використовують криптографічний метод.

Коли первинне зберігання ДТС ділить дані та відправляє їх у хмарну базу даних. Ідентифікатор, відповідає індексу в усіх кортежах та базах данних. Всі атрибути пов'язаний із індексом розбиття.

Метою даної моделі є зниження навантаження в машину споживача. Щоб забезпечити безпеку сервіс зберігає дані в зашифрованому вигляді, це ускладнює розшифрування інформації для зловмисника. Модель ДТС виконує велику частину робочого навантаження. На початку вона розбиває дані на секції, керує запитом, та використовує результати процесу розбиття. В моделі відзначається ДТС який має в усіх організаціях свій ключ. Переваги системи ДТС:

- Основна роботи в машині споживача, мінімальні.

- Весь процес є розділений на дві частини
- Данні зберігаються у зашифрованому вигляді.

3.3.4 Методи розбиття даних

Модель Віктора Телло використовує будь-які методи розбиття даних. Пропонуються два методи: перший це розбиття на основі бісекції дерева і другий це розбиття в основі гістограм.

Перший метод полягає у обробці великої кількості запитів у хмарі чи у сервері ДТС без розшифрування даних. Обробка призводить до мінімалізації розрахунків у споживачській машині. Методика починається у визначенні атрибутів, що використовуються у всіх запитах. Для секцій розбиття, потрібно кожне значення атрибута яке буде співставлено із певним діапазоном.

Другий метод - відображає статистичну інформацію, одним її видів є метод рівномірного розподілу у ширині. На рисунку 9 показані ідентифікатори присвоєні п'яти секціям атрибута. В таблиці 3 показаний результат поділу за секціями.

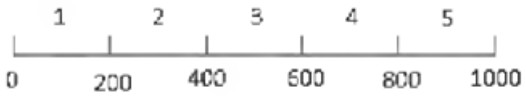

Рисунок 9 - Розбиття значень атрибута на 5 частин

Таблиця 3 - Результат розбиття на секції

Діапазон значень	Секція
[0 - 200]	1
[200 - 400]	2
[400 - 600]	3
[600 - 800]	4
[800 - 1000]	5

Алгоритм розбиття: початок це читання таблиці бази даних, яку необхідно розбити та зберегти, та кінець в ньому алгоритм видає секції розбиття.

Вхідні дані: Таблиця бази даних, та присутні атрибути, які потрібно розбити

Вихідні дані: секції розбиття

1. Ініціалізуйте Набір секцій розбиття = {}
2. Визначте Кількість секцій
3. Визначте Мінімальне та Максимальне значення атрибута, що потрібно розбити
4. Розмір діапазону = ((Максимальне значення – Мінімальне значення) / Кількість секцій)
5. Початок діапазону = Мінімальне значення; Номер секції = 0
6. Кінець діапазону = Початок діапазону + Розмір діапазону
7. Додайте Початок діапазону, Кінець діапазону та Номер секції до Набору секцій розбиття
8. Початок діапазону = Кінець діапазону; Номер секції = Номер секції + 1
9. Якщо (Номер секції < Кількість секцій) Перейдіть до кроку 6
10. Поверніть Набір секцій розбиття
11. Кінець

3.4 Розробка методу розбиття на основі частоти використання

Початок методу розбиття це отримання лог-файлу запитів та бази даних за період часу, за допомогою цього файлу робиться генерація статистичної матриці в усіх умовах оператора WHERY, що будуть виконуватися в цих таблицях. Статистична матриця є створена для кожного атрибута даних, який розбивається в секції. Знизу зображено приклад лог- файлу:

Where заробітня_плата> = 400 та заробітня_плата<=700

Where заробітня_плата> = 400 та заробітня_плата<=800

Where заробітня_плата> = 400 та заробітня_плата<=750

Where заробітня_плата> = 700 та заробітня_плата<=1200

Where заробітня_плата> = 5000 та заробітня_плата<=7000

Where заробітня_плата> = 1000 та заробітня_плата<=3000

Where заробітня_плата> = 1200 та заробітня_плата<=2500

Where заробітня_плата> = 400 та заробітня_плата<=4000

Where заробітня_плата> = 400 та заробітня_плата<=800

Where заробітня_плата> = 400 та заробітня_плата<=2500

Where заробітня_плата> = 2500 та заробітня_плата<=5000

Where заробітня_плата> = 2500 та заробітня_плата<=7000

... … … … …

У таблиці 4 є статистична матриця що розроблена для атрибута заробітної плати. В ній видно, скільки разів секція розбиття використовується у умовах оператора WHERE. Після того як створили статистичну матрицю потрібно забрати будь-яку секцію розбиття. Оптимальне значення порогу відрізняється для всіх запитів і технічних характеристик системи, також загальних рекомендацій.

Таблиця 4 - Статистична матриця

Діапазон до / Діапазон з	750	800	1200	2500	3000	4000	5000	7000
400	5	100	50	200	7	100	100	50
700		4	15	3	1	0	0	5
800			200	100	4	200	100	50
1000			17	13	10	5	2	5
1200				300	5	200	200	100
2500					5	300	200	100
4000							600	100
5000								200

У таблиці 5 видно результат усунення секцій розбиття, через те що секції зустрічались менше порогового значення.

Таблиця 5 - Статистична матриця після усунення значень менше порогового

Діапазон до / Діапазон з	800	1200	2500	4000	5000	7000
400	100	50	200	100	100	50
800		200	100	200	100	50
1200			300	200	200	100
2500				300	200	100
4000					600	100
5000						200

Таблиця 6 показує кількість записів у кожній секції розбиття, за допомогою таблиці видно, що отриманні менші секції розбиття поліпшили швидкодію отримання записів із бази даних.

Таблиця 6 - Кількість записів по діапазонах

Діапазон з	Діапазон по	Кількість записів
400	800	20
800	1200	30
1200	2500	30
2500	4000	25
4000	5000	10
5000	7000	5

Наприклад, секція розбиття $[800, 1200]$ має 200 запитів та 30 записів із таблиці, приносить $200 * 30 = 6000$. Але коли вибрали секцію розбиття $[400, 1200]$, то вона має $20 + 30 = 50$ записів що принесе $250 * 50 = 12500$ записів, при пошуку значень з 800 до 1200. Тому секції розбиття $[400, 1200]$ розбиваються на дві секції $[400, 800]$ і $[800, 1200]$ це дозволяє покращити показники. Покращення становить $12500 - 6000 = 6500$. У таблиці 7 наведені секції розбиття.

Таблиця 7 - Підсумкові секції розбиття

Секція розбиття	Індекс
[400 - 800]	1
[800 - 1200]	2
[1200 - 2500]	3
[2500 - 4000]	4
[4000 - 5000]	5
[5000 - 7000]	6

Атрибути мають дискретні дані, які приймати певні значення. Дискретні дані є числовими, наприклад кількість студентів, також вони є категорійними, як чоловік чи жінка або менеджер чи програміст. Атрибут посада, має дискретне значення, та значеннями для атрибута є менеджер і програміст та архітектор. Після вивчення лог-файлу для атрибута було створено статистичну матриця, що показана у таблиці 8. Наступним кроком видаляємо секцію для розбиття яка менша 90. Таблиця 8 показує результат, забираючи секції розбиття менші за 90. Далі усі усунуті секції об’єднуємо в одну секцію. У таблиці 10 показаний результат цього розбиття.

Таблиця 8 - Статистична матриця для атрибуту Посада

Посада	Менеджер	Програміст	Архітектор БД	Бухгалтер
Менеджер	250	50	40	200
Програміст		30	20	20
Архітектор БД			20	30
Бухгалтер				300

Таблиця 9 - Статистична матриця без значень меншого порогу

Посада	Менеджер	Бухгалтер
Менеджер	250	200
Бухгалтер		300

Таблиця 10 – Розбиття на секції

Секція розбиття	Індекс
Менеджер	1
Бухгалтер	2
Програміст & Архітектор БД	3

Початок алгоритму це читання лог- файлу запитів в таблиці бази даних, які ми зберегли в забрали, кіннець алгоритму видає секції розбиття.

Вхідні дані: Лог-файл вихідні дані: секції розбиття

1. Ініціалізуйте Набір секцій розбиття = {}
2. Прочитайте лог-файл, знайдіть все команди WHERE для атрибуту, що потребує розбиття, розпарсіть значення та додайте їх до двовимірного масиву Статистична матриця, в якому значення початків діапазона знаходяться на місці рядків, а значення кінців діапазона знаходяться на місці стовпчика та на перетині рядка і стовпчика проставлено значення кількості запитів з таким діапазоном
3. Відсортуйте масив Статистична матриця за першим стовпцем і першим рядком у порядку зростання
4. Визначте поріг мінімально допустимого значення частоти
5. Отримайте масив Список_пар_діапазонів, який складається з пар виду { Початок_секції_розбиття, Кінець_ секції_розбиття } взявши попарно значення з нового масиву Статистична матриця після видалення секцій, значення яких менше за поріг
6. Ініціалізуйте Індекс = 0
7. Отримайте Поточний_діапазон = значення на місці Індекс в масиві Список_пар_діапазонів
8. Якщо (Поточний_діапазон ∉ Набір секцій розбиття) Додайте Поточний_діапазон в Набір секцій розбиття
9. Якщо (Індекс < Довжина Список_пар_діапазонів – 1) Установіть Індекс = Індекс + 1 та перейдіть до кроку 7

3.5 Схема структури системи безпеки

Після того як розглянули методи аудиту, контролю доступу і криптографічного захисту бази даних їх потрібно скласти в загальну результуючу схему використовуючи метод розбиття даних що зображений на рисунку 10.

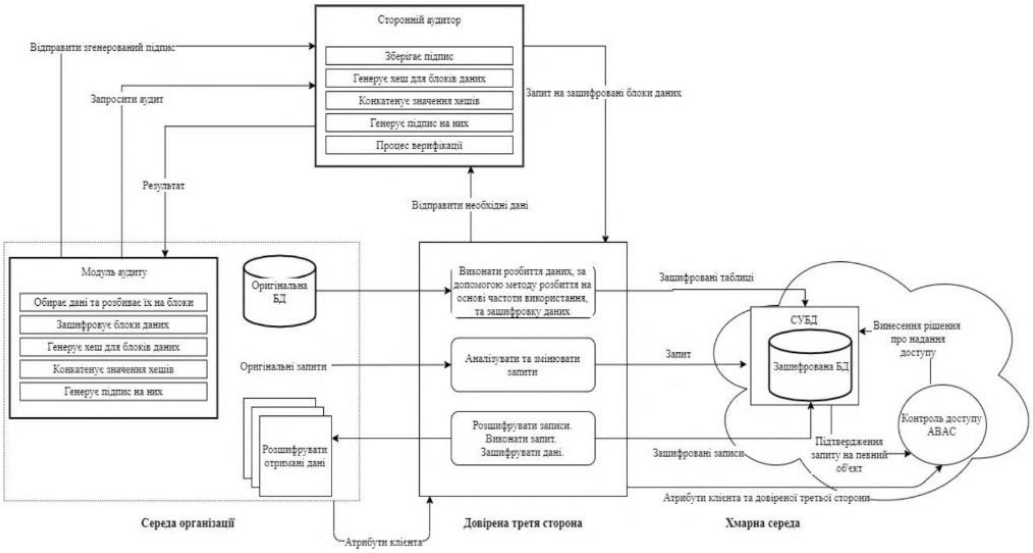

Рисунок 10 - Структура системи захисту хмарної бази даних

4 Безпека життєдіяльності, основи хорони праці

4.1 Вимоги безпеки щодо організації робочих місць

Конструкція робочого місця, його розміри та взаємне розташування його елементні повинні відповідати антропометричним, фізіологічним та психофізіологічним характеристикам людини, а також облаштоване згідно з вимогами стандартів робоче місце. Це досягається регулюванням положення крісла, висоти та кута нахилу підставки для ніг за умови її використання, або висоти та розмірів робочої поверхні. Повинне забезпечуватись виконання трудових операцій в зонах моторного поля (оптимальної досяжності, легкої досяжності та досяжності) в залежності від необхідної точності і частоти дій.

Організація робочих місць повинна забезпечувати стійке положення та вільність рухів працівника, безпеку виконання трудових операцій, виключати або допускати лише в деяких випадках роботу в незручних позиціях, котрі зумовлюють підвищену втомлюваність.

Загальні принципи організації робочого місця:

- на робочому місці не повинно бути нічого зайвого; всі необхідні для роботи предмети повинні знаходитись поряд з працівником, але не заважати йому;
- ті предмети, котрими користуються частіше, розташовуються ближче, ніж ті предмети, котрими користуються рідше;
- предмети, котрі беруть лівою рукою, повинні знаходитись зліва, а ті предмети, котрі беруть правою рукою, повинні знаходитись справа,
- якщо використовують обидві руки, то місце розташування пристосувань вибирається з врахуванням зручності захоплювання його двома руками;
- небезпечніше, з точки зору можливості травмування працівника обладнання повніше розташовуватись вище, ніж менш небезпечне Однак слід враховувати, що важкі предмети під час роботи зручніше та легше опускати, ніж піднімати;

- робоче місце не повинне захаращуватися заготовками і готовими деталями;
- організація робочого місця повинна забезпечувати необхідну оглядовість. Засоби відображення інформації повинні бути розташовані в зонах інформаційного поля робочого місця з врахуванням частоти та значущості інформації, типу засобів відображення інформації, точності і швидкості спостереження та зчитування.

4.2 Природні загрози, характер їхніх проявів та дії на людей, тварин, рослин

До природних небезпек відносяться стихійні явища, які являють безпосередню загрозу для життя та здоров'я людей. Наприклад, землетруси, виверження вулканів, снігові лавини, селі, зсуви, каменепади, повені, шторми, цунамі, тропічні циклони, смерчі, блискавки, тумани, космічні випромінювання і багато інших явищ. Будучи природними феноменами життя та розвитку природного середовища вони в той же час сприймаються людиною як аномальні. У безпеці життєдіяльності розглядаються не всі природні катастрофи і стихійні явища, а лише ті з них, які можуть завдати шкоди здоров'ю або призвести до загибелі людей.

Деякі природні небезпеки порушують або утруднюють нормальне функціонування систем та органів людини. До таких небезпек відноситься, наприклад, туман, ожеледиця, спека, холод, спрага.

Незважаючи на глибокі відмінності, по суті всі природні небезпеки підпорядковуються деяким загальним закономірностям.

По-перше, для кожного виду небезпек характерна певна просторова приуроченість. По-друге, встановлено, що чим більша інтенсивність (потужність) небезпечного явища, тим рідше воно трапляється. По-третє, кожному виду небезпек передують певні специфічні ознаки (передвісники). По-четверте, за всієї непередбачуваності тієї чи іншої природної небезпеки, її прояв

може бути передбачений. Насамкінець, по-п’яте, у багатьох випадках можуть бути передбачені пасивні та активні захисні заходи від природних небезпек.

Розглядаючи природні небезпеки, потрібно відзначити роль антропогенного впливу на їх прояв. Відомі численні факти порушення рівноваги у природному середовищі в результаті діяльності людства, які призводять до посилення небезпечного впливу. Так, згідно даних міжнародної статистики, походження близько 80 % сучасних зсувів пов’язане із діяльністю людини. У результаті вирубок лісу зростає активність селів, збільшуються паводкові витрати.

Нині масштаби використання природних ресурсів суттєво зросли. Це призвело до того, що стали відчутно виявлятися риси глобальної екологічної кризи. Природа наче мстить людині за грубе вторгнення у її володіння. Про це 200 років тому попереджав видатний англійський економіст Мальтус Томас Роберт (1766—1834), виклавши у праці «Опыт о законе народонаселения» (1798) свою концепцію про те, що механізмом регуляції людських популяцій стануть епідемії, тобто фактори, що залежать від густоти населення. Над цією проблемою людство почало серйозно замислюватися тільки останнім часом. Дотримання природної рівноваги є найважливішим профілактичним фактором, урахування якого дає змогу скоротити кількість небезпечних явищ.

Між природними небезпеками існує взаємозв’язок. Одне явище може правити за причину, спускний механізм для наступних явищ.

Наприклад, землетрус може викликати снігові лавини, дощі та снігопади, повені, водну ерозію, селі, зсуви, гірські обвали та каменепади, шторми, тайфуни та припливи.

За наявними оцінками, кількість природних явищ на Землі з плином часу не зростає або майже не зростає, але людські жертви та матеріальна шкода збільшуються. Щорічна імовірність загибелі мешканця планети Земля від природних небезпек орієнтовно дорівнює 10-5, тобто на кожні сто тисяч мешканців гине одна людина. Передумовою успішного захисту від міських небезпек є вивчення їх причин та механізмів. Знаючи суть процесів, можна їх передбачувати. А своєчасний та точний прогноз небезпечних явищ є

найважливішою передумовою ефективного захисту. Захист від природних небезпек може бути активним (будівництво інженерно-технічних споруд, інтервенція та механізм явища, мобілізація природних ресурсів, реконструкція природних об’єктів) та пасивної (використання укриттів). У більшості випадків активні та пасивні методи поєднуються.

За локалізацією природні небезпеки можуть бути з певною мірою умовності поділені на 4 групи: літосферні (землетруси, вулкани, зсуви);небезпеки гідросфери (повені, цунамі, шторми) атмосферні (урагани, бурі, смерчі, град, дощ); космічні (астероїди, планети, випромінювання).

4.3 Висновок до розділу безпека життєдіяльності, основи охорони праці

В четвертому розділі кваліфікаційної роботи висвітлено питання БЖД та ОП із застосуванням для практичної роботи.

В першому питані описано вимоги безпеки щодо організації робочих місць. Вжито заходи щодо зручного розташування столів і інших пристроїв.

Друге питання було розглянуто дію природних загроз на організм людина та знижує дію на працю людини.

Висновки

В даній роботі було проаналізовано поняття хмарних обчислень і технологій, та архітектуру з основними властивостями. Також виявлено актуальність у використані. Безпеку інформації провайдери не ставлять у перше місце, через це появляється вразливості що були розглянуті в другому розділі. Потрібно будувати таку систему щоб вона не пропускала ці вразливості.

Ми виявили найкращі технологічні рішення щоб складають безпеку хмарних даних таких як:

- Аудиту: схема аудиту для третьої сторони
- Контролю доступу: атрибути щоб керувати доступом
- Криптографічний захист безпеки даних: в системі Віктора Тепло

Також був запропонований метод що дає приріст швидкості запитів.

Результати можна використати щоб побудувати систему захисту інформації для будь яких компаній.

ПЕРЕЛІК ДЖЕРЕЛ

1. Вимоги безпеки щодо організації робочих місць [Електронний ресурс] - https://studopedia.org/7-164693.html
2. Природні загрози, характер їхніх проявів та дії на людей, тварин, рослин [Електронний ресурс] - https://pidru4niki.com/1628041450814/bzhd/prirodni_zagrozi_harakter_yihnih_proyaviv_diyi_lyudey_tvarin_roslin_obyekti_ekonomik
3. Wadiwala R. Cloud Database – DbaaS [Електронний ресурс] - https://labs.sogeti.com/cloud-database-dbaas-database-as-a-service/
4. PaaS, DBaaS, SaaS… Что все это значит? [Електронний ресурс] - https://habr.com/ru/company/kingservers/blog/310022/
5. Липак Х. Формування консолідованого інформаційного ресурсу за допомогою хмарних технологій: Праці 2 міжнар. наук.-техн. конф. (Львів, 9—12 жовт. 2018). Львів, 2018. С. 157—160.
6. Miller R. Who Has the Most Web Servers? [Електронний ресурс] - https://www.datacenterknowledge.com/archives/2009/05/14/whos-got-the-most-web-servers/
7. McRee R. PTA: Practical threat analysis [Text] / R. McRee // Proceedings of the Information Systems Security Association, London, September - pp. 37-40
8. Demchenko Y. Defining inter-cloud architecture for interoperability and integration [Text] / Y. Demchenko, C. Ngo, M.X. Makkes, R. Strijkers, C. de Laat // Proceeding of the 3rd International Conference on Cloud Computing, GRIDs and Virtualization, Nice, Francewww.matbio.org/2013/Oplachko_8_449.pdf - pp. 174-180
9. Who Has the Most Web Servers? [Електронний ресурс] - https://www.datacenterknowledge.com/archives/2009/05/14/whos-got-the-most-web-servers

10. Оплачко Е.С. Облачньїе технологии и их применение в задачах вьічислительной биологии [Електронний ресурс] - http://www.matbio.org/2013/Oplachko_8_449.pdf
11. Загрози інформаційної безпеки хмарних технологій. [Електронний ресурс] - https://qipu.ru/uk/yota/ugrozy-informacionnoi-bezopasnosti-oblachnyh-tehnologii-informacionnaya.html
12. Можливості та недоліки хмарних обчислень [Електронний ресурс] - https://sites.google.com/site/hmarnitekhnolohiyi/mozlivosti-ta-nedoliki-hmarnih-obcislen
13. Вибір системи контролю та управління доступом [Електронний ресурс] - https://ssbb.com.ua/uk/sistemy-kontrolya-dostupa/sistema-kontrolyu-dostupu/vybor-sistemy-kontrolya-i-upravleniya-dostupom/
14. Система контролю доступу доступом [Електронний ресурс] - https://www.elvis.com.ua/ua/access-ua.html
15. Система шифрування БД для СУБД Firebird [Електронний ресурс] - https://cipher.com.ua/uk/products/system-encrypt-db