**БЕЗПЕКА БАЗИ ДАНИХ І ВИВЧЕННЯ МЕТОДІВ ШИФРУВАННЯ ДАНИХ В ХМАРНОМУ СХОВИЩІ.**

**к.т.н., проф. А.М. Філоненко , магістр Є.М. Щербініна , магістр Б.В. Марценюк, НТУ «ХПІ», м. Харків**

Безпека даних - найважливіше завдання в сучасному світі. Влада, компанії та інші організації втратили багато грошей, а багато інших закрилися, через діяльність хакерів і зловмисників. За минулі роки були розроблені різні схеми шифрування для захисту бази даних від атак зловмисників. Оскільки дані - це життєвий канал кожної організації, існує потреба в безпечному зберіганні даних, щодня генерованих цими організаціями. Хмарне сховище необхідно для віддаленого зберігання даних.

Для багатьох підприємств безпека даних є однією з основних проблем при відправці файлів в хмару. Вони турбуються про те, що їх файли будуть переглянуті або навіть скомпрометовані.

Для захисту цілісності даних, що зберігаються, необхідні методи шифрування даних. У минулому багато компаній відчували себе комфортно, дозволяючи постачальникам хмарних послуг управляти всіма своїми даними, вважаючи, що ризиками безпеки можна керувати за допомогою контрактів, засобів контролю і аудиту. Однак з часом стало очевидно, що постачальники хмарних послуг не можуть виконувати такі зобов'язання.

У цьому документі обговорюється важливість шифрування бази даних і дається детальний огляд різних методів шифрування.

**Ключові слова:** шифрування, криптографія, хешування, безпека, база даних, хмарне сховище, зашифрований текст.

**I. Вступ**

У століття технологій всю нашу роботу виконують комп'ютери. Від спілкування з друзями в соціальних мережах до здійснення онлайн-платежів через Інтернет-банкінг - все робиться онлайн через комп'ютери. Оскільки ці засоби ефективні і полегшують нашу роботу, ми використовуємо їх так чи інакше. Це означає, що при використанні цих онлайн-сервісів ми зберігаємо всі наші особисті та конфіденційні дані в базах даних цих веб-сайтів і додатків, що робить ці дані уразливими. Таким чином, захист цих важливих для користувача даних є одним з основних пріоритетів щоб уникнути неправомірного використання даних [1].

Хмарне сховище - це система віддаленого зберігання, обслуговування, управління і резервного копіювання даних. Послуга доступна користувачам через мережу, якою зазвичай є Інтернет.

Важливий спосіб захисту цих даних - це шифрування даних, що зберігаються в базах даних цих веб-сайтів.

**II. Навіщо потрібне шифрування даних**

Необхідність шифрування даних перед їх збереженням в базі даних полягає в тому, що обмеження доступу за допомогою авторизації і аутентифікації даних може допомогти до певної межі, але, що якщо зловмисник якимось чином потрапить до бази даних. У нього є всі дані бази даних, і він може використовувати їх на свій розсуд, тут вступає в гру шифрування даних перед їх збереженням в базі даних. Якщо дані зашифровані перед збереженням в базі даних, навіть з доступом до бази даних, зловмисник не може використовувати ці дані.

Тож, **метою** статті є обговорення важливості шифрування бази даних та детальний огляд різних методів шифрування.

**III. Шифрування бази даних**

Шифрування бази даних - це процес шифрування даних в базі даних [2]. Це ключова стратегія захисту вмісту даних в базі даних. Основна ідея полягає в тому, що зловмисник якимось чином зможе дістатися до бази даних системи; через шифрування він не зможе зловживати даними в базі даних.

На рис.1 показана основна робота процесу шифрування і дешифрування бази даних. Простий текст / дані, які будуть збережені в базі даних, спочатку перетворюються в зашифрований текст з використанням відповідного алгоритму і певного ключа. Потім цей зашифрований текст зберігається в базі даних. Коли користувач хоче отримати дані з бази даних, зашифрований текст перетворюється назад в простий текст з використанням алгоритму дешифрування і того ж ключа, який використовувався при шифруванні. Це поверне користувачеві простий текст за запитом.



Рисунок 1 – Процес шифрування і дешифрування бази даних.

Шифрування бази даних можна виконати двома способами:

Шифрування. Це процес, в якому простий текст перетворюється в зашифрований текст за допомогою ключа, а потім, використовуючи той же ключ, ми можемо розшифрувати зашифрований текст назад в простий текст [3]. Шифрування виконується з використанням різних алгоритмів, кожен з яких має свої переваги і недоліки. Найчастіше використовуються алгоритми шифрування DES, RC2, AES128, AES 256 і т. Д. На рис.2 показано виконання простого процесу шифрування.

Хешування: це односторонній процес, в якому простий текст перетворюється в хешоване значення (зашифрована форма). Після того, як дані хешируются за допомогою функції хешування, їх не можна змінити назад на звичайний текст [3]. Зазвичай цей підхід використовується для шифрування паролів, щоразу, коли нам потрібно увійти в систему, введений пароль шифрується з використанням хеш-функції, а потім зіставляється з паролем, що зберігаються в базі даних, яка вже знаходиться в зашифрованою формі, якщо обидва збігаються користувачі отримують доступ, інакше він отримує повідомлення про невірне ім’я користувача/пароль. Найбільш часто використовувані хеш-функції - це MD4, MD5, SHA, SHA-1 і т. Д. На рис. 3 показана робота хешування.

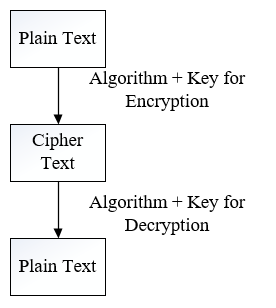


Рисунок 2 – Процес шифрування



Рисунок 3 – Процес хешування

**IV. Шифрування даних хмари**

Хмарне шифрування - це послуга, запропонована постачальниками хмарного сховища, при якій дані або текст перетворюються з використанням алгоритмів шифрування і потім поміщаються в хмару сховища. Хмарне шифрування практично ідентично внутрішньому шифрування з однією важливою відмінністю: клієнтові хмарного сервісу необхідний час, щоб дізнатися про політику і процедуру для шифрування і управління ключами шифрування. Можливості хмарного шифрування постачальника послуг повинні відповідати рівню конфіденційності розміщуваних даних. Оскільки шифрування споживає більше ресурсів процесора, багато постачальників хмарних послуг пропонують базове шифрування тільки для декількох полів бази даних, таких як паролі та номери облікових записів [4]. У цей момент використання провайдером шифрування всієї бази даних клієнта може стати настільки дорогим, що може мати сенс зберігати дані всередині компанії або зашифрувати дані перед їх відправкою в хмару. Щоб знизити витрати, деякі постачальники хмарних послуг пропонують альтернативи шифруванню, які не вимагають такої обчислювальної потужності. Ці методи включають в себе редагування або обфускацію даних, які повинні залишатися конфіденційними, або використання власних алгоритмів шифрування, створених постачальником [5].

**V. Методи хмарного шифрування даних**

- Блокові шифри і потокові шифри

Один з основних методів категоризації для широко використовуваних методів шифрування заснований на формі вхідних даних, з якими вони працюють. Це два типи: блоковий шифр і потоковий шифр. У цьому розділі обговорюються основні функції двох типів, робочий режим, і порівнюються їх з точки зору безпеки і продуктивності.

Блоковий шифр

У цьому методі шифрування дані шифруються і дешифруються, якщо дані знаходяться в одному з блоків. У найпростішому режимі ви ділите простий текст на блоки, які потім передаються в систему шифрування для створення блоків зашифрованого тексту. ECB (режим електронної кодової книги) - це основна форма блочного шифру, в якій блоки даних зашифровуються безпосередньо для генерації відповідних зашифрованих блоків.

Потокові шифри

Потоковий шифр працює з потоком даних, працюючи з ним біт за бітом. Потоковий шифр складається з двох основних компонентів: генератора ключового потоку і функції змішування. Функція змішування зазвичай являє собою просто функцію XOR, в той час як генератор потоку ключів є основним блоком в техніці шифрування потокового шифрування. Наприклад, якщо генератор ключового потоку видає серію нулів, що виводиться зашифрований потік буде ідентичний вихідному простому тексту.

- Симетричне та асиметричне шифрування

Процедури шифрування даних в основному діляться на дві категорії в залежності від типу ключів безпеки, які використовуються для зашифровування / розшифрування захищених дані. Ці дві категорії: Асиметричні і симетричні методи шифрування.

Симетричне шифрування

У цьому типі шифрування відправник і одержувач узгоджують секретний (загальний) ключ. Потім вони використовують цей секретний ключ для шифрування і дешифрування своїх відправлених повідомлень. На рис. 4 показаний процес симетричної криптографії. Вузли A і B спочатку узгоджують метод шифрування, який буде використовуватися при шифрування і дешифрування даних, що передаються. Потім вони домовляються про секретні ключі, які вони обидва будуть використовувати в зв'язку з цим. Після завершення налаштування шифрування вузол A починає відправляти свої дані, зашифровані за допомогою загального ключа, на іншій стороні вузол B використовує той же ключ для дешифрування зашифрованих повідомлень.

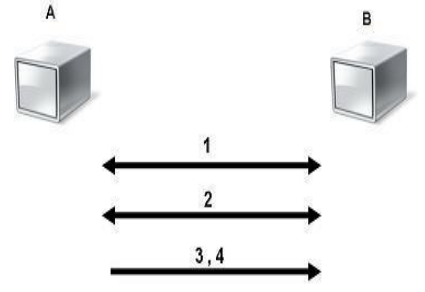


Рисунок 4 – Симетричне шифрування.

Основна проблема симетричного шифрування полягає в тому, як безпечно розділити секретний ключ між двома вузлами. Якщо ключ стає відомим, з якої-небудь причини, вся система валиться. Управління ключами для цього типу шифрування проблематично, особливо якщо унікальний секретний ключ використовується для кожного однорангового з'єднання, тоді загальна кількість секретних ключів, які будуть збережені для n вузлів, буде n (n- 1) / 2.

Асиметричне шифрування

Воно також відоме як криптографія з відкритим ключем (PKC), тому що користувачі зазвичай використовують два ключі: відкритий ключ, який відомий всім, і закритий ключ, який відомий тільки користувачеві. На рис.5 нижче показано використання двох ключів між вузлом A і вузлом B. Після узгодження типу шифрування, який буде використовуватися в поєднанні, вузол B відправляє свій відкритий ключ вузла A. Вузол A використовує отриманий відкритий ключ для шифрування його повідомлення. Потім, коли приходять зашифровані повідомлення, вузол B використовує свій закритий ключ для їх розшифровки.

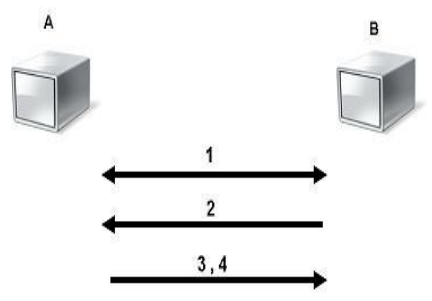


Рисунок 5 – Асиметричне шифрування.

Ця можливість вирішує проблему симетричного шифрування управління секретними ключами. Але з іншого боку, ця унікальна особливість шифрування з відкритим ключем робить його математично більш уразливим для атак. Більш того, методи асиметричного шифрування майже в 1000 разів повільніші, ніж методи симетричного шифрування, оскільки вони вимагають більшої обчислювальної потужності. Щоб отримати переваги обох методів, зазвичай використовується гібридна техніка. У цьому методі асиметричне шифрування використовується для обміну секретним ключем, а потім використовується симетричне шифрування для передачі даних між відправником і отримувачем [6].

Наприклад, у відкритій системі, враховуючи будь-які два принципала X і Y, X повинен бути в змозі зашифрувати повідомлення, яке може бути розшифровано тільки Y. Якщо існує деяка прив'язка між основними ідентифікаторами і відкритими ключами, то ці операції можуть легко бути виконаним. Наївна схема могла б працювати в такий спосіб:

1. принципал X шукає відкритий ключ KY для принципала Y та використовує його для обчислення деякої функції ;
2. потім Y, отримавши це повідомлення, обчислює .

Але у цієї схеми є суттєва проблема, з огляду на наші визначення безпеки для шифрування з загальним ключем: вона не задовольняє семантичної безпеки, оскільки зловмисник легко обчислить та і порівняє їх із заданими шифротекстами в різних моделях атак. І знову ми бачимо, що без імовірнісного шифрування не може бути семантичної безпеки. Це особливо вірно при налаштуванні відкритого ключа, оскільки кожен принципал за визначенням має доступ до функції шифрування для кожного іншого принципала. Особливо, коли простір можливих повідомлень невеликий, легко просто перевірити всі повідомлення з допомогою функції шифрування, щоб з'ясувати, що було зашифровано.

**Висновки**

З розвитком технологій в даний час все робиться за допомогою комп'ютерів, тому безпека цих даних в базі даних стає важливою проблемою. Багато дослідників працювали над цим і пропонували різні алгоритми і архітектури. У кожної схеми є свої переваги і недоліки. Але жоден з них не є повністю безпечним і містить певні лазівки або недоліки, які можуть використовуватися хакерами і зловмисниками для отримання доступу до бази даних. Так що в цій області є можливості для поліпшення.

Багато проблеми ще належить визначити. Криптографічні методи використовуються для забезпечення безпечного зв'язку між користувачем і хмарою. Симетричне шифрування швидке і обчислювально ефективне, щоб обробляти шифрування великих обсягів даних в хмарному сховищі. У цій статті запропоновано алгоритм симетричного шифрування для безпечного зберігання даних користувача в хмарному сховищі. Пропонований алгоритм шифрування описаний детально, а процес дешифрування є зворотним шифруванню. Цей алгоритм використовується для шифрування даних користувача в хмарі. Оскільки після виходу з сеансу користувач не може контролювати дані, ключ шифрування діє як первинна аутентифікація для користувача.

**Список літератури**

1. Бараані-Дастджерді, Ахмад, Йозеф Пепшік і Рейхані Сафаві-Найн. «Безпека в базах даних: дослідження». Департамент комп'ютерних наук Університету Вуллонгонга (1996 г.).
2. Денні Черрі і Томас Ларок, «2 - Шифрування бази даних при захисті SQL-сервера», під редакцією Денні Черрі, Томас Ларок, Syngress, Бостон, 2011 р, сторінки 27-71, ISBN: 9781597496254.
3. Кесслер, Гері К. «Огляд криптографії». (2003). (Http://www.sciencedirect.com/science/article/pii/B97815974962541000 22).
4. Вамс Крішна, Ярлагадда і Шрірам Рамануджі, «Безпека даних в хмарних обчисленнях», Журнал комп'ютерних і математичних наук, том 2 (1), стор. 15-23, 2011 р
5. Пітер Мелл, Тім Гранса, «Ефективне і безпечне використання парадигми хмарних обчислень», NIST, Лабораторія інформаційних технологій, http://www.csrc.nist.gov/groups/SNS/cloud-computing / cloudcomputingv26.ppt. 2009 р
6. Еман М. Мохамед, Хатем С. Абделькадер і Шериф Ель-Етрібі, «Модель безпеки даних для хмарних обчислень», Дванадцята міжнародна конференція по мережах, ISBN: 978-1-61208-245-5, стор 66-74, 2013 .