УДК 004.056.55

М. А. Кушта
Ю. Є. Яремчук

ВДОСКОНАЛЕНА МОДЕЛЬ CP-ABE ДЛЯ КОНТРОЛЮ ДОСТУПУ ДО ХМАРНИХ РЕСУРСІВ

Вінницький національний технічний університет

Анотація

У роботі розглянуто підхід до організації контролю доступу до хмарних ресурсів із застосуванням вдосконаленої моделі CP-ABE. Основну увагу приділено формалізації вимог, моделі загроз, архітектурним рішенням, механізму версійності атрибутів та алгоритмам шифрування, розшифрування і відкликання доступу. Запропоновано використання гібридного шифрування, за якого хмарний ресурс захищається симетричним ключем, а сам ключ шифрується за політикою CP-ABE. Механізм версійності атрибутів дає змогу обмежувати подальший доступ користувачів після відкликання прав без повного перешифрування всіх ресурсів.

Ключові слова: хмарні ресурси, контроль доступу, CP-ABE, атрибутне шифрування, гібридне шифрування, версійність атрибутів, відкликання доступу.

Abstract

The paper considers an approach to organizing access control to cloud resources using an improved CP-ABE model. The main focus is placed on the formalization of requirements, the threat model, architectural solutions, the attribute versioning mechanism, and the algorithms for encryption, decryption, and access revocation. A hybrid encryption approach is proposed, in which a cloud resource is protected with a symmetric key, while the key itself is encrypted according to a CP-ABE access policy. The attribute versioning mechanism makes it possible to restrict further access for users after privilege revocation without full re-encryption of all resources.

Keywords: cloud resources, access control, CP-ABE, attribute-based encryption, hybrid encryption, attribute versioning, access revocation.

Вступ

Хмарні ресурси стали основою зберігання та оброблення службових документів, персональних даних, резервних копій, криптографічних ключів і журналів подій. Водночас перенесення даних у хмару посилює залежність власника інформації від правильності налаштування IAM-політик, стану облікових записів, API-ключів і хмарної інфраструктури [1], [10]. У разі помилкового надання прав або компрометації окремого компонента адміністративний контроль доступу може виявитися недостатнім.

Традиційні моделі DAC, MAC, RBAC та ABAC зручні для організаційного керування правами, однак вони здебільшого покладаються на рішення серверного компонента авторизації [2]. Тому для захисту конфіденційних хмарних ресурсів доцільно доповнювати адміністративні механізми криптографічним розмежуванням доступу. У такій моделі користувач отримує дані не лише після формального дозволу системи, а лише за умови можливості виконати процедуру розшифрування [3], [4].

Одним із перспективних підходів є CP-ABE, тобто шифрування на основі атрибутів із політикою доступу, вбудованою у шифротекст [4], [5]. У цій моделі власник ресурсу задає політику доступу під час шифрування, а користувач може розшифрувати захищений ключ лише тоді, коли його набір атрибутів відповідає заданим умовам [3]. Для практичного використання у хмарних середовищах базову CP-ABE-модель потрібно доповнювати механізмами динамічного керування атрибутами, відкликання доступу та інтеграції з хмарним сховищем [6-9].

Метою роботи є обґрунтування підходу до контролю доступу до хмарних ресурсів із застосуванням вдосконаленої моделі CP-ABE.

Результати дослідження

Формалізація запропонованого підходу починається з визначення вимог і моделі загроз. Хмарне сховище в межах запропонованого підходу розглядається як середовище, якому не можна повністю довіряти, що узгоджується з принципами хмарної безпеки та керування ризиками [1], [10]. Воно може коректно зберігати об'єкти та метадані, але не повинно отримувати доступ до відкритого вмісту файлів або ключового матеріалу. Тому в системі передбачено, що у хмарі зберігаються лише зашифровані ресурси, зашифровані ключі, політики доступу та службові ідентифікатори.

До функціональних вимог системи належать автентифікація користувачів, керування атрибутами, створення політик доступу, шифрування ресурсів, перевірка права доступу, розшифрування, відкликання атрибутів і ведення журналу подій. Безпекові вимоги охоплюють криптографічне виконання політики доступу, недоступність відкритого файлу для хмарного сховища, підтримку актуального стану атрибутів, фіксацію спроб доступу та неможливість отримання ресурсу користувачем із відкликаними правами [2], [10].

Удосконалена модель CP-ABE будується навколо трьох основних множин: множини користувачів U, множини ресурсів R та множини атрибутів A. Для користувача u у момент часу t формується актуальна множина атрибутів $S_u^{(t)}$, яка містить не лише назви атрибутів, а й їхні версії, строки чинності та ознаку активності. Ресурс r пов'язується з політикою P_r, що описує логічну умову доступу. Доступ дозволяється лише тоді, коли $S_u^{(t)}$ задовольняє P_r, усі потрібні атрибути активні, їхні версії відповідають актуальним значенням, а час запиту належить до дозволеного інтервалу [3-5], [7].

Формально умову доступу можна подати як $Access(u, r, t) = 1$, якщо $S_u^{(t)} \vDash P_r$, $attr.version = current\_version(attr)$, $attr.is\_active = true$ та $valid\_from \leq t \leq valid\_to$. Якщо хоча б одна з цих умов не виконується, система не повинна надавати ключ розшифрування. Така формалізація дозволяє поєднати гнучкість ABAC-підходу з криптографічним виконанням політики CP-ABE [2], [4], [5].

Запропонована архітектура ґрунтується на модульному принципі. До її складу входять модуль автентифікації, Attribute Manager, Policy Manager, Cryptographic Engine, Revocation Manager, Cloud Storage Adapter, реляційна база даних і Audit Logger. Модуль автентифікації відповідає за встановлення особи користувача. Attribute Manager підтримує набір атрибутів, їхні версії, строки чинності та статус. Policy Manager забезпечує створення й перевірку політик доступу. Cryptographic Engine виконує гібридне шифрування та розшифрування. Revocation Manager оновлює версії атрибутів або деактивує їх. Cloud Storage Adapter ізолює взаємодію з хмарним сховищем від внутрішньої логіки системи, а Audit Logger фіксує критичні події [6-9], [12].

Запропонована архітектура не передбачає створення нового криптографічного примітива. Її авторське значення полягає у прикладному поєднанні CP-ABE, гібридного шифрування, версійності атрибутів і журналювання в єдиному циклі контролю доступу до хмарного ресурсу [4], [5], [7]. Це важливо для бакалаврської роботи, оскільки акцент робиться не на математичному доведенні нової схеми, а на проєктуванні працездатної системи, придатної для програмної реалізації та тестування.

Алгоритм шифрування ресурсу складається з кількох послідовних етапів. Спочатку власник або адміністратор обирає файл і задає політику доступу P_r, наприклад: role = admin OR (department = security AND clearance = high). Далі генерується випадковий симетричний ключ K_r, яким шифрується сам файл. Після цього ключ K_r шифрується за допомогою CP-ABE відповідно до політики P_r, а до бази даних записуються ідентифікатор ресурсу, посилання на зашифрований файл, політика доступу, зашифрований ключ і службові метадані [4], [5], [11].

Алгоритм перевірки доступу та розшифрування виконується у зворотному порядку. Користувач надсилає запит на ресурс, система отримує його актуальні атрибути з бази даних, перевіряє їхні версії, строки чинності та статус активності. Якщо атрибути задовольняють політику, CP-ABE-механізм відновлює симетричний ключ K_r, після чого файл розшифровується симетричним алгоритмом. Якщо атрибут відкликаний, має застарілу версію або не входить до політики, операція завершується відмовою з фіксацією події в журналі аудиту [3-5], [7].

Механізм відкликання доступу ґрунтується на зміні стану атрибутів. Для відкликання права користувача система може деактивувати конкретний атрибут, змінити його строк чинності або збільшити номер поточної версії атрибута. Після цього користувач зі старим набором атрибутів не задовольняє

актуальну політику доступу [7-9]. Водночас слід враховувати практичне обмеження: система може блокувати наступні операції розшифрування, але не може технічно “відібрати” відкриту копію файлу, яку користувач уже отримав до моменту відкликання.

Інформаційне забезпечення системи доцільно реалізувати у вигляді реляційної бази даних. Центральними сутностями є users, attributes, user_attributes, resources, policies, encrypted_keys, revocation_log та audit_log. Таблиця users зберігає облікові записи користувачів, attributes - довідник атрибутів і поточні версії, user_attributes - прив’язку атрибутів до користувачів із часовими параметрами, resources - метадані зашифрованих ресурсів, policies - логічні умови доступу, encrypted_keys - зашифровані ключі ресурсів, revocation_log - історію відкликання, а audit_log - спроби доступу та результати операцій [12].

Для підтвердження прикладної придатності запропонованого підходу обґрунтовано використання Python, SQLite, бібліотеки cryptography та веб-інтерфейсу. Python спрощує реалізацію логіки системи й тестових сценаріїв, SQLite достатня для локального прототипу та демонстрації структури даних, а cryptography забезпечує надійні засоби симетричного шифрування [11], [12]. У межах прототипу CP-ABE-компонент може бути поданий як логічний модуль перевірки політик і захисту ключового матеріалу, що дозволяє продемонструвати принцип роботи системи без надмірного ускладнення програмної частини.

Таблиця 1 - Узагальнення проєктних рішень системи контролю доступу

Проєктне рішення	Основне призначення	Очікуваний результат
Модель загроз	Врахування недовіри до хмарного сховища, компрометації облікових записів і помилкових політик	Формування вимог до криптографічного контролю доступу
Версійність атрибутів	Зберігання актуальної версії, строків чинності та статусу атрибута	Можливість відкликання прав без повного перешифрування всіх ресурсів
Гібридне шифрування	Шифрування файлу симетричним ключем і CP-ABE-захист ключа	Зменшення обчислювальних витрат при захисті великих файлів
Модульна архітектура	Поділ системи на модулі автентифікації, атрибутів, політик, криптографії, відкликання та аудиту	Спрощення реалізації, тестування та подальшого розширення системи
Реляційна база даних	Зберігання користувачів, атрибутів, політик, ресурсів, ключів і журналів	Цілісна інформаційна основа для роботи програмного прототипу
Журналювання подій	Фіксація спроб доступу, шифрування, розшифрування та відкликання	Підтримка аудиту та аналізу інцидентів безпеки

Отже, результати проєктування показують, що вдосконалена CP-ABE-модель може бути використана як основа для побудови системи контролю доступу до хмарних ресурсів, у якій доступ визначається не лише адміністративним дозволом, а й криптографічною можливістю розшифрування ресурсу [3-5]. Поєднання атрибутних політик, версійності атрибутів і гібридного шифрування дає змогу підвищити захищеність даних у середовищі, де хмарне сховище розглядається як потенційно недовірене [7-10].

Висновки

У результаті дослідження обґрунтовано доцільність застосування вдосконаленої моделі CP-ABE для контролю доступу до хмарних ресурсів. Запропонований підхід дозволяє перенести ключову частину перевірки доступу з адміністративного рівня на криптографічний рівень, що зменшує ризик розкриття даних у разі помилкових IAM-політик або часткової компрометації серверної інфраструктури [1], [4], [5].

Запропонована архітектура системи поєднує модулі керування атрибутами, формування політик доступу, шифрування, розшифрування, відкликання доступу, взаємодії з хмарним сховищем і

журналювання подій. Такий поділ забезпечує логічну зрозумілість системи та створює основу для реалізації програмного прототипу.

Механізм версійності атрибутів і часових параметрів доступу дозволяє реалізувати практичне відкликання прав користувачів. У разі зміни версії або деактивації атрибута користувач втрачає можливість виконати наступне розшифрування ресурсу, якщо його актуальні атрибути більше не відповідають політиці доступу [7-9].

Використання гібридного шифрування є доцільним для хмарних ресурсів, оскільки CP-ABE застосовується лише для захисту симетричного ключа, а основні дані шифруються швидшим симетричним алгоритмом. Це дає змогу зменшити обчислювальне навантаження та зробити запропоновану систему придатною для практичного застосування [4], [5], [11].

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Mell P., Grance T. The NIST Definition of Cloud Computing. NIST Special Publication 800-145. 2011.
2. Hu V. C., Ferraiolo D., Kuhn R., Schnitzer A., Sandlin K., Miller R., Scarfone K. Guide to Attribute Based Access Control (ABAC) Definition and Considerations. NIST Special Publication 800-162. 2014.
3. Goyal V., Pandey O., Sahai A., Waters B. Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data. Proceedings of the 13th ACM Conference on Computer and Communications Security. 2006.
4. Bethencourt J., Sahai A., Waters B. Ciphertext-Policy Attribute-Based Encryption. IEEE Symposium on Security and Privacy. 2007.
5. Waters B. Ciphertext-Policy Attribute-Based Encryption: An Expressive, Efficient, and Provably Secure Realization. Public Key Cryptography. 2011.
6. Li J., Fan Y., Bian X., Yuan Q. Online/Offline MA-CP-ABE with Cryptographic Reverse Firewalls for IoT. Entropy. 2023. Vol. 25. No. 4.
7. Cianfriglia M., Onofri E., Pedicini M. mRLWE-CP-ABE: A revocable CP-ABE for post-quantum cryptography. Journal of Mathematical Cryptology. 2024. Vol. 18. No. 1.
8. Ren Z., Yan E., Chen T., Yu Y. Blockchain-based CP-ABE data sharing and privacy-preserving scheme using distributed KMS and zero-knowledge proof. Journal of King Saud University - Computer and Information Sciences. 2024. Vol. 36. No. 3.
9. Tian J. Zero trust anonymous access algorithm for multi cloud storage system based on CP-ABE. Egyptian Informatics Journal. 2025. Vol. 30.
10. ISO/IEC 27001:2022. Information security, cybersecurity and privacy protection - Information security management systems - Requirements.
11. Python Cryptography Authority. Cryptography documentation [Електронний ресурс]. Режим доступу: https://cryptography.io
12. SQLite Documentation [Електронний ресурс]. Режим доступу: https://www.sqlite.org/docs.html

Кушта Максим Андрійович - студент групи 2КІТС-22б, факультет менеджменту та інформаційної безпеки, Вінницький національний технічний університет, м. Вінниця, email:

Яремчук Юрій Євгенович - доктор технічних наук, професор, професор кафедри менеджменту та безпеки інформаційних систем, директор Центру інформаційних технологій і захисту інформації, науковий керівник науково-дослідної лабораторії технічного захисту інформації, академік Академії наук вищої освіти України, Вінницький національний технічний університет, м. Вінниця, email: yurevyar@vntu.edu.ua.

Kushta Maksym A. - student of group 2KITS-22b, Faculty of Management and Information Security, Vinnytsia National Technical University, Vinnytsia, email:

Yaremchuk Yurii Ye. - Doctor of Technical Sciences, Professor, Professor of the Department of Management and Security of Information Systems, Director of the Center for Information Technologies and Information Protection, Scientific Supervisor of the Research Laboratory of Technical Information Protection, Academician of the Academy of Sciences of Higher Education of Ukraine, Vinnytsia National Technical University, Vinnytsia, email: yurevyar@vntu.edu.ua