УДК 004.056.77

ПРОПОЗИЦІЯ ВАРІАНТА ФРЕЙМВОРКУ З БЕЗПЕЧНОГО КЕРУВАННЯ КЛЮЧАМИ В ХМАРНИХ СЕРЕДОВИЩАХ

М. А. Тарасов[1,a], О. В. Козленко[1]

[1] Навчально-науковий Фізико-технічний інститут

Анотація

В цій роботі розглянуті сервіси управління ключами найпопулярніших хмарних провайдерів та проблем пов'язаних із керуванням ключів у багатохмарному середовищі. Метою даної роботи було розроблення варіанта фреймворку для безпечного управління ключами на основі вже існуючих фреймворків безпеки та рекомендацій від хмарних провайдерів. Реалізовано прототип програмного рішення для виявлення й усунення конфігураційних помилок відносно сервісів керування ключами.

Ключові слова: фреймворк безпеки, хмарні середовища, сервіси керування ключами (KMS), контролі безпеки, ротація ключів,

Вступ

Криптографічні ключі — це ключі, що шифрують і розшифровують конфіденційні дані, автентифікують користувачів і пристрої, а також забезпечують безпечні канали зв'язку. Процес керування ключами відноситься до створення, розповсюдження, зберігання, підтримки, резервного копіювання, відновлення, анулювання та знищення криптографічних ключів для захисту цифрових активів організації [1]. Для полегшення процесу їхнього керування було розроблено систему керування ключами. Key Management System (KMS) — це система, яка побудована навколо криптографічного модуля та використовує криптографічні функції і керування життєвим циклом ключів [2].

Із поширенням використання хмарних провайдерів та їхніх послуг серед організацій за останні роки, підхід до зберігання, обробки та керування даними, які зберігаються на віддалених хмарних ресурсах, змінюється. Таке поширення користування хмарними послугами можна пояснити перевагами, які надають хмарні провайдери, а саме

- Масштабованість та гнучкість ресурсів
- Економія коштів
- Швидке розгортання ресурсів
- Висока відмовостійкість
- Відсутність потреби в обслуговуванні

Масовий перехід на використання послуг, що надають хмарні провайдери, створює нові ризики, пов'язані з віддаленим зберіганням та опрацюванням даних, що підвищує вразливість до кібератак. Оскільки такі дані зберігаються віддалено, організації не можуть мати полного контролю над такими ресурсами. Саме тому впровадження надійних заходів захисту, таких як шифрування за допомогою криптографічних ключів, є критично важливим для забезпечення безпеки інформації у хмарних середовищах.

[a] m.tarasov.w@gmail.com

1. Ситеми керування ключами (KMS) від хмарних провайдерів

В даній роботі було розглянуто сервіси керування ключами від трьох найпопулярніших хмарних провайдерів. На основі дослідження [3] були обрані наступні провайдери (рис. 1):

- Amazon Web Services (AWS)
- Microsoft Azure
- Google Cloud Platform (GCP)

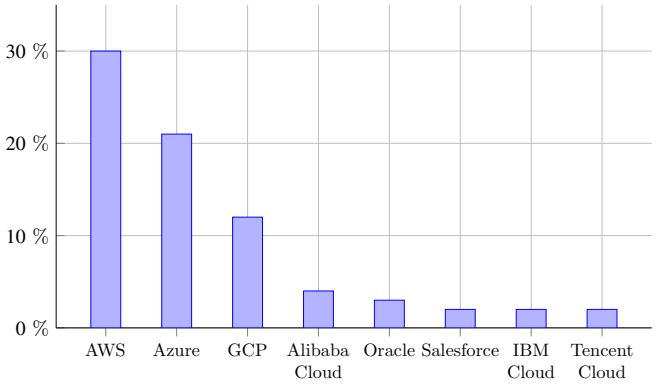

Рис. 1. Частка світового ринку провідних постачальників хмарних послуг в 4 кварталі 2024 року

Кожен із цих провайдерів пропонує свої власні сервіси керування ключами:

- **AWS KMS** — сервіс AWS для створення та керування ключами шифрування, що захищають дані. Він інтегрується з іншими сервісами AWS і використовує апаратні модулі безпеки для надійного захисту ключів [4].

- **Azure Key Vault** — хмарна служба Azure для безпечного зберігання ключів, паролів і сертифікатів. Вона забезпечує зручне керування секретами та їх захист за допомогою апаратних модулів безпеки [5].
- **GCP Cloud KMS** — сервіс Google Cloud для керування криптографічними ключами в хмарних додатках. Він підтримує створення ключів і їх інтеграцію з сервісами Google Cloud [6].

2. Проблеми пов'язані з процесами безпечного керування ключами у мульти-клауд середовищі

Основні проблеми, пов'язані з керуванням ключами в мульти-клауд середовищі включають відсутність стандартизованих політик безпеки для керування ключами, складнощі в їх управлінні у різних хмарних середовищах, нечіткий розподіл відповідальності між хмарними провайдерами та користувачами, а також зростання зацікавленності кіберзловмисників до хмарних інфраструктур [7].

- **Відсутність стандартизованих політик безпеки** — Недостатність уніфікованих фреймворків безпеки для захисту сервісів керування ключами, що враховують специфічність та унікальність різних сервісів керування ключами у хмарних провайдерах, призводить до недостатку уніфікованих політик та контролів безпеки, які б покривали усі аспекти кожного KMS.
- **Складнощі керування ключами в хмарних середовищах** — Кожен KMS має власний функціонал та вимоги до безпечного керування ключами, що призводить до складнощів із безпечного керування ключами в різних хмарних середовищах. Окрім цього, хмарні середовища постійно розвиваються та наповнюються новим функціоналом, що підкреслює важливість адаптуватися до нових нововведень для ефективного та безпечного керування ключами.
- **Нечіткий розподіл відповідальності** — Модель спільної відповідальності в хмарних середовищах ускладнює визначення ролей провайдерів і клієнтів у забезпеченні безпеки ключів. Це може призвести до помилок в конфігурації або до недостатніх заходів з безпеки хмарного середовища.
- **Зростання зацікавленності кіберзловмисників до хмарних інфраструктур** — Із поширенням використання хмарних технологій серед організацій, сервіси керування ключами стають привабливою ціллю для зловмисників.

Враховуючи проблеми, які були зазначені вище, існує потреба у створенні фреймворку безпеки для мульти-клауд середовища, який врахує специфіку кожного KMS у різних хмарних провайдерах та зможе ефективно керувати ключами відповідно до найкращих практик безпеки.

3. Концепція фреймворку з безпечного керування ключами в хмарних середовищах

3.1. Структура фреймворку

У даній роботі запропоновано варіант фреймворку із безпечного керування ключами, структура якого наведена нижче:

- **Політика безпеки** — політика, що містить рекомендації щодо безпечного процесу керування ключами, яка буде заснована на вже відомих фреймворках безпеки для KMS, такі як:
  - NIST
  - CSA Cloud Security Aliance
  - CIS Benchmark
  - Trend Micro

  А також буде враховано рекомендації із безпечного керування ключами від самих провайдерів, а саме:
  - AWS KMS best practices
  - Best practices for using Azure KeyVault
  - GCP Best practices for using CMEKs
- **Контролі безпеки** — файл у форматі YAML, який буде містити відповідні контролі для перевірки виконання політик безпеки для AWS, Azure та GCP, а також інструкції для автоматичного виправлення непройдених контролів.
- **Програмна реалізація перевірки контролів** — за допомогою мови програмування Python було реалізовано прототип програми, що буде опрацьовувати наданий YAML файл із контролями безпеки, і перевіряти відповідність цим контролям на усіх трьох провайдерах. У випадку невідповідності, користувачу буде запропоновано одразу виправити такі місконфігурації.

Під час реалізації фрагменту даного фреймворку, процес автентифікації та авторизації у Azure та GCP провайдерах не було імплементовано. Відповідно, для перевірки працездатності програми, відповідним функціям для перевірки контролів було передано тестові відповіді (mock API responses) у якості вхідних аргументів.

3.2. Розглянутий контроль безпеки

В демонстраційних цілях, відповідно до існуючих фреймворків безпеки KMS, а також до «best-security practices» рекомендацій для кожного провайдера, було створено наступний контроль:

- «Переконайтеся, що автоматична ротація ключів увімкнена в службах KMS (≤ 90 днів)»

Ротація ключів — це процес створення нових ключів шифрування для заміни існуючих. Роблячи ротацію ключів шифрування регулярно або після певних подій, потенційні наслідки компрометації ключа будуть зменшені [8]. Період, зазначений в контролі, було обрано як оптимальний із усіх опрацьованих рекомендацій.

3.3. Структура YAML файлу

Структура YAML файлу складається із переліку контролів, які необхідно перевірити. Кожен контроль містить у собі секції для кожного провайдера (якщо такий контроль застосовний до цього KMS), а також наступні секції:

- **input_data** — визначає що буде відправлятись на API ендпоінти хмарних провайдерів для отримання необхідної інформації для подальшої перевірки контролю.
- **positive_results** — визначає умову, яка буде перевірятись на основі отриманої попередньо відповіді для визначення чи був пройдений контроль.
- **remediation** — необхідна інформація, яка, у випадку згоди користувача, буде надсилатись на API ендпоінти хмарного провайдера для виправлення непройденого контролю.

На (рис. 2) зображено фрагмент вмісту такого YAML файлу.

```yaml
AWS:
  input_data:
    x_amz_request: TrentService.GetKeyRotationStatus
    post_body: '{"KeyId": "<CUSTOM_IDENTIFIER_KEY_ID>"}'
  positive_results:
    operand: contains
    key:
      - KeyRotationEnabled
      - RotationPeriodInDays
    value:
      - True
      - '<91'
  remediation:
    remediation_text: Enable automatic key rotation (90d)
    x_amz_request: TrentService.EnableKeyRotation
    post_body: '{"KeyId": "<CUSTOM_IDENTIFIER_KEY_ID>", "RotationPeriodInDays": 90 }'
```

Рис. 2. YAML файл із контролем перевірки автоматичної ротації ключів для AWS KMS

3.4. Результати роботи програми

Нижче наведено результати виконання програми, що перевіряє відповідність до зазначеного вище контролю ротації ключів на двох провайдерах — AWS та Azure. На (рис. 3) та (рис. 4) зображено результати перевірки контролю статусу ротації ключів на AWS KMS. Приклад перевірки контролю на Microsoft Azure провайдері зображено на (рис. 5).

```
Performing: Ensure that automatic key rotation is enabled on the KMS services (<= 90d)

[+]-------------------------------------------------[+]
{'KeyId': 'arn:aws:kms:eu-north-1:535002858286:key/a54669ca-9a9c-47bf-8171-e895491d0f67
', 'KeyRotationEnabled': False}

Check was failed for the a54669ca-9a9c-47bf-8171-e895491d0f67 KMS key id
Wanna do remediation action "Enable automatic key rotation (90d)"? (Y/N):
```

Рис. 3. Перевірка завершилась невдачею, оскільки ротація ключів не увімкнена

```
{'KeyId': 'arn:aws:kms:eu-north-1:535002858286:key/c113c0b1-e0b1-4ad9-915d-549dc24372d8
', 'KeyRotationEnabled': True, 'NextRotationDate': 1755163698.153, 'RotationPeriodInDay
s': 365}
365

Check was failed for the c113c0b1-e0b1-4ad9-915d-549dc24372d8 KMS key id
Wanna do remediation action "Enable automatic key rotation (90d)"? (Y/N):
```

Рис. 4. Перевірка завершилась невдачею, оскільки період ротації ключів довший ніж у контролі

```
GET https://mock-keyvault.vault.azure.net/abcd-efgh-
1234-5678/rotationpolicy?api-version=7.4
Check passed for abcd-efgh-1234-5678 KMS key

GET https://mock-keyvault.vault.azure.net/ijkl-mnop-
9012-3456/rotationpolicy?api-version=7.4
Check passed for ijkl-mnop-9012-3456 KMS key
```

Рис. 5. Перевірка на основі mock-відповідей від Azure пройшла успішно

Висновки

У межах цієї роботи було проведено ознайомлення із поняттями криптографічних ключів та системами їхнього керування. Було досліджено найпоширеніші сервіси управління ключами в хмарних середовищах. Проаналізовано проблеми що виникають під час управління ключами в багатохмарних середовищах. На основі цих проблем запропоновано фреймворк для безпечного керування ключами, який включає контролі безпеки у YAML форматі. Контролі засновані на вже існуючих фреймворках безпеки та рекомендаціях від провайдерів. Розроблено код для перевірки та виправлення виявлених помилок конфігурації на основі цих контролів.

Перелік використаних джерел

1. *Cloud Security Alliance*. Key Management Lifecycle Best Practices / Cloud Security Alliance. — 15.12.2023. — URL: https://cloudsecurityalliance.org/download/artifacts/key-management-lifecycle-best-practices.
2. *Cloud Security Alliance*. Key Management in Cloud Services / Cloud Security Alliance. — 11.09.2020. — URL: https://cloudsecurityalliance.org/download/artifacts/key-management-when-using-cloud-services.
3. *Statista*. Chart: Amazon Maintains Cloud Lead as Microsoft Edges Closer. — 27.02.2025. — URL: https://www.statista.com/chart/18819/worldwide-market-share-of-leading-cloud-infrastructure-service-providers/.
4. *Amazon Web Services*. AWS Key Management Service Developer Guide: Overview. — 2025. — URL: https://docs.aws.amazon.com/kms/latest/developerguide/overview.html.
5. *Microsoft*. Azure Key Vault basic concepts. — 2025. — URL: https://learn.microsoft.com/en-us/azure/key-vault/general/basic-conceptss.
6. *Google Cloud*. Cloud Key Management Service Documentation. — 2025. — URL: https://cloud.google.com/kms/docs/key-management-service.
7. Machine learning-based intelligent security framework for secure cloud key management / S. Ahmad, S. Mehfuz, S. Urooj, N. Alsubaie // Cluster Computing. — 2024. — Т. 27. — С. 5953—5979.
8. *Google Cloud*. Key rotation in Cloud Key Management Service. — 2025. — URL: https://cloud.google.com/kms/docs/key-rotation.