ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ЗАХИСТУ ІНФОРМАЦІЇ

КАФЕДРА ІНФОРМАЦІЙНОЇ ТА КІБЕРНЕТИЧНОЇ БЕЗПЕКИ

КВАЛІФІКАЦІЙНА РОБОТА

на тему:

«ТЕХНОЛОГІЇ ЗАХИСТУ ХМАРНОЇ ІНФРАСТРУКТУРИ AMAZON AWS ВІД ВЕБ-ЗАГРОЗ»

на здобуття освітнього ступеня магістра

зі спеціальності _____125_____Кібербезпека_____

(код, найменування спеціальності)

освітньо-професійної програми_Інформаційна та кібернетична безпека____

(назва)

*Кваліфікаційна робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело*_____АНДРУЩЕНКО Максим

Виконав: здобувач вищої освіти групи БСДМ-62

АНДРУЩЕНКО Максим
(ПРІЗВИЩЕ, ім'я)

Керівник
к.т.н, доцент СОБЧУК Андрій
(ПРІЗВИЩЕ, ім'я)

Рецензент
к.т.н, доцент _____
(ПРІЗВИЩЕ, ім'я)

КИЇВ – 2024

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ

НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ЗАХИСТУ ІНФОРМАЦІЇ

Кафедра Інформаційної та кібернетичної безпеки

Ступінь вищої освіти Магістр

Спеціальність 125 Кібербезпека

Освітньо-професійна програма Інформаційна та кібернетична безпека

ЗАТВЕРДЖУЮ
Завідувач кафедри ІКБ
_____Гайдур Г.І
«___»_____2023 року

З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Андрущенку Максиму Вікторовичу
(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи: «Технології захисту хмарної інфраструктури Amazon AWS від веб-загроз»

керівник кваліфікаційної роботи Собчук А.В., к.т.н, доцент кафедри
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом Державного університету інформаційно-комунікаційних технологій від «19» жовтня 2023р. №145.

2. Строк подання студентом кваліфікаційної роботи 15.12.2023 р.

3. Вихідні дані до кваліфікаційної роботи

1) Веб загрози;

2)Хмарна інфраструктура Amazon AWS ;

3) Наукова та технічна література. Стандарти. Рекомендації.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

1) Аналіз особливостей розгортання хмарної інфраструктури;

2) Виокремлення поширених загроз, спрямованих на хмарні технології та інфраструктуру;

3) Механізми та засоби захисту хмарної інфраструктури Amazon AWS від веб-загроз.

5. Перелік ілюстративного матеріалу:

1) Мета, об'єкт та предмет дослідження;

2) Дослідження інфраструктури ресурсів AWS;

3) Веб-загрози в хмарному середовищі;

4) Огляд вразливостей реалізація Amazon S3;

5) Аналіз витоку облікових даних для запису AWS;

6) Дослідження механізмів та засобів захисту хмарної інфраструктури Amazon AWS від веб-загроз;

7) Дослідження особливостей використання AWS CloudTrail;

8) Рекомендації щодо забезпечення безпеки в Amazon Web Services (AWS) від веб-загроз;

9) Висновки.

6. Дата видачі завдання 19.10.2022 р.

КАЛЕНДАРНИЙ ПЛАН

№ зп	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1.	Аналіз науково-технічної літератури	05.11.2023 р.	виконано
2.	Аналіз особливостей розгортання хмарної інфраструктури	12.11.2023 р.	виконано
3.	Виокремлення поширених загроз, спрямованих на хмарні технології та інфраструктуру	17.11.2023 р.	виконано
4.	Дослідження служб безпеки AWS	21.11.2023 р.	виконано
5.	Аналіз рішення AWS CloudTrail	24.11.2023 р.	виконано
6.	Виокремлення рекомендацій щодо створення та налаштування безпечного облікового запису в AWS	27.11.2023 р.	виконано
7.	Реферат, вступ, висновки	29.11.2023 р.	виконано
8.	Підготовка презентації	01.12.2023 р.	виконано

Здобувач вищої освіти _____ Максим АНДРУЩЕНКО
(підпис) *(Ім'я, ПРІЗВИЩЕ)*

Керівник кваліфікаційної роботи _____ Андрій СОБЧУК
(підпис) *(Ім'я, ПРІЗВИЩЕ)*

РЕФЕРАТ

Текстова частина магістерської роботи: 68 сторінок, 42 рисунка, 1 таблиця, 24 джерел.

Об'єкт дослідження – захист хмарної інфраструктури від веб-загроз.

Предмет дослідження – технології захисту хмарної інфраструктури AMAZON AWS від веб-загроз .

Мета роботи – розробка рекомендацій щодо захисту хмарної інфраструктури Amazon AWS від потенційних веб-загроз.

Методи дослідження – теорія інформації, стандарти у сфері кібербезпеки, політики безпеки, практичне тестування програмного забезпечення.

В роботі проаналізовано хмарні технології та хмарну інфраструктуру. Зазначено основні загрози безпеці хмарних інфраструктури, що включають: порушення даних, втрата даних, викрадення облікових записів, незахищені API, відмова в обслуговуванні, зловмисні інсайдери, зловживання хмарними службами, тощо.

Досліджено інфраструктурні ресурси Amazon Web Services (AWS). Приведено огляд служб безпеки AWS, що включають: управління ідентифікацією та доступом AWS, віртуальну приватну хмару AWS, систему управління ключами (KMS), Shield AWS, брандмауер веб-додатків AWS (WAF), AWS CloudTrail, AWS CloudWatch, AWS Config, артефакт AWS, тощо.

Виокремлено алгоритм налаштування механізмів розшифрування файлів CloudTrail. Розроблено рекомендації щодо забезпечення безпеки в Amazon Web Services (AWS) від веб-загроз.

Галузь використання – кібербезпека.

AMAZON WEB SERVICES, AWS, ІНФРАСТРУКТУРНІ РЕСУРСИ, ВЕБ-ЗАГРОЗИ, CLOUDTRAIL, БЕЗПЕКА, БРАНДМАУЕР, ЛОГУВАННЯ, ВРАЗЛИВОСТІ, ВЕБ-ДОДАТКІВ, CLOUDWATCH, AWS CONFIG, АУТЕНТИФІКАЦІЯ.

ABSTRACT

Master's thesis: 68 pages, 42 figures, 1 table, 24 sources.

The object of research – protection of cloud infrastructure against web threats.

The subject of research – technologies for protecting the Amazon AWS cloud infrastructure against web threats.

The aim of research – to develop recommendations for protecting the Amazon AWS cloud infrastructure against potential web threats

Research methods – information theory, cybersecurity standards, security policies, practical software testing.

The work analyzes cloud technologies and cloud infrastructure. It identifies the main threats to the security of cloud infrastructure, including data breaches, data loss, account hijacking, insecure APIs, denial of service, malicious insiders, abuse of cloud services, etc.

The infrastructure resources of Amazon Web Services (AWS) have been examined. An overview of AWS security services is provided, including AWS Identity and Access Management, AWS Virtual Private Cloud, Key Management System (KMS), AWS Shield, AWS Web Application Firewall (WAF), AWS CloudTrail, AWS CloudWatch, AWS Config, AWS Artifact, etc.

A specific algorithm for configuring CloudTrail file decryption mechanisms is highlighted. Recommendations for ensuring security in Amazon Web Services (AWS) against web threats are developed.

Field of use – cybersecurity.

AMAZON WEB SERVICES, AWS, INFRASTRUCTURE RESOURCES, WEB THREATS, CLOUDTRAIL, SECURITY, FIREWALL, LOGGING, VULNERABILITIES, WEB APPLICATIONS, CLOUDWATCH, AWS CONFIG, AUTHENTICATION.

ЗМІСТ

Стор.

ВСТУП ……..……... 9

1 **АНАЛІЗ ОСОБЛИВОСТЕЙ РОЗГОРТАННЯ ХМАРНОЇ ІНФРАСТРУКТУРИ**……. 11
1.1 Особливості розгортання хмарної інфраструктури……..... 11
1.2 Аналіз поширених загроз, спрямованих на хмарні технології та інфраструктуру……... 14
1.3 Виокремлення моделі порушника хмарної інфраструктури…… 19
1.4 Основні рішення AWS……. 21
Висновки до розділу 1…… 25

2 **ДОСЛІДЖЕННЯ ВРАЗЛИВОСТЕЙ ТА ВЕБ-ЗАГРОЗ ХМАРНІЙ ІНФРАСТРУКТУРІ AMAZON AWS**……...……... 27
2.1 Дослідження інфрастурктурних ресурсів Amazon Web Services (AWS)…… 27
2.2 Виокремлення критичних аспектів забезпечення безпеки ресурсів Amazon Web Services (AWS) …….. 29
2.3 Дослідження особливостей реалізації Amazon S3…… 31
2.4 Аналіз особливостей протидії криптоджекінгу та електронному скімінгу в сегментах Amazon S3…… 37
2.5 Дослідження інструментів впровадження контейнерів в хмарній інфраструктурі……. 39
2.6 Аналіз витоку облікових даних для запису AWS…….. 47
Висновки до розділу 2…… 50

3 **ДОСЛІДЖЕННЯ МЕХАНІЗМІВ ТА ЗАСОБІВ ЗАХИСТУ ХМАРНОЇ ІНФРАСТРУКТУРИ AMAZON AWS ВІД ВЕБ-ЗАГРОЗ** …… 52
3.1 Дослідження служб безпеки AWS…….. 52
3.2 Виокремлення основних обов'язків користувачів щодо безпеки… 55
3.3 Дослідження особливостей використання AWS CloudTrail…… 56
3.4 Виокремлення рекомендацій щодо створення та налаштування безпечного облікового запису в AWS…… 65
3.5 Розробка рекомендацій щодо забезпечення безпеки в Amazon Web Services (AWS) від веб-загроз…… 73
Висновки до розділу 3…… 74

ВИСНОВКИ …….. 76
ПЕРЕЛІК ПОСИЛАНЬ 78
ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ (Презентація) 81

ВСТУП

Актуальність дослідження. Однією з найбільш суттєвих перешкод для широкого впровадження хмарних сервісів є сприйняття проблеми безпеки в хмарних обчисленнях. Серед сучасних загроз можна виділити порушення контролю доступу, порушення цілісності повідомлень, можливість витоку даних, недостатність гарантій повного видалення даних, можливість введення шкідливого коду, використання зловмисного програмного забезпечення та нестачу досвіду у галузі хмарних технологій.

Кроки щодо дослідження та сертифікації хмарних служб з питань безпеки є достатньо ефективними заходами спрямованими на подолання зростаючої тривожності громадськості щодо збереження конфіденційності, цілісності та доступності даних, що зберігаються у хмарних середовищах.

Декілька чинних моделей безпеки хмарних інфраструктур, призначених для боротьби із загрозами, включають шифрування всіх даних під час зберігання та передачі. Також активний моніторинг баз даних, файлів і URL-фільтри, заходи для запобігання втраті даних, виявились ефективними для виявлення та запобігання несанкціонованій міграції даних із хмар та всередині них. Необхідно зауважити, що ще не існує гарантій повного видалення даних від хмарних провайдерів на запит клієнтів, але сучасні технології, методи та засоби можуть стати розв’язанням цих проблем.

Вищенаведені аргументи актуалізують тему даної кваліфікаційної роботи, зміст якої становлять дослідження щодо технології захисту хмарної інфраструктури Amazon AWS від веб-загроз.

Об’єкт дослідження – безпечне функціонування хмарної інфраструктури Amazon AWS.

Предмет дослідження – механізми та засоби захисту хмарної інфраструктури Amazon AWS від веб-загроз.

Мета роботи – розробка рекомендацій щодо захисту хмарної інфраструктури Amazon AWS від потенційних веб-загроз.

Наукові завдання:

- проаналізувати хмарні технології та хмарну інфраструктуру;
- проаналізувати основні загрози безпеці хмарній інфраструктури;
- дослідити інфраструктуру Amazon Web Service (AWS);
- дослідити служби безпеки AWS;
- розробити рекомендації щодо забезпечення безпеки в Amazon Web Services (AWS) від веб-загроз.

Методи дослідження – теорія інформації, стандарти у сфері кібербезпеки, політики безпеки, практичне тестування програмного забезпечення.

Практичне значення одержаних результатів полягає в розробці рекомендації щодо забезпечення безпеки в Amazon Web Services (AWS) від веб-загроз.

Апробація результатів. Основні наукові результати роботи доповідалися та обговорювалися на Всеукраїнській науковій конференції «Актуальні проблеми кібербезпеки».

1 АНАЛІЗ ОСОБЛИВОСТЕЙ РОЗГОРТАННЯ ХМАРНОЇ ІНФРАСТРУКТУРИ

1.1. Особливості розгортання хмарної інфраструктури

Хмарна інфрастурктура представляють собою нову технологічну парадигму, спрямовану на спільне використання обчислювальних ресурсів з метою підвищення ефективності та зниження витрат на адміністрування та інші IT-витрати. Ця технологія надає зручний та повсюдний доступ до якісних послуг на вимогу шляхом створення еластичного пулу обчислювальних ресурсів, включаючи мережі, сервери, сховища, програми та сервіси. За визначенням Національного інституту стандартів і технологій (NIST), хмарна інфраструктура є моделлю для доступу до спільного пулу конфігурованих обчислювальних ресурсів через мережу на вимогу.

Amazon Web Service (AWS) вважається одним з перших гігантів хмарних інфраструктур, та й Eucalyptus став першою платформою з відкритим кодом для розгортання хмарних обчислень.

Завдяки своїй масштабованості, доступності та моделі оплати за фактом використання (Pay-As-You-Use), хмарні інфрастурктури стали особливо привабливими для початківців та компаній. Це означає, що компанії можуть почати використовувати великі обчислювальні ресурси всього за кілька хвилин, зареєструвавшись у постачальника хмарних послуг, і сплачуючи лише за фактичне використання, без значних початкових інвестицій в IT-персонал, інфраструктуру, програмне забезпечення та технічне обслуговування[2].

Існує чотири типи розгортання хмарних інфраструктур:

- **Приватна хмара.** Інфраструктура та керування розташовані на території конкретної організації і використовуються виключно для внутрішнього використання цією організацією;
- **Хмара спільноти.** Використовується кількома організаціями зі

спільною метою та може надаватися різними постачальниками хмарних послуг. Може бути розгорнута в будь-якому приміщенні організацій-учасниць або ззовні;

- **Публічна хмара.** Загальнодоступна інфраструктура, доступна для всіх, хто може зареєструвати облікові записи у постачальнику хмарних послуг;
- **Гібридна хмара.** Поєднання двох типів розгортання хмарних інфраструктур, наприклад, приватної хмари та доступу до загальнодоступних хмарних ресурсів.

На рис.1.2 представлена модель типів розгортання хмарних інфраструктур.

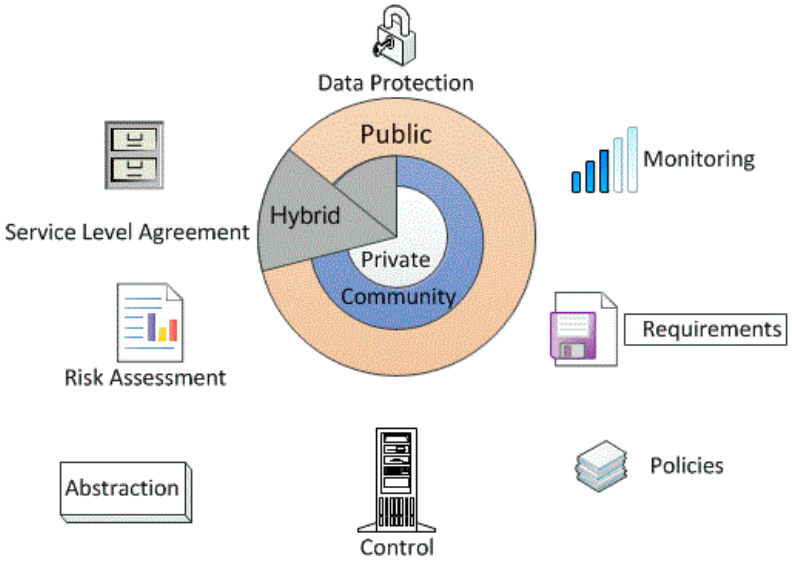


Рис.1.1. Типи розгортання хмарних інфраструктур

Типи хмарного розгортання включають три моделі надання послуг (SDM):

- **Інфраструктура як послуга (IaaS).** В даній моделі постачальники хмарних послуг та споживачі орендують та керують обчислювальними сховищами, мережами, операційними системами, програмним забезпеченням тощо. Споживач не має контролю над основною інфраструктурою.
- **Платформа як послуга (PaaS).** PaaS надає можливість хмарним споживачам розгортати програми та програмне забезпечення на платформі постачальника хмарних послуг.
- **Програмне забезпечення як послуга (SaaS).** Споживачі використовують програми, надані постачальниками хмарних послуг, такі як хмарні електронні листи, електронні таблиці, системи ERP, програми для розробки

інтерфейсу користувача тощо.

Деякі документи також відносять Composite as a Service до категорії хмарних SDM. Ця модель поширена серед постачальників хмарних послуг для підкреслення їхніх продуктів під назвами «N + aaS», наприклад, «Безпека як послуга (SaaS)».

На рис.1.3 представлені моделі надання хмарних послуг з різними постачальниками хмарних послуг і їхніми клієнтами.

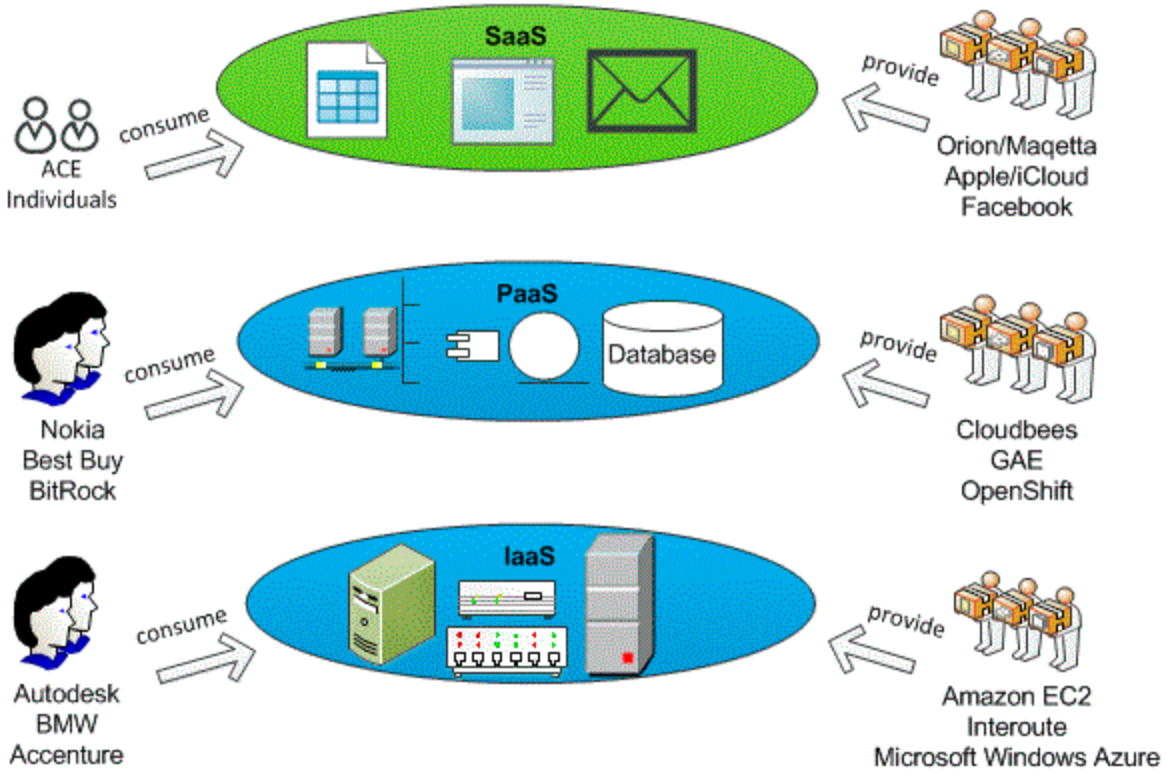


Рис.1.2. Моделі надання хмарних послуг, постачальники хмарних послуг та їхні клієнти

У ранні роки розвитку хмарних інфраструктур відсутні були стандарти. Різні постачальники послуг використовували різні фреймворки та платформи для реалізації своїх пропозицій на основі сервіс-орієнтованих архітектур, службових обчислень і віртуалізації. Однак з плином часу, зі зростанням попиту на сумісність хмар, збільшенням інтересу споживачів і зростанням обурення з питань безпеки та конфіденційності, виникла необхідність в стандартизації.

Організації стандартизації, такі як Cloud Security Alliance (CSA), National Institute of Standards and Technology (NIST), Open Cloud Manifesto (OCM), Federal

Information Security Management Act (FISMA) та Federal Risk and Authorization Management Program (FedRAMP), активно працюють над акредитацією, сертифікацією та стандартизацією хмарних інфраструктур.

За даними IDC 87,5% IT-директорів вважають безпеку та конфіденційність основними перешкодами для впровадження хмарних інфраструктур. Загроза безпеці через хмарну інфраструктуру для окремих осіб і компаній надзвичайно важлива, і це стало очевидним через події, такі як збій хмари Amazon та витік даних з хмари Google та Netflix. Зловмисники можуть використовувати хмарні інфраструктури для запуску шкідливих програм та вірусних атак на інші хмарні інфраструктури[3].

Згідно з CSA, у 2023 році існує дев'ять основних загроз безпеці хмарних інфраструктур: порушення даних, втрата даних, викрадення облікових записів, незахищені API, відмова в обслуговуванні, зловмисні інсайдери, зловживання хмарними службами, недостатній рівень обережності, спільні технологічні проблеми.

Додаткові виявлені загрози безпеці включають:

- Неправильна реалізація безпеки гіпервізора в неприватних хмарних середовищах IaaS.
- Неправильне використання брандмауера та неправильне впровадження брандмауера та контролю доступу.
- Переважне використання вразливих розподілених систем баз даних (DDS).
- Шкідливі програми.

1.2 Аналіз поширених загроз, спрямованих на хмарні технології та інфраструктуру

- Загрози IaaS

Зловживання хмарними сервісами. Фізична взаємодія при придбанні хмарних сервісів відсутня, і більшість постачальників хмарних послуг (CSP) не здійснюють

регулярний моніторинг активності користувачів IaaS через правила конфіденційності. У випадках, коли такі правила не діють, зловмисники можуть змінювати свої шаблони та стратегії з часом, ускладнюючи виявлення атак. Зловмисники з фальшивими або вкраденими банківськими реквізитами можуть придбати хмарні послуги IaaS та використовувати їх для обслуговування шкідливого програмного забезпечення або проведення фішингових атак на інших користувачів хмари. Це може перетворити хмару в найбільший ботнет. Ця загроза також стосується хмари PaaS і інфраструктури.

Відсутність досвіду в галузі хмарних технологій. Дослідження показали, що брак компетентного персоналу з питань безпеки ускладнює роботу з обсягами даних, зберіганих в хмарних середовищах. Додатково, проблеми виникають з безпекою хмари через неправильні налаштування безпеки та використання вразливих систем баз даних. Це може викликати порушення безпеки та конфіденційності користувачів хмари.

- **Загрози PaaS**

*Неможливість гарантувати повне видалення даних.*Завдяки розподіленій природі зберігання даних в хмарі та співпраці між різними хмарами, гарантування повного видалення даних клієнта на їх вимогу є проблематичним. Це може вплинути на кількох споживачів хмари та порушити конфіденційність.

Порушення цілісності повідомлень. Деякі CSP не застосовують належний рівень шифрування, інші не шифрують всі дані в хмарі через високі витрати або неможливість обробки зашифрованих даних належним чином. Це може призвести до порушень цілісності повідомлень та загрози конфіденційності користувачів хмари. Також важливо зауважити, що конфіденційність файлів користувачів, збережених в хмарі, може бути піддається загрозі від атаки дедуплікації, дозволяючи зловмисникам легко ідентифікувати файли інших користувачів.

- **Загрози SaaS**

Неадекватний моніторинг. У хмарному середовищі складно впроваджувати детальний аудит та відстеження через обмежені файли журналу. Правила конфіденційності також можуть створювати загрозу безпеці, оскільки більшість

клієнтів бажають, щоб їх діяльність контролювалася виключно для цілей нарахування та обліку.

- **Поширені загрози**

Проблеми автентифікації. Чинні системи автентифікації на основі паролів мають свої обмеження, які призводять до ризику. Розподіл довіри для контексту безпеки автентифікованого користувача між різними взаємодіючими хмарами ще не повністю реалізована в хмарних середовищах.

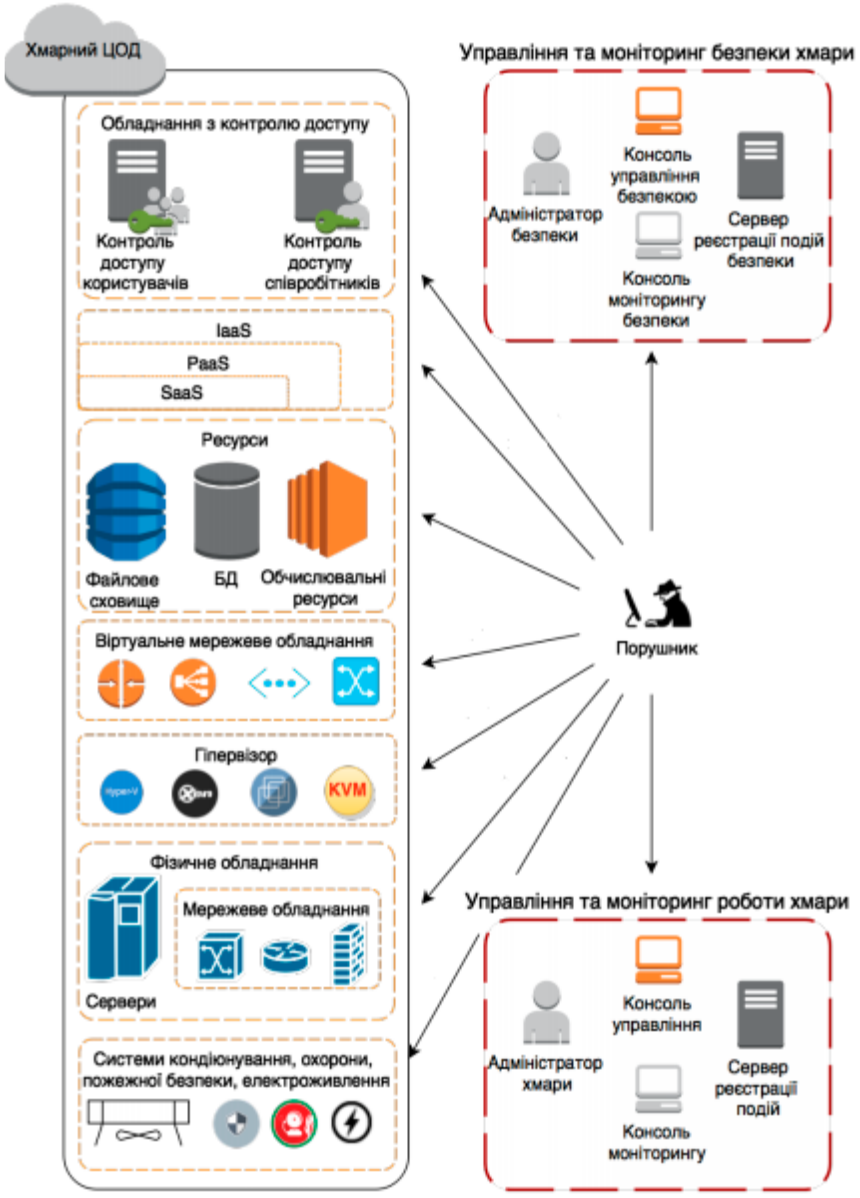


Рис.1.3. Модель поширених загроз

Найбільшу ймовірність мають загрози, що спрямовуються на складові компоненти хмарної інфраструктури, які мають інтерфейси доступу з зовні та/або знаходяться в віртуалізованому середовищі.

Витік даних. Характеристика мультиорендування хмарних середовищ створює значну загрозу витоку даних, як було вказано раніше. Документовані розриви віртуальних машин та обхід URL-адреси дозволяють зловмисникам здійснювати атаки на інших орендарів. Витік даних під час розриву хмари є поширеним явищем, особливо через різницю в вимогах безпеки та довіри між різними орендарями, що призводить до компромісу безпеки.

Незахищені інтерфейси та API. Більшість проблем із безпекою програмного забезпечення виникають через погану практику розробки програм. У хмарних середовищах також використовуються програми, які спочатку були призначені для внутрішнього використання, і розгортання таких програм може призводити до прийняття помилок, які були допущені у цих програмах. Системи DDS, такі як CouchDB, FlockDB, Hive, MongoDB, RavenDB і SimpleDB, використовують виклики API (NoSQL) для виконання операцій CRUD програми замість SQL. Існує документація щодо вразливостей у DDS, такі як обмежена підтримка інструментів для динамічного тестування додатків (DAST), відсутність підтримки нативного шифрування та хешування, а також загрози SQL-ін'єкцій.

Соціальна інженерія, використання експлойтів та внутрішні загрози. Інтернет насичений обманливим вмістом, який намагається переконати користувачів завантажити зловмисне програмне забезпечення. З цієї причини атаки соціальної інженерії, Drive-by exploits або Drive-by downloads залишаються значущими загрозами для безпеки хмарних інфраструктур.

Порушення контролю доступу. Здатність емулювати апаратне забезпечення за допомогою програмного забезпечення (віртуалізація) є однією з ключових технологій, що стоїть за хмарними інфраструктурами. Вона дозволяє запускати численні віртуальні машини (VM) на стеку програмного забезпечення емуляції (гіпервізора), створюючи можливість для мультиоренду та перерозподілу ресурсів для досягнення еластичності та масштабованості. Проте такий перерозподіл

ресурсів може призвести до того, що сектори, що містять дані одного орендаря, розміщені на віртуальній машині іншого співкористувача, можуть бути доступні для останнього. Крім того, IP-адреси віртуальних машин або хостів, звільнені під час переміщення серверів, можуть повернутися до публічних пулів та бути використані зловмисниками для доступу до хмарного середовища. Неоднорідність контролю доступу до хмари може стати серйозною проблемою безпеки. Порушення контролю доступу загрожують усім моделям надання хмарних обчислювальних послуг[4].

Таблиця 1.1.

Актуальні загрози безпеки постачальників хмарних послуг (CSP)

Постачальник	**Загрози безпеці CSP**
Cisco	Відсутність стандартів хмарних інфраструктур, недостатня безпека автоматизації хмари, відсутність автоматизації надання послуг безпеки, недостатнє управління ключами шифрування та надмірні витрати на шифрування.
Citrix	Адміністративні помилки та відсутність схвалених робочих процесів.
Expedient	Недостатні цілочисельні ланцюги довіри та недостатнє управління ключами шифрування.
HyTrust	Недостатня безпека віртуалізації, аутентифікація та контроль доступу.
McAfee	Проблеми з розподіленою зберіганням даних, проблеми з контролем доступу, закріплення споживачів хмари та неповне видалення даних.
Trapezoid	Гарантія безпеки даних незалежно від їх постійно змінного розташування.
Virtustream	Неповне видалення даних, проблеми з підтвердженням цілісності програмного забезпечення, недостатнє шифрування даних під час передачі та передача автентифікації додатків.

1.3 Виокремлення моделі порушника хмарної інфраструктури

У процесі створення схеми атак на хмарну інфраструктуру, використовується техніка, яка починається з розробки узагальненого представлення потенційного агресора, ідентифікації всіх можливих загроз, а потім визначення методів їх імплементації, закінчуючи формуванням схеми загроз. Ці схеми допомагають встановити критерії для захисної системи в межах хмарної інфраструктури.

Під схемою атаки мається на увазі абстрактне, формалізоване або невизначене зображення дій агресора, що відтворює його практичні і теоретичні можливості, попередні знання, час та місце проведення атаки і так далі. Викликом при розробці такої схеми для хмарної інформаційно-телекомунікаційної системи є потреба урахування багатьох змінних, включаючи модель впровадження хмари, сервісну модель, власника, рівень інформаційного контролю, а також контроль провайдера та користувачів над хмарною інфраструктурою. Схема атаки повинна також точно відображати дійсну поведінку агресора.

Рис. 1.4. Модель хмарних інфраструктур

Аналіз NIST показує, що в контексті хмарних ІТС атаки можуть здійснюватися як зсередини (наприклад, співробітниками або користувачами системи), так і ззовні (сторонніми особами), причому внутрішні атаки є особливо

небезпечними. Навіть на рівні інфраструктурних послуг (IaaS), внутрішня структура хмари, включаючи передачу даних, перебуває під контролем хмарного провайдера.

Рис.1.5 представляє схему хмарних інфраструктур, яка включає:

- різноманіття учасників, серед яких може з'явитися атакуючий суб'єкт;
- гіпотези щодо рівня компетенції, освіти, обізнаності з ІТС і характеру дій можливого атакувача;
- стратегії та інструментарій, які атакуючий суб'єкт може застосовувати;
- цілі, які атакуючий суб'єкт ставить перед собою;
- компоненти системи, на які атакуючий суб'єкт має намір напасти.

Категоризуючи атакуючих суб'єктів, можливими варіантами є:

- інтерні атакуючі: це можуть бути співробітники провайдера хмарних послуг, працівники клієнта, або зовнішні особи, які мають доступ до хмарних ІТ ресурсів. Щоб їх ідентифікувати, необхідно ретельно проаналізувати потенціал нелегітимного доступу до ІТ ресурсів кожного зі співробітників провайдера та клієнта, а також можливості зовнішніх осіб отримати нелегітимний доступ до ІТС хмари, враховуючи існуючу систему обмеження доступу;
- екстерні атакуючі: особи, що не мають прямого доступу до хмарних ІТ ресурсів. Їх ідентифікація вимагає глибокого аналізу потенційних витоків інформації та слабких місць у системі.

Щодо рівня кваліфікації атакуючих суб'єктів, їх можна розділити на чотири класи:

- на першому рівні атакуючий має змогу виконувати певний набір завдань (програм), які виконують передбачувані функції обробки даних;
- на другому рівні атакуючий може створювати і запускати свої програми для нових функцій обробки даних;
- на третьому рівні атакуючий має змогу управляти функціонуванням хмарної ІТС, тобто впливати на базове програмне забезпечення та конфігурацію обладнання системи;

- на четвертому рівні атакуючий володіє всіма можливостями, необхідними для проектування, втілення та ремонту апаратних компонентів ITC хмари, включно з можливістю додавання в систему власних пристроїв із новими функціями обробки даних[5].

1.4 Основні рішення AWS

Amazon Web Services (AWS) пропонує обширний набір інструментів для охорони робочих навантажень своїх клієнтів, проте клієнти часто не повністю усвідомлюють свою роль у забезпеченні безпеки та відповідальності за моніторинг та управління своїми даними у хмарі AWS.

AWS надає різноманітні сервіси та інструменти, такі як ідентифікація та доступ, шифрування, ведення журналів, нагляд та дотримання норм, для забезпечення безпеки в хмарному середовищі. Ці сервіси AWS дозволяють виконувати широкий спектр завдань для задоволення всіх вимог безпеки, реєстрації користувачів, аудиту та відповідності в хмарному середовищі.

AWS Identity and Access Management (IAM) дає можливість контролювати доступ до вашого облікового запису AWS, тоді як Virtual Private Cloud (VPC) дозволяє створити віртуальну мережу, яка пов'язана з власною мережею.

Ключові управління службами (KMS) дозволяють управління шифрувальними ключами для додаткового захисту даних. AWS Shield і AWS Web Application Firewall (WAF) захищають ресурси та програми AWS від розповсюджених загроз, таких як DDoS-атаки, через налаштовувані брандмауери. AWS Config, CloudTrail та CloudWatch забезпечують засоби для аудиту та управління конфігурацією вашого ресурсу AWS, тоді як AWS Artifact пропонує документи про відповідність, необхідні для аудитора.

Зараз безпека в хмарі вже не вважається перешкодою для перенесення даних, програм та робочих навантажень у хмару, а, навпаки, стала однією з основних причин для такого переходу. Безпека в хмарі часто сприймається як більш передова, надійна та економічно вигідна, ніж традиційна безпека в місцевих дата-

центрах. Ці висновки підкріплені даними з різних географічних регіонів та галузей з жорсткими вимогами безпеки, і вони стосуються керівників різного рангу та спеціалістів у сфері IT.

В результаті, адаптація хмарних технологій швидко зростає, особливо великими підприємствами, де безпека є вирішальним фактором. AWS пропонує інтегровані та уніфіковані рішення безпеки, що полегшують перехід клієнтів до хмарних сервісів. Прогнози від провідних аналітичних компаній, таких як Gartner та IDC, передбачають значне зростання хмарних обчислень. AWS позиціонується як одне з найбільш гнучких та безпечних хмарних рішень, знявши більшість забезпечення безпеки, яке зазвичай пов'язане з місцевою IT-інфраструктурою, і пропонуючи конфіденційність та вбудовані функції безпеки, які використовують процеси безпеки AWS.

Захист даних у хмарному середовищі має свої особливості порівняно з традиційними дата-центрами. У процесі перенесення серверів, даних і робочих навантажень на платформу AWS, виникає спільна відповідальність між клієнтом та AWS щодо захисту цих ресурсів. AWS займається захистом фундаментальної інфраструктури, що підтримує хмарні послуги, включаючи управління мережею регіонів, зон доступу, а також розміщення та кінцеві точки ресурсів.

З іншого боку, клієнти несуть відповідальність за захист вмісту, який вони розгортають у хмарі, такого як дані та додатки, а також за забезпечення безпеки з'єднань між хмарою і їх власними дата-центрами. Відповідно, клієнти також відповідальні за управління доступом до своєї віртуальної мережі та ресурсів у хмарі. Цей підхід називається моделлю спільної відповідальності за безпеку в AWS. На рис.1.5 продемонстровано концепцію моделі

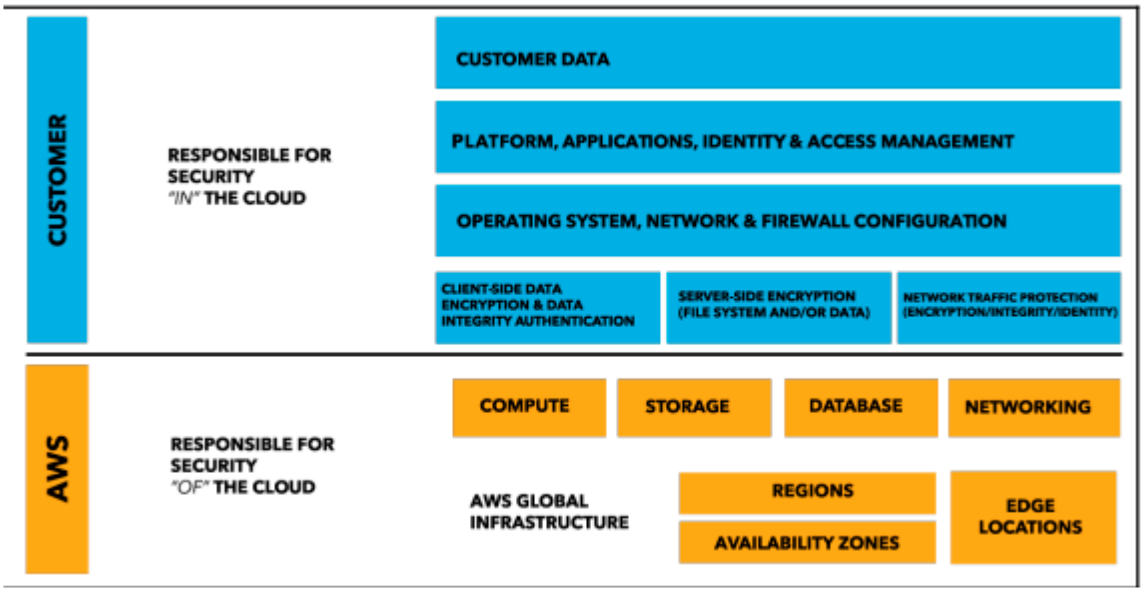


Рис 1.5. Концепція моделі спільної відповідальності за безпеку AWS

Щоб впевнено керувати безпекою на AWS, важливо розібратися в тому, яку частину відповідальності за безпеку несе AWS і яка частина лягає на користувачів. AWS презентує широкий спектр послуг, які можна класифікувати у три основні групи: інфраструктурні, контейнеризовані та високорівневі послуги. Кожна з цих груп має свою специфіку у власності на безпеку, в залежності від того, як користувачі взаємодіють з послугами та як вони використовують їх можливості:

● **Інфраструктурні послуги.** До цієї групи належать обчислювальні ресурси як Amazon EC2 та пов'язані з ними сервіси, такі як Amazon EBS, Elastic Load Balancing і Amazon VPC. Ці послуги дозволяють створювати захищені віртуальні мережі в хмарі, схожі на ті, що існують у фізичних дата-центрах, і можуть інтегруватися з локальними мережами. Забезпечується управління операційною системою, налаштування мережевих правил і систем ідентифікації користувачів.

● **Контейнерні послуги.** AWS пропонує рішення на базі контейнерів, які забезпечують кероване середовище для застосунків. За безпеку таких сервісів, як AWS Elastic Beanstalk, EMR та Amazon RDS, відповідають шляхом налаштування мережевих правил, що дозволяють користувачам та системам отримувати доступ до цих сервісів через IAM.

● **Високорівневі послуги.** Ці сервіси, такі як Amazon S3 чи DynamoDB, надають платформу, що приховує технічні деталі управління платформою чи системою. Вони доступні через API і включають послуги обміну повідомленнями, електронну пошту, бази даних NoSQL та сховища. AWS керує підсистемами, що лежать в основі цих сервісів, ізолюючи ваші дані від інших користувачів, тоді як користувачі користуються забезпеченими можливостями безпечного доступу через IAM.

AWS надає пари ключів для Amazon EC2, що включають публічний та приватний ключі, і використовуються для автентифікації та забезпечення доступу до ваших екземплярів EC2. При запуску нового екземпляра EC2 перед користувачами стоятиме вибір створення нової пари ключів або використання наявної, причому ігнорування ключів не рекомендується для забезпечення безпеки. Можна створити до 5000 таких пар для екземплярів EC2 в рамках AWS облікового запису, але вони призначені лише для доступу до екземплярів EC2 і не використовуються для авторизації в інших сервісах AWS чи консолі управління. Різні пари ключів можуть використовуватися для доступу до різних екземплярів EC2.

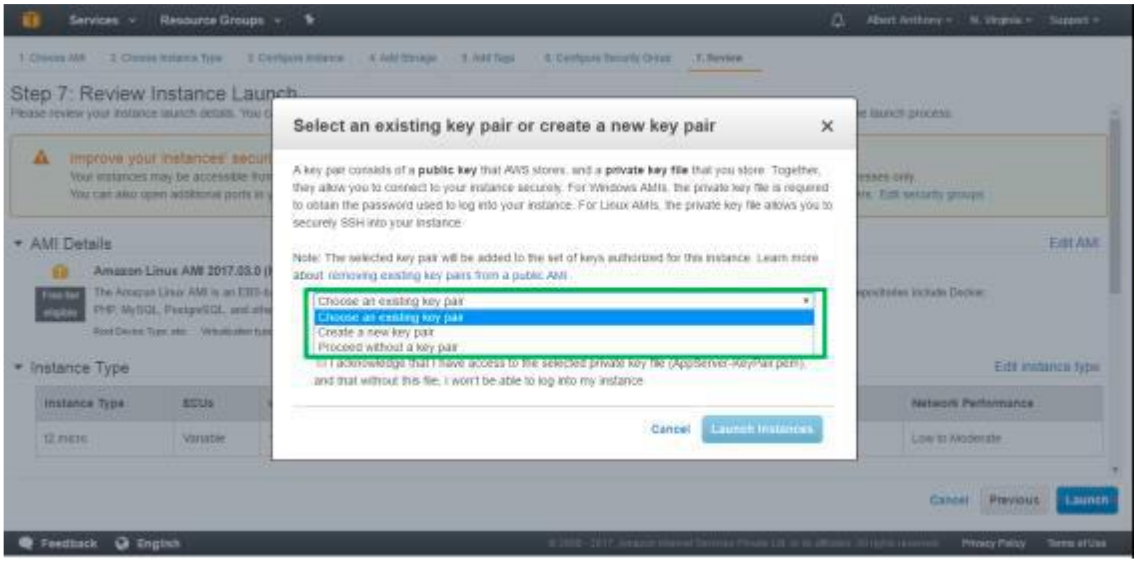


Рис.1.6. Приклад підходу щодо додавання пари ключів при запуску сервера

Можна надати Amazon Web Services завдання генерувати пари ключів EC2, або створити власні пари ключів EC2 за допомогою стандартних засобів, таких як OpenSSL.

У першому випадку, при запуску екземпляра, AWS забезпечує як відкритий, так і приватний ключі RSA. Приватний ключ необхідно зберігати у безпеці, оскільки втрата ключа означатиме неможливість його відновлення через AWS, та потребуватиме створення нової пари ключів.

При розгортанні нового екземпляра Linux EC2 із стандартного образу AWS AMI, відкритий ключ із збереженої в AWS пари ключів EC2 додається до операційної системи. Для з'єднання можна використати клієнт SSH, налаштований на використання імені користувача EC2, наприклад ec2-user, та приватного ключа для аутентифікації.

При запуску нового екземпляра Windows EC2 за допомогою служби ec2config зі стандартного образу AWS AMI, ec2config генерує новий випадковий пароль адміністратора для цього екземпляра Windows і шифрує його за допомогою відкритого ключа відповідної пари ключів EC2. Шифрований пароль розшифровується за допомогою приватного ключа, який використовується для аутентифікації на Windows екземплярі.

AWS надає зручні інструменти для управління ключами EC2 та доступом до екземплярів EC2, однак у випадках, коли потрібен вищий рівень безпеки через бізнес-вимоги або дотримання регулятивних стандартів, можна розглянути впровадження альтернативних механізмів аутентифікації, наприклад LDAP, та відмовитися від аутентифікації за допомогою ключів EC2[6].

Висновки до розділу 1

Проаналізовано хмарні технології та підкреслено, що хмарна інфрастурктура представляє собою нову технологічну парадигму, спрямовану на спільне використання обчислювальних ресурсів з метою підвищення ефективності та зниження витрат на адміністрування та інші IT-витрати.

Представлено модель типів розгортання хмарних інфраструктур, що може бути реалізована як: приватна хмара, хмара спільноти, публічна хмара та гібридна хмара. Виокремлено модель порушника хмарної інфраструктури.

Зазначено, що існує дев'ять основних загроз безпеці хмарних інфраструктур: порушення даних, втрата даних, викрадення облікових записів, незахищені API, відмова в обслуговуванні, зловмисні інсайдери, зловживання хмарними службами, недостатній рівень обережності, спільні технологічні проблеми.

Описано і додатково виявлені загрози безпеці, що включають: неправильну реалізацію безпеки гіпервізора в неприватних хмарних середовищах IaaS, неправильне використання брандмауера та неправильне впровадження брандмауера та контролю доступу, переважне використання вразливих розподілених систем баз даних (DDS) та шкідливі програми.

Проведено аналіз поширених загроз, спрямованих на хмарні технології та інфраструктуру:

- загрози IaaS включають зловживання хмарними сервісами та відсутність досвіду в галузі хмарних технологій;
- загрози PaaS включають неможливість гарантувати повне видалення даних та порушення цілісності повідомлень;
- до загроз SaaS окремо відносять неадекватний моніторинг;
- до поширених загроз відносять проблеми автентифікації.

Зазначено, що Amazon Web Service (AWS) вважається одним з перших гігантів хмарних інфраструктур, та пропонує різноманітні сервіси та інструменти, такі як ідентифікація та доступ, шифрування, ведення журналів, нагляд та дотримання норм, для забезпечення безпеки в хмарному середовищі. Ці сервіси AWS дозволяють виконувати широкий спектр завдань для задоволення всіх вимог безпеки, реєстрації користувачів, аудиту та відповідності в хмарному середовищі.

2 ДОСЛІДЖЕННЯ ВРАЗЛИВОСТЕЙ ТА ВЕБ-ЗАГРОЗ ХМАРНІЙ ІНФРАСТРУКТУРІ AMAZON AWS

2.1 Дослідження інфраструктурних ресурсів Amazon Web Services (AWS)

Інфраструктурні ресурси Amazon Web Services (AWS) - це основа для будь-якої інфраструктури або додатку, розгортаного в хмарному середовищі AWS. AWS пропонує різноманітні інфраструктурні ресурси, які можна налаштовувати та масштабувати згідно потреб користувача. Серед цілої низки різних варіантів можна виокремити наступні інфраструктурні ресурси AWS:

Amazon Elastic Compute Cloud (Amazon EC2). Amazon EC2 надає віртуальні машини, які можна налаштовувати за розміром, обсягом пам'яті, типом процесора та операційною системою. Це ідеальний ресурс для розгортання власних серверів або обчислювальних потоків в хмарному середовищі.

Amazon Simple Storage Service (Amazon S3). Amazon S3 - це послуга зберігання об'єктів, призначена для зберігання та управління даними, такими як файли, зображення та відео. Дані зберігаються у вигляді «ведер» (букашок), і доступ до них може бути налаштований через політики доступу.

Amazon Relational Database Service (Amazon RDS). Amazon RDS надає керовані послуги реляційних баз даних, включаючи PostgreSQL, MySQL, Oracle та інші. Ця послуга спрощує управління базами даних, забезпечуючи автоматизовану резервне копіювання, масштабування та планування.

Amazon Virtual Private Cloud (Amazon VPC). Amazon VPC дозволяє користувачам створювати власні ізольовані мережі у хмарному середовищі.

Таким чином, використовуючи зазначені складові, можна здійснювати налаштування мережі, підмережі, маршрутів та правил безпеки для контролю мережевого доступу до ресурсів.

Додатково виокремлюються послуги, що спрощують вивчення та дослідження доступних послуг AWS, а саме:

Amazon Elastic Load Balancing (Amazon ELB). Amazon ELB надає послуги балансування навантаження для розподілення трафіку між кількома Amazon EC2 екземплярами. Це підвищує доступність та надійність додатків, розгортаних у хмарі.

Amazon Route 53. Amazon Route 53 - це послуга доменних імен та системи DNS, яка дозволяє реєструвати та управляти доменними іменами для ваших додатків. Вона також надає можливість для глобального розподілення трафіку та обслуговування доменів.

Amazon Simple Queue Service (Amazon SQS). Amazon SQS надає послугу управління чергами для обміну повідомленнями між компонентами додатку. Це допомагає розробникам створювати розділені та масштабовані додатки.

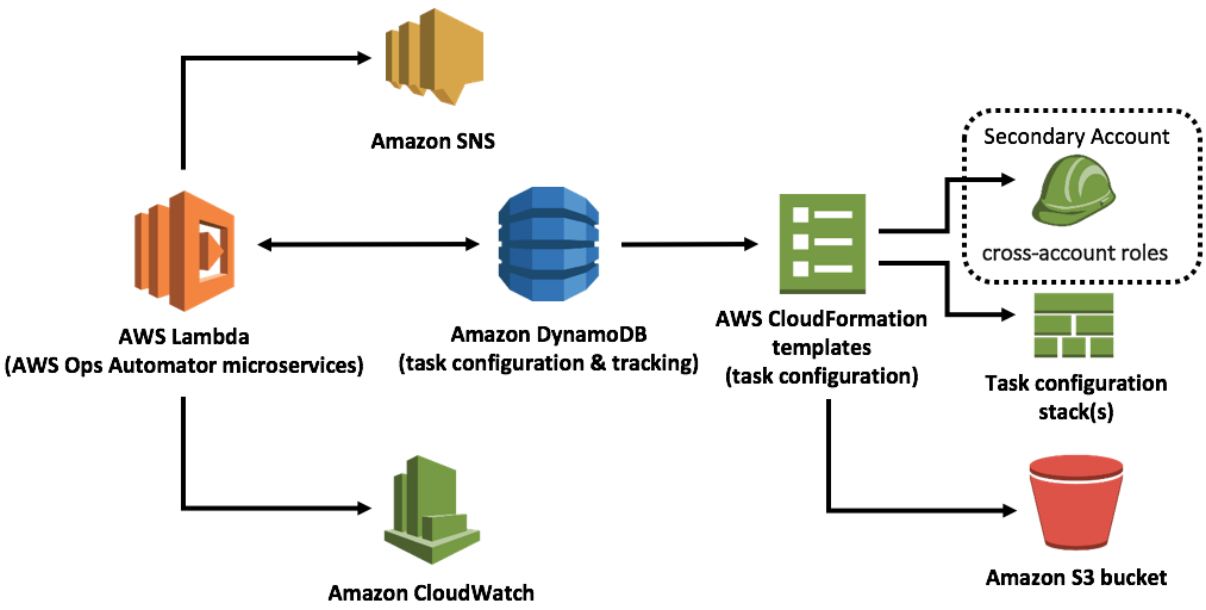


Рис.2.1. Інфраструктурні ресурси Amazon Web Services (AWS)

Amazon Elastic Block Store (Amazon EBS). Amazon EBS надає блочне сховище для зберігання даних, яке може бути прикріплене до Amazon EC2 екземплярів. Це дозволяє створювати постійне зберігання для вашого обчислювального ресурсу.

Amazon Identity and Access Management (IAM). Amazon IAM дозволяє керувати ідентифікацією та управлінням доступом до ресурсів AWS. Ви можете

налаштовувати ролі, політики та правила для забезпечення безпеки ваших інфраструктурних ресурсів.

Зазначено лише кілька прикладів інфраструктурних ресурсів, які надає AWS. Ці ресурси дозволяють користувачам розгортати та управляти інфраструктурою у хмарному середовищі з високою гнучкістю та масштабованістю. Вони є фундаментом для будь-якого проекту або додатку, який працює на платформі AWS[7].

2.2. Виокремлення критичних аспектів забезпечення безпеки ресурсів Amazon Web Services (AWS)

Безпека в Amazon Web Services (AWS) - це критичний аспект для користувачів, оскільки AWS забезпечує низку інструментів і послуг для забезпечення безпеки даних, додатків та інфраструктури в хмарному середовищі. Огляд безпеки AWS може бути наступним:

Ідентифікація та управління доступом (IAM). AWS Identity and Access Management дозволяє керувати доступом до ресурсів і послуг AWS. Користувачі можуть налаштовувати ролі, політики та правила для точного керування доступом.

Віртуальні закладки та сегменти Amazon VPC. Amazon VPC дозволяє створювати ізольовані мережі з власними політиками безпеки. Користувачі можуть налаштовувати правила мережевої безпеки для контролю мережевого доступу.

AWS Identity Services. AWS Cognito надає послуги ідентифікації та автентифікації для додатків. Amazon Inspector виявляє потенційні вразливості у додатках і системах.

Захист даних. Amazon S3 дозволяє налаштовувати доступ до об'єктів та застосовувати шифрування для даних в спокої. AWS Key Management Service надає кероване шифрування для зберігання ключів.

Моніторинг та аудит. AWS CloudTrail записує всі події у обліковому запису для аудиту і моніторингу. Amazon GuardDuty надає миттєвий аналіз на предмет потенційних загроз.

Безпека мережі. AWS WAF забезпечує захист від веб-атак і відмов у обслуговуванні (DDoS). AWS Firewall Manager дозволяє керувати політиками брандмауера для захисту всієї інфраструктури.

Автоматизація та налаштування безпеки. AWS Config дозволяє контролювати конфігурацію ресурсів та виявляти зміни. AWS Systems Manager допомагає автоматизувати планування та налаштування для забезпечення безпеки.

Фізична безпека та забезпечення доступу до даних. AWS забезпечує фізичну безпеку своїх центрів даних та обмежує доступ до них.

Адаптовані безпечні образи Amazon Machine Image (AMI). AWS Marketplace надає перевірені образи AMI для застосунків з підвищеною безпекою. AWS надає користувачам інструменти та ресурси для реалізації високого рівня безпеки в хмарному середовищі, що робить його популярним вибором для багатьох компаній та розробників.

Ретельне налаштування та дотримання найкращих практик безпеки дозволяють забезпечити безпечне функціонування ваших додатків і даних в AWS.

Необхідно підкреслити, що веб-загрози у хмарному середовищі становлять серйозний ризик для безпеки даних, додатків та інфраструктури, розгортаної в хмарних платформах, таких як Amazon Web Services (AWS), Microsoft Azure та Google Cloud Platform (GCP). До ключових можна віднести:

DDoS атаки (атаки на змовлене відмовлення обслуговування). Хмарні інфраструктури легко доступні для атак типу DDoS, де зловмисники спрямовують велику кількість трафіку на ресурс, щоб перевантажити його і заблокувати доступ для легітимних користувачів.

Злам аккаунту (Account Compromise). Зловмисники можуть намагатися зламати акаунти користувачів або адміністраторів для незаконного доступу до хмарних ресурсів.

SQL Injection та інші атаки на додатки. Зловмисники можуть намагатися використовувати вразливості додатків, такі як SQL Injection, для отримання доступу до баз даних та конфіденційної інформації.

Cross-Site Scripting (XSS). Атаки XSS можуть допомогти зловмисникам впроваджувати зловісний JavaScript у веб-сторінки, що використовуються користувачами, для крадіжки сесій або інших даних.

Зловмисні завантаження файлів. Зловмисники можуть намагатися завантажити зловісні файли або програми в хмарні системи для виконання атак або крадіжки даних.

Фішинг-атаки. Зловмисники можуть намагатися вести фішингові атаки на користувачів хмарних платформ, щоб отримати їхні облікові дані або іншу конфіденційну інформацію.

Недоліки налаштування. Неправильна конфігурація хмарних ресурсів може призвести до витоку конфіденційних даних або неправильного доступу до ресурсів.

Зловмисники, що використовують служби хмарних постачальників. Зловмисники можуть намагатися використовувати послуги хмарних постачальників для атак на інші користувачі або послуги[8].

2.3 Дослідження особливостей реалізації Amazon S3

Хмарні постачальники мають широкий спектр послуг, які пропонують можливості зберігання файлів. Сховище Amazon S3 є однією з найбільш впізнаваних послуг у галузі, якою користується значна кількість клієнтів, від маленьких користувачів до великих корпорацій.

З моменту запуску в 2006 році сервіс зберігання об’єктів став одним з основ Amazon Web Services (AWS). Однак широке розповсюдження Amazon S3 стало причиною поширення загроз з боку зловмисників, які прагнуть скористатися будь-якими недоліками в Amazon для створення передумов витоку конфіденційних даних.

Неправильну конфігурацію Amazon S3 часто використовують для зловмисних цілей, таких як криптовикрадення, електронний скімінг і викрадання даних. Згідно з звітом Інституту SANS щодо хмарної безпеки в 2021 році, до

найпоширеніших загроз та атак, якими скористалися, відносять: викрадення облікового запису (облікових даних), некоректні конфігурації та зловживання привілейованими користувачами.

Криптоджекінг через доступ до записів «відра» Amazon S3. У лютому 2018 року було виявлено майнера криптовалюти Monero в JavaScript субдомені веб-сайту США. Газета мала веб-сайт, розміщений на AWS, який статично розміщувався в сегменті Amazon S3. Весь веб-сайт містився в сегменті, включаючи всі зображення, файли каскадних таблиць стилів (CSS) і файли JavaScript, які зберігалися як об'єкти в цьому сегменті.

Відро Amazon S3 – це логічний контейнер, в якому можна зберігати файли та дані. Кожне відро має унікальне ім'я та може містити різноманітні об'єкти, такі як файли, зображення, документи тощо.

Інтерфейс HTTP(S) забезпечував доступ лише до читання вмісту сегмента Amazon S3. Однак «відро» було неправильно налаштовано. Тому зловмисник отримав доступ до нього через рідний протокол Amazon S3, та зміг змінити дозволи. В наступні роки проблему було пом'якшено завдяки створенню «відра», яке було важче неправильно налаштувати.

Неправильна конфігурація ACL дозволила записувати доступ до всього «відра» з будь-якого облікового запису. Це, в свою чергу, дозволило зловмисникові додати свій майнер криптовалюти Monero до коду JavaScript, який виконувався кожного разу, коли відвідувач відкривав веб-сайт (рис.2.2).

Така помилка розповсюджена серед користувачів-початківців, які використовують Amazon S3 для розміщення веб-вмісту. Під час створення веб-сайту, розміщеного на Amazon S3, не завжди очевидно для користувачів, що весь вміст стає доступним не лише через HTTP(S), але і через сам API AWS, а також ім'я сегмента може бути отримано з імені домену HTTP(S).

Якщо розробник помилково встановить «відро» як вільний для запису, неавторизовані користувачі можуть змінювати його. Надалі, AWS впровадили основну політику контролю доступу, яка забороняє створення «відра» Amazon S3 доступного для загального читання або запису через консоль. Навіть якщо відро

було створено через командний рядок, публічний доступ можна все ще заблокувати через інтерфейс командного рядка AWS (CLI).

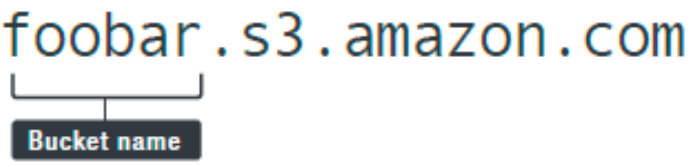


Рис.2.2. Частина назви сегмента в URL-адресі віртуального хостингу AWS

У випадку із зазначеною вразливістю, зловмисник проаналізував вміст сайту і зрозумів, що він розміщений на AWS. URL-адреса сайту віртуального стилю дозволила зловмиснику визначити назву «відра» Amazon S3. Зловмисник зміг легко взяти назву сегмента та перевірити, чи можна її змінювати. Оскільки це було можливо, зловмисник встановив майнер криптовалюти, який потім запускався в кожному браузері, що відкривав цю сторінку.

Викрадання даних через вільний для запису «відро» Amazon S3. Зловмисники часто використовують сегменти Amazon S3, доступні для запису для всього світу, для отримання прибутку. За допомогою телеметрії можна визначити, що з даних Akamai для інфраструктури Trend Micro Smart Protection Network було завантажено велику кількість інших веб-сайтів, які, принаймні частково, були розміщені на сегментах Amazon S3 та завантажені через віртуальні URL-адреси.

Інфраструктура Smart Protection Network надала додаткову інформацію про характер таких атак. Багато веб-сайтів-жертв, які були ідентифіковані за допомогою телеметричних даних Smart Protection Network, з червня по вересень 2019 року завантажували статичний вміст, такий як зображення або файли JavaScript, із сегментів Amazon S3. Наприклад, в квітні 2019 року було виявлено групу зловмисників, яким вдалося змінити деякий зміст японського вебсайта, розміщеного на AWS (рис.2.3).

Рис.2.3. Скріншот скомпрометованого японського веб-сайту

Проаналізувавши сайт, було зроблено висновок, як його було зламано. Сайт завантажив численні сценарії з «відра» Amazon S3, яке містило зображення та файли CSS (рис.2.4).

Рис.2.4. Завантажені сценарії з сегмента Amazon S3

Також завантажені файли JavaScript (рис.2.5)

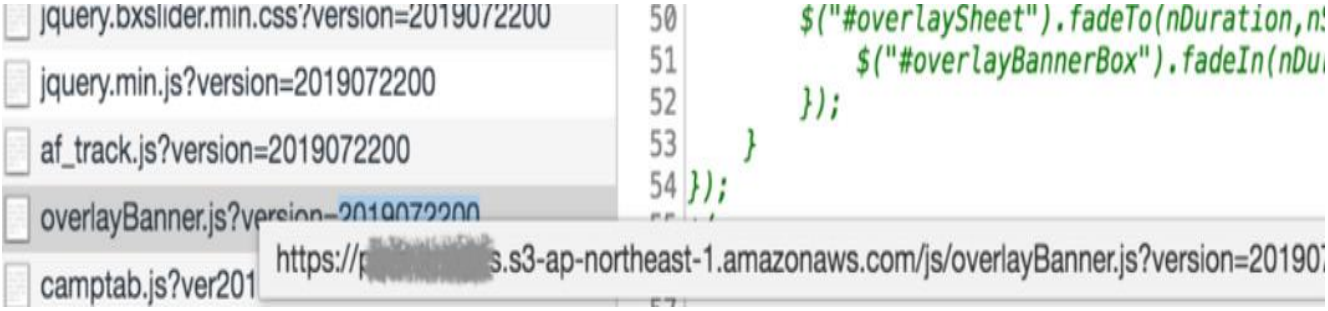


Рис.2.5. Завантажені файли JavaScript

Отримати назву сегмента Amazon S3 з URL-адреси JavaScript було дуже просто. Власники веб-сайту встановлюють дозволи сегмента Amazon S3, щоб вони були доступні для запису та читання, щоб проста команда на AWS CLI призвела до списку файлів, що зберігаються у «відрі» (рис.2.6).

```
aws s3 ls s3://###-######
                           PRE ORxYUSMSRf.jsp/
                           PRE WEzXTtAEBU.jsp/
                           PRE cs-csv/
                           PRE css/
                           PRE diZPqEAuJM.jsp/
                           PRE guide/
                           PRE hqmail/
                           PRE images/
                           PRE img/
                           PRE js/
                           PRE json/
                           PRE mail/
                           PRE material/
                           PRE point/
                           PRE promotion_mail/
                           PRE report/
                           PRE tmp/
                           PRE tomorrow_mail/
                           PRE xml/
2018-06-13 18:03:43        199
208b605c01bc1fd2b9ad92a96f77a169a84643cdeb82a9e64204e23f501afa17371012ec4c29
28fda5477f19eaecf9ff449e2accaef00c2d842bf9654e48a232.txt
2019-06-13 07:36:56       1742 404.html
2018-01-15 02:02:43        162 BugDisclosure.txt
2017-12-05 15:34:22        226 poc.txt
2018-09-12 19:17:58         91 rdttk78549.txt
2018-01-26 01:06:18         54 t.txt
2017-06-07 18:29:44         27 test.txt
2018-07-19 17:20:08        365 testupload.txt
```

Рис.2.6. Інформація, яку повертає команда AWS CLI, включаючи назву сегмента Amazon S3 із URL-адреси JavaScript, при запиті

Деякі зловмисники спеціально шукають відкриті «відра» (рис.2.7 та рис.2.8).

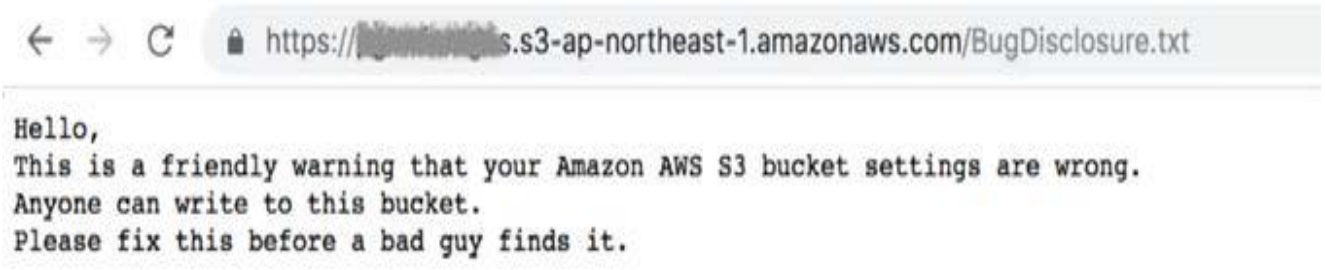


Рис.2.7. Анонімне попередження, яке сповіщає власника сегмента про неправильно налаштовані параметри

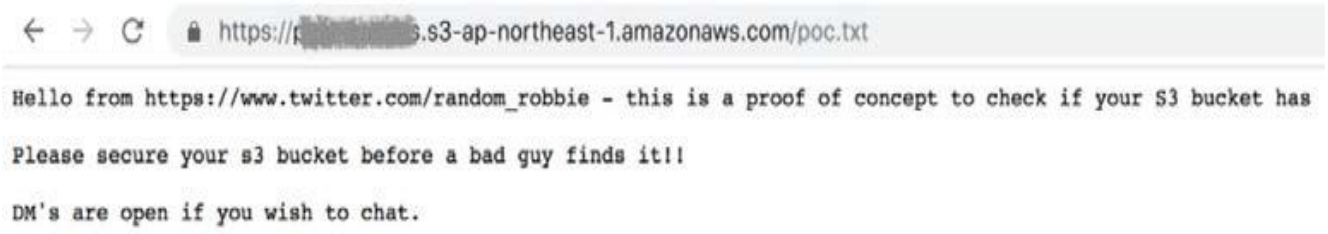


Рис.2.8. Повідомлення від «Випадкового Роббі», що пропонує потенційні консультаційні послуги власникові сегменту Amazon S3

«Випадковий Роббі» - псевдонім Роббі Віггінса, військового тестувальника на проникнення, який розробив скрипт для виявлення сегментів Amazon S3 з можливістю запису та завантаження файлу poc.txt до сегмента, імовірно, в надії, що власник помітить це і виправить проблему з дозволами. За словами Віггінса, він позначив тисячі сегментів Amazon S3 таким чином у 2018 році.

Файли JavaScript, розміщені на сегментах Amazon S3 з можливістю запису, можуть бути змінені для витягнення даних відвідувачів веб-сайту. Дані із інфраструктури Smart Protection Network свідчать про наявність серії атак на вилучення даних протягом більшої частини 2019-2021 років, в результаті чого було скомпрометовано низку веб-сайтів, включаючи сайти роздрібних торгових компаній, де були вкрадені дані їхніх клієнтів[9].

Як висновок можна зазначити, що видалення JavaScript-коду, який викрадає дані, не є вичерпною мірою. Поки сегмент Amazon S3 залишається доступним для запису, зловмисник може легко відновити його, просто вставивши код в існуючі

файли JavaScript на веб-сайтах. Тому конфігурацію слід перевіряти щоразу, коли з'являються загрози.

Оскільки зловмисники залишаються надзвичайно активними та продуктивними, важливо, щоб організації та власники сегментів Amazon S3 приймали активні заходи для запобігання таким компромісам. Власники облікових записів AWS можуть визначити доступність сегмента для запису, використовуючи Amazon S3 API.

AWS також надає інструмент командного рядка для спрощення цього процесу. Команда *aws s3api get-bucket-acl --bucket <BUCKET_NAME>* використовується для отримання інформації щодо списків керування доступом (ACL) для сегментів, включаючи інформацію про власника та дозволи.

Поле «власника» містить ідентифікатор власника сегмента Amazon S3. Поле «грантів» містить список дозволів, наданих групам або конкретним користувачам AWS. Наприклад, для японського веб-сайту команда повертає наступну інформацію (рис.2.9).

```
"Grantee": {
   "Type": "Group",
   "URI": "http://acs.amazonaws.com/groups/global/AuthenticatedUsers"
},
"Permission": "FULL_CONTROL"
```

Рис.2.9. Повернення інформації після запиту до японського веб-сайту

Веб-сайт, розміщений на Amazon S3, має бути загальнодоступним і зазвичай кешується за допомогою Amazon CloudFront. Але «відро», яке не використовується для розміщення, ймовірно, містить файли компанії, які навряд чи мають бути загальнодоступними.

2.4. Аналіз особливостей протидії криптоджекінгу та електронному скімінгу в сегментах Amazon S3

Невірна конфігурація, яка дозволяє повний контроль за сегментом будь-кому, хто створює організаційні ресурси, робить їх вразливими перед такими атаками, як криптоджекінг і електронний скімінг.

Існують також інструменти, які аналізують перерахування читабельних «відер» і шукають цікаві файли. Наприклад, AWSBucketDump використовує ключові слова для пошуку файлів у «відрах» (рис.2.10).

URL
http://s3.amazonaws.com:80/#######/55mark/tbsetup_0218.exe
http://s3.amazonaws.com:80/#######/releases/DownloadManagerV2.exe
http://s3.amazonaws.com:80/#######/55mark/tbsetup_0122_3.exe
http://s3.amazonaws.com:80/############/ServantKeeper7Update7.0.50.exe
http://s3.amazonaws.com:80/###/downloads/winmaximizer/WinMaximizer_Setup_2015.exe
http://s3.amazonaws.com:80/###-#######/WL2KGopMRgSoTcy6PfD6_agressive2.exe
·://s3.amazonaws.com:80/#######/releases/DownloadManagerV2.exe
http://s3.amazonaws.com:80/###-#######/gP84JywRSKqWHKgdvMdj_MineCheat%20v1.0.3.exe
http://s3.amazonaws.com:80/###-#######/JY8IsFPqQ5uvokZzRjNe_%D0%A1%D0%B0%D0%BC%D1%8B%D0%B9%20%D0%BB%D1%83%D1%87%D1%88%D0%B8%D0%B9%20%D1%87%D0%B8%D1%82!.exe

Рис.2.10. Приклад переліку доступних для читання та виконання сегментів та файлів Amazon S3

Загальнодоступні сегменти Amazon S3 можуть використовуватися кіберзлочинцями для інших зловмисних цілей: використання командно-контрольного (C&C) сервера, як точки входу для викрадання конфіденційних даних або для зберігання дочірніх даних експлуатаційного матеріалу з метою уникнення відстеження.

Також можуть бути скомпрометовані сегменти Amazon S3, що може призвести до розгортання коду та моніторингу журналу.

Серед інших негативних проблем можна виокремити застарілі шляхи. Розміщені у сегментах Amazon S3, ім'я хоста є загальним ім'ям хоста Amazon S3, зазвичай s3.amazonaws.com, але воно також може містити регіон. На відміну від цього, більш поширений стиль віртуального розміщення включає ім'я сегмента, наприклад, <назва-відра>.s3.amazonaws.com.

Якщо кіберзлочинці використовують схему стилю шляху, то сайти не можуть бути заблоковані лише за іменем хоста, і це також може призвести до блокування багатьох легальних сайтів. В разі шкідливих сайтів, які використовують схему віртуального розміщення, можливе налаштування фільтру або внутрішньої системи доменних імен (DNS) для блокування певних хостів, визначених як шкідливі.

Наразі AWS застосовує застарілу схему стилю шляху. Для розв'язання цих проблем і надання можливості користувачам обійти цензуру, AWS дозволяє продовжити використовувати сегменти з іменуванням у стилі шляху, які були зареєстровані до 30 вересня 2020 року.

На даний момент AWS впроваджує політику блокування публічного доступу до Amazon S3, яка, коли ввімкнена, блокує всі нові сегменти, створені через консоль управління AWS, не дозволяючи публічний доступ без відповідного налаштування.

Відкриті сегменти Amazon S3 - це лише одна з проблем захисту хмарних ресурсів та систем. Наприклад, в разі витоку даних американського банківського холдингу використовувалися інші методи для скомпрометування даних, зберіганих у сегментах Amazon S3. Доки сегменти Amazon S3 не були опубліковані у читабельному вигляді, зловмиснику вдалося взяти на себе роль (XXX-WAF-Role) та викрасти дані. Позов чітко не пояснює, як зловмиснику вдалося взяти на себе роль, але вказує на те, що «невірна конфігурація брандмауера дозволила командам досягати та виконувати певний сервер», включаючи отримання ролі XXX-WAF, яка мала доступ до папок та сегментів Amazon S3[10].

2.5 Дослідження інструментів впровадження контейнерів в хмарній інфраструктурі

Ще однією важливою категорією послуг, які пропонують постачальники комплексних хмарних послуг, є обчислення. Спочатку пропозиції в цій категорії

були в основному віртуалізованими серверами, такими як Amazon Elastic Compute Cloud (Amazon EC2).

З часом вони розширилися, включаючи більш легкі пропозиції на основі контейнеризації. Спочатку необхідно розглянути Docker, який є популярною контейнерною платформою.

В AWS контейнерні служби запускаються за допомогою Amazon Elastic Container Service (Amazon ECS), і Amazon ECS Clusters можуть бути запущені за допомогою AWS Fargate, реалізації Apache Mesos або Amazon Elastic Kubernetes Service (Amazon EKS), реалізації Kubernetes.

Починаючи з Amazon EKS і AWS Fargate, які використовують екземпляри Amazon EC2 (віртуалізований серверний сервіс), багато фахівців з області хмарних технологій та DevOps обирають розміщувати свої Docker-служби безпосередньо на власних серверах Amazon EC2.

Docker. Docker - це система управління, розроблена на основі CGROUPS та просторових імен Linux, і є формою легкої віртуалізації. У звичайному розгортанні Docker, ядром Linux контролюються такі ресурси, як пам'ять, дисковий простір та обчислювальна потужність CPU. Зазвичай не повинно виникати неконтрольованих взаємозв'язків між контейнерами Docker, де запущені процеси.

Це схоже на віртуалізацію на основі гіпервізора, але менш важке та безпечне. Цей підхід став популярним в практиці DevOps, оскільки дозволяє розробникам пакувати програму з усіма залежностями, включаючи операційну систему, у вигляді образу, і запускати її без турботи про можливу несумісність з базовою системою.

Образи Docker створюються з усіма встановленими вимогами і можуть бути легко розгорнуті за допомогою API сервера Docker або командного рядка, який використовує цей API. Керування виконанням контейнера може здійснюватися через Docker API. Використання кількох контейнерів дозволяє ефективніше використовувати ресурси сервера завдяки спільному використанню.

Зазвичай розробники встановлюють сервер Docker на віртуалізованому сервері у хмарному середовищі. Це надає їм можливість спочатку розробляти та

налаштовувати програму локально і потім розгортати її в хмарі без додаткових зусиль. Важливо зауважити, що сервер Docker не повинен бути відкритий для мережі Інтернет без відповідного захисту.

Однак часто в даних виявляються вразливості, що призводить до необхідності в забезпеченні додаткової безпеки, зокрема у випадках сканування Shodan.

Якщо обмежили пошук наступним чином: аналізувати сервери, які Shodan вважає постачальниками хмарних послуг, та сервери, що мають відкритий порт 2375 або містять рядок «docker» (за допомогою рядка пошуку «tag:cloud порт:2375 docker»), будуть зібрані близько 20 000 записів протягом місяця, із них близько 4 000 з унікальними IP-адресами. Однак лише 45 серверів доступні для перевірки. Таким чином, хоча виявлено лише декілька вразливих серверів, імовірно, їх було набагато більше.

Тому рекомендується організаціям ще раз переглянути налаштування брандмауерів та віртуальних приватних мереж (VPN), щоб уникнути випадкового розкриття даних та систем. Важливо відзначити, що кількість серверів Docker, які можуть використовувати для сканування, може значно зростати.

Сервери Docker можуть стати проксі-серверами для численних незахищених серверів у мережі. З використанням інструменту командного рядка Docker можна отримувати додаткову інформацію про контейнери та образи, наприклад, за допомогою команди *docker -H <docker-host-fqdn>* ps можна отримати список усіх запущених контейнерів разом із пов'язаними з ними метаданими. Однак намагатися отримати доступ до серверів без дозволу є незаконним.

Типове розгортання Docker в хмарному середовищі базується на обчислювальних екземплярах з автоматичним масштабуванням. Коли використання процесора перевищує встановлений поріг, створюються нові екземпляри обчислень за встановленим шаблоном (рис.2.11). Тому витрати стають відчутними (наприклад, Amazon EC2).

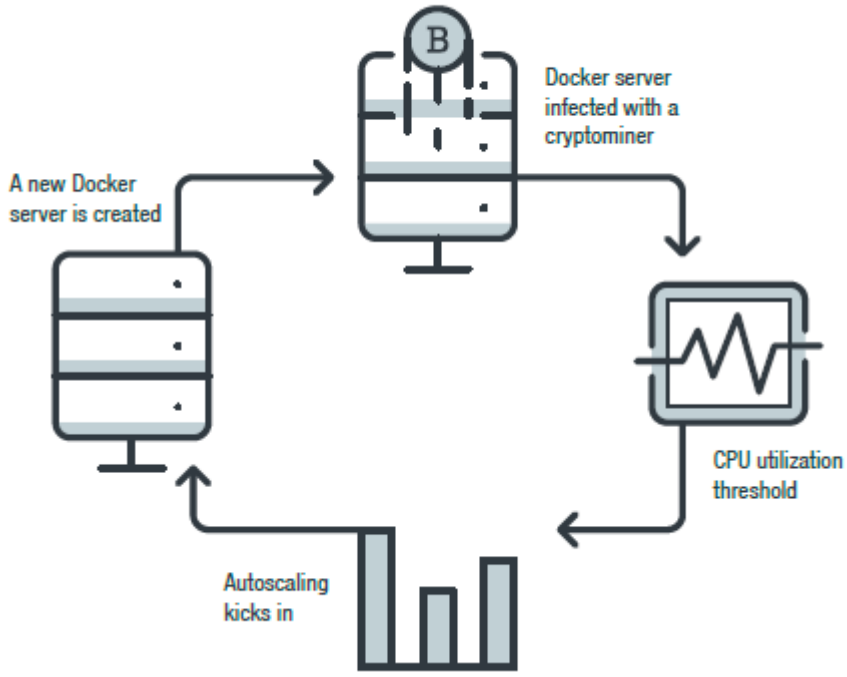


Рис.2.11. Схема створення нового сервера Docker при перевищенні позначки порога використання ЦП

Важливо зрозуміти, що подібно до ситуації, коли облікові дані AWS можуть бути скомпрометовані і використані недобросовісною особою, будь-який контейнер в системі Docker з відкритим кодом може бути перевірений, і його дані можуть бути витягнуті. Крім того, ці контейнери можуть дозволяти бічним атакам на інші ресурси компанії. Це означає, що зловмисник може вже використовувати всі можливі вектори атаки через ці контейнери Docker і вкрасти цінні дані, перш ніж встановити майнер криптовалюти для збільшення свого прибутку. Отже, на перший погляд те, що може здатися майнером криптовалюти без подальшого пояснення, може свідчити про існування інших потенційних загроз безпеці в системі.

Існує багато способів захисту серверів Docker. Docker надає документацію з питань безпеки, а також експерти розробили найкращі практики. Також рекомендується використовувати Transport Layer Security (TLS) для забезпечення безпеки зв'язку між клієнтом і сервером, а також використовувати сертифікати на стороні клієнта для контролю доступу в середовищі DevOps. Важливо також мати

на увазі, що CGROUPS Linux не призначені для повної ізоляції процесів, тому будь-який контейнер, який запускається, може мати можливості порушити інші контейнери або навіть сам хост, попри відсутність прямого доступу до сервера Docker.

AWS Lambda. AWS Lambda - це легкі процеси, які виконуються протягом обмеженого часу, зазвичай у відповідь на певні події, такі як доступ до веб-шлюзу API або доступні нові дані в сховищі Amazon S3. Lambda є частиною безсерверної архітектури, яка виявляється особливо економічно вигідною для програм з непередбачуваними моделями використання, такими як Інтернет речей (IoT). Вона також знаходить застосування в потоці обробки даних.

Розробники часто припускають, що оскільки імена лямбда-функцій не відомі зловмиснику, це гарантує певний рівень захисту. Тому під час розробки часто створюються лямбда-функції без належної аутентифікації. Параметри до цих функцій часто передаються незахищеним способом або розкривають конфіденційну інформацію.

Зловмисники можуть легко виявляти лямбда-функції, аналізуючи журнали проксі-сервера, досліджуючи вихідний код веб-сторінок, які використовують лямбда-функції з веб-шлюзом API на задній частині, або акуратно прослуховуючи мережевий трафік з мережевим аналізатором.

Під час дослідження були виявлені лямбда-функції, які не мали належної аутентифікації і могли потенційно розкривати конфіденційну інформацію, таку як файли конфігурації або навіть внутрішня структура додатків під час обробки запитів. Наприклад, лямбда-функція за таким посиланням: https://##########.execute-api.us-west-2.amazonaws.com/dev/coach/replicated-site?coachId=556794&locale=en_US потенційно може розкривати особисту інформацію при виклику[12].

```
[
  {
    "coachId": ######,
    "preferredLanguage": "en_US",
    "first_name": "S#####",
    "last_name": "R##s",
    "email": "s#####hr##s@gmail.com",
    "phoneCountry": "1",
    "phone": "",
    "bio": "Welcome to my Team B########® page. As your Coach, I would
love to work with you to help you achieve your health and fitness goals. If
you'd like to connect or have any questions, please reach out - I'm here
to help.",
    „profileImage": „https://images.coach.teamb########.com/original/
mysite/######/en_US/avatar/original.jpg",
    "aboutMeImage": "https://images.coach.teamb########.com/original/
mysite/######/en_US/about_me/original.jpg",
    "nextChallengeGroupDate": "2019-02-18",
    "twitter": "",
    "facebook": "https://www.facebook.com/#####s####r##s/",
    „instagram": „https://ww.instagram/#####s####r##s",
    "snapchat": "",
    "pinterest": "https://www.pinterest.com/thef###f####fe/",
    "youtube": "",
    "personal_website": "https://www.#####s####r##s.com",
    "promo1": "",
    "promo2": "",
    "promo3": "",
    "promo4": "",
    "promo5": "",
    "showShop": 1,
    "showFree": 1,
    "showCoach": 1,
    "showBod": 1
  }
]
```


Рис.2.12. Приклад вразливості лямбда-функції

Також не вважається хорошою практикою безпеки передавати ключ API в самій URL-адресі, навіть якщо використання. Протокол HTTPS повинен допомогти зменшити ризик (рис.2.13).


```
https://##########.execute-api.us-east-1.amazonaws.com/prod?apiKey=3_#####
###1########a########a##########v-########-G##########9####&domain=https%3
A%2F%2Fnd-frontend-prod.cfeu.#######.com

https://##########.execute-api.us-east-1.amazonaws.com/prod?apiKey=3_#####
###f#########c########n##########1-#####-A########g########L&domain=https%
3A%2F%2Ftranslate.googleusercontent.com
```


Рис.2.13. Приклад використання протоколу HTTPS

Розробники не повинні розглядати ефемерну природу лямбда-функції і незрозумілі URL-адреси як захист від атак. Безпека через невідомість проти рішучого зловмисника не ефективна. Той факт, що лямбда не працює довше максимум п’ятнадцяти хвилин, не означає, що цього достатньо для забезпечення безпеки.

Код, який виконується на тисячах лямбда-процесорів із використанням одного веб-шлюзу API, буде являтися ціллю зловмисників, котрі намагатимуться використовувати будь-яку знайдену вразливість, щоб отримати доступ, навіть якщо це триватиме недовго.

Kubernetes. Контейнерні служби, засновані на CGROUPS, зазвичай вимагають певного рівня керування для розгортання складних додатків, моніторингу активності контейнерів або забезпечення функціонування мереж для контейнерів. Однією з таких систем є Kubernetes(рис.2.14).

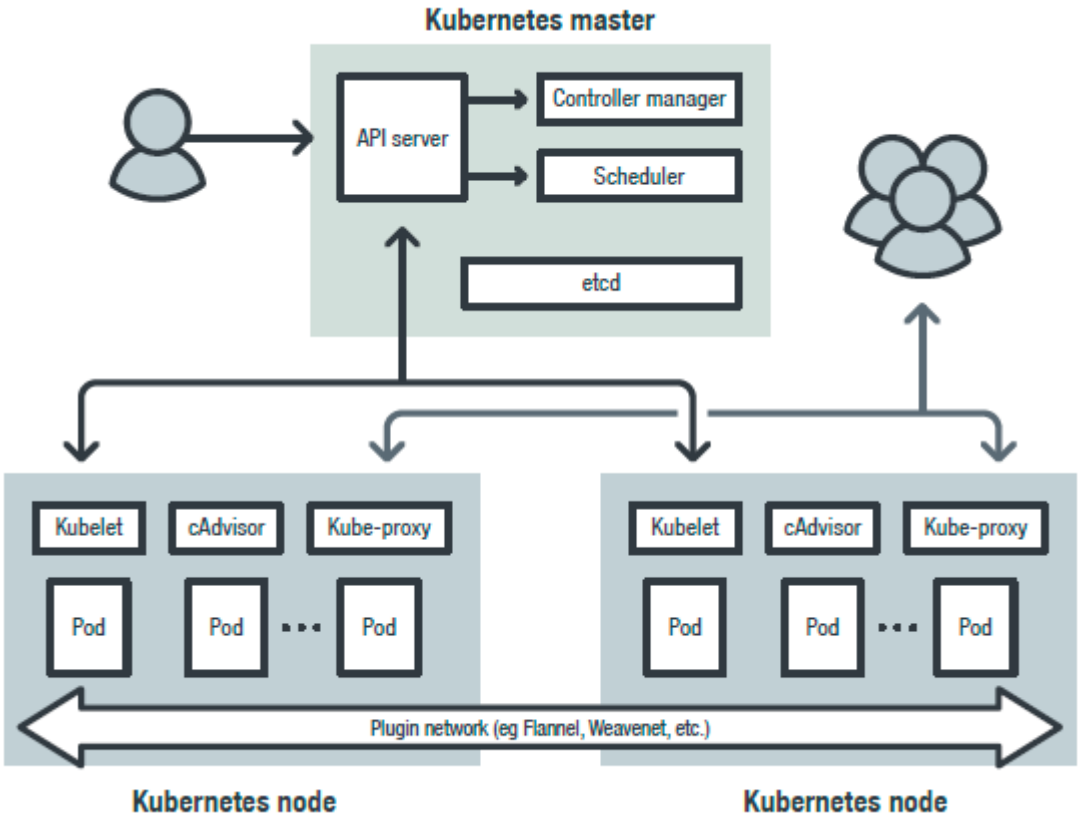


Рис.2.14. Діаграма Kubernetes і його компонентів Kubelet

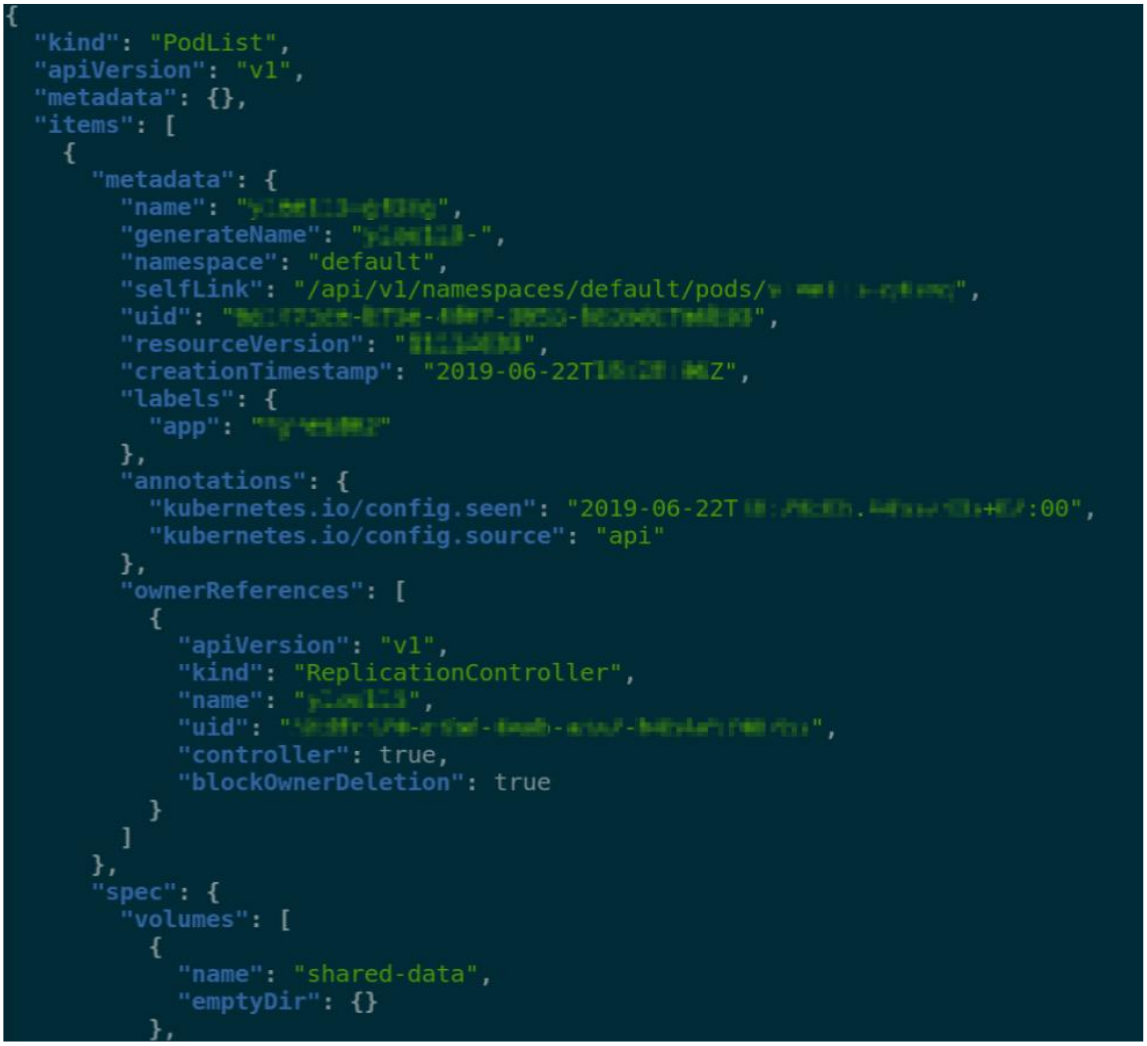
Підкомпонент Kubernetes, який відповідає за управління кожним вузлом, називається Kubelet. API Kubelet використовується Kubernetes для управління контейнерами на кожному вузлі. На жаль, у старих версіях Kubernetes API Kubelet не був захищений. Kubelet відкриває два порти: 10255 - http-порт лише для читання і 10250 - https-порт, за допомогою якого можна управляти сервером Kubelet. Лише в версії 1.10, була введена можливість налаштування автентифікації для контрольного порту 10250. Про те, скільки інсталяцій фактично використовують автентифікацію, невідомо[13].

Навіть якщо порт управління захищений, порт для читання також не повинен бути відкритим. Інформація, доступна через цей порт, надає важливу інформацію щодо того, що встановлено та працює на сервері (рис.2.15).

Рис.2.15. Код, що відображає інформацію, отриману через порт даних, призначений лише для читання

Після аналізу цих файлів можна виявити, що порт керування вже був використаний зловмисниками. Зафіксовані численні випадки інсталяції майнера Monero XMrig на серверах Kubelet. Незабаром стало відомо про наявність

програмного забезпечення для експлуатації, яке може виявляти та використовувати ці інсталяції - Kubelet-Exploit.



```
{
  "kind": "PodList",
  "apiVersion": "v1",
  "metadata": {},
  "items": [
    {
      "metadata": {
        "name": "
        "generateName": "
        "namespace": "default",
        "selfLink": "/api/v1/namespaces/default/pods/
        "uid": "
        "resourceVersion": "
        "creationTimestamp": "2019-06-22T
        "labels": {
          "app": "
        },
        "annotations": {
          "kubernetes.io/config.seen": "2019-06-22T
          "kubernetes.io/config.source": "api"
        },
        "ownerReferences": [
          {
            "apiVersion": "v1",
            "kind": "ReplicationController",
            "name": "
            "uid": "
            "controller": true,
            "blockOwnerDeletion": true
          }
        ]
      },
      "spec": {
        "volumes": [
          {
            "name": "shared-data",
            "emptyDir": {}
          },
```


Рис.2.16. Код, що показує неправильно використаний порт управління

2.6. Аналіз витоку облікових даних для запису AWS

Проблема з обробкою облікових даних є складною в будь-якому розгортанні, включаючи хмарні середовища. Для розробників важливо забезпечити безпеку доступу до ресурсів та даних, і це повинно бути автентифіковано та захищено від ризику витоку або компрометації даних. Для цього їм потрібно правильно впровадити необхідні облікові дані у процес таким чином, щоб не розкривати їх.

Спочатку розробники повинні бути обережні при обміні інформацією з іншими розробниками на форумах чи в інших місцях. Наприклад, було знайдено цікавий випадок на Pastebin (рис.2.17).

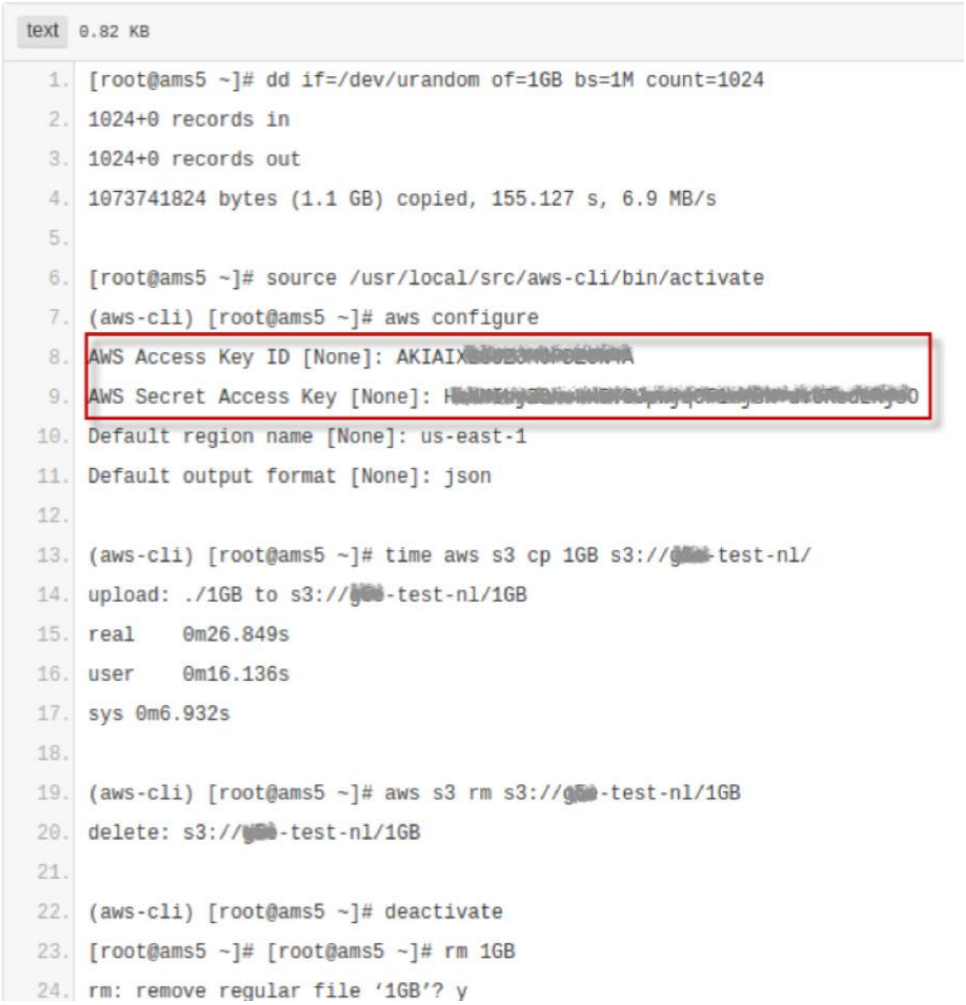


Рис.2.17. Фрагмент коду з Pastebin, що демонструє реальний API-ключ та ідентифікатор для облікового запису AWS

Цей фрагмент видається дійсним ідентифікатором API та ключем для облікового запису AWS, іншими словами, це готовий набір облікових даних для будь-кого, хто знайде його. З цією інформацією зловмисник може отримати доступ до усього облікового запису.

Справжній API-сервер Kubernetes, який використовується за замовчуванням, захищений сертифікатами на стороні клієнта. Однак це не завадило деяким розробникам використовувати ці сертифікати у вихідному коді, розміщеному на

GitHub, Pastebin та інших публічних службах. На Pastebin виявлено близько 50 екземплярів сертифікатів та скриптів конфігурації, які були загальнодоступні.

Сертифікати ніколи не повинні зберігатися у вигляді звичайного тексту в будь-якому місці, де їх можна легко видалити. Видалення сертифіката з GitHub є досить складною задачею, оскільки потрібно очистити цей файл у всіх версіях та змінити історію репозиторію. Також важливо перевіряти історію репозиторію на предмет інших можливих витоків облікових даних. Проте, навіть якщо розробник зробив це, будь-які локальні копії інших розробників все ще можуть містити ці облікові дані сертифіката. Тому рекомендовано створити новий сертифікат і змінити код так, щоб завантажувати цей файл з захищеного місця, а не зберігати його у вигляді звичайного тексту.

Попередження про можливі наслідки незахищеного управління обліковими даними, їх викрадення або витік. Більш недавнім прикладом є порушення даних, яке сталося в Imperva у 2019 році через витік ключа API AWS, що дозволив зловмиснику отримати доступ до даних клієнта, таких як адреси електронної пошти та паролі. Очікується, що такі дорогі інциденти будуть продовжуватися, доки питання управління обліковими даними не стане пріоритетним.

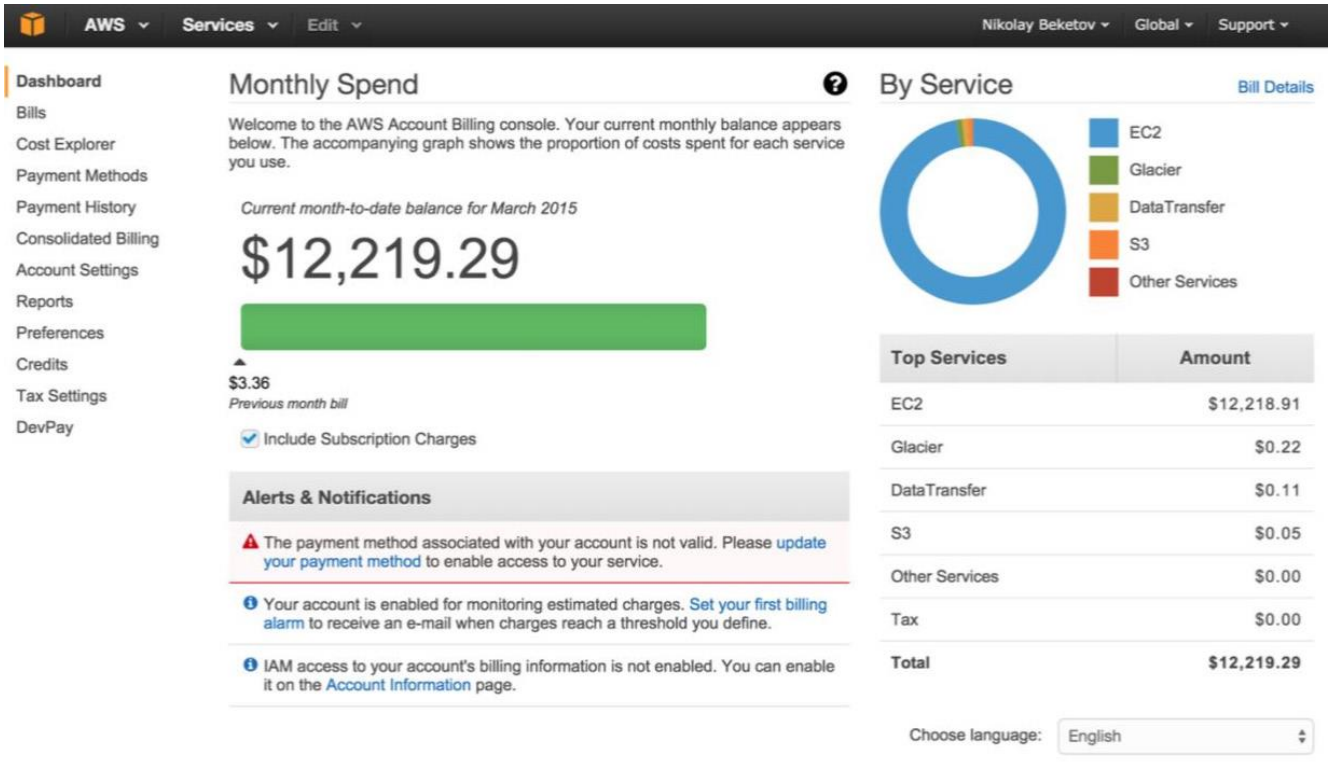


Рис.2.18. Скріншот інтерфейсу керування AWS компанії

Тому найкращим способом збереження даних в інфраструктурі, що ґрунтується на хмарних технологіях, є уникнення виокремлення необхідності в облікових даних[14].

Для більшості користувачів можна надати дозволи в рамках хмарної інфраструктури для доступу до хмарних ресурсів без потреби включати облікові дані в коді користувача. Зазвичай це стосується застарілих програм або гібридних хмарних архітектур, які досі потребують облікових даних у своєму коді (рис.2.19).

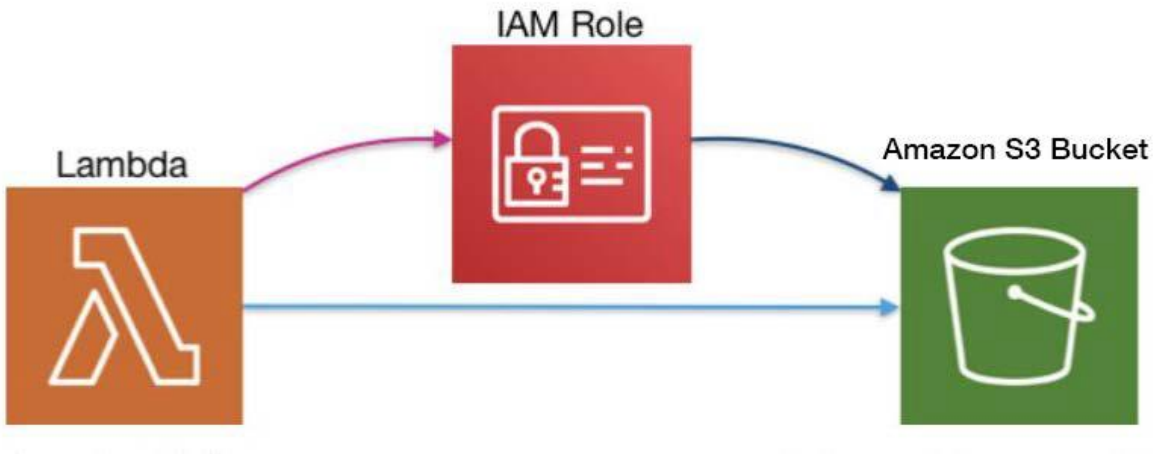


Рис.2.19. Приклад розподілення ролей в сучасній хмарній інфраструктурі без необхідності викриття облікових даних

Висновки до розділу 2

Досліджено інфраструктурні ресурси Amazon Web Services (AWS), які являються основою для будь-якої інфраструктури або додатку. Зазначено лише кілька прикладів інфраструктурних ресурсів, які надає AWS. Ці ресурси дозволяють користувачам розгортати та управляти інфраструктурою у хмарному середовищі з високою гнучкістю та масштабованістю і включають: Amazon Elastic Compute Cloud (Amazon EC2), Amazon Simple Storage Service (Amazon S3), Amazon Relational Database Service (Amazon RDS), Amazon Virtual Private Cloud (Amazon VPC).

Додатково виокремлено складові, що спрощують вивчення та дослідження доступних послуг AWS, а саме: Amazon Elastic Load Balancing (Amazon ELB).

Amazon Route 53, Amazon Simple Queue Service (Amazon SQS), Amazon Elastic Block Store (Amazon EBS), Amazon Identity and Access Management (IAM).

Виокремлено, що безпека в Amazon Web Services (AWS) це критичний аспект для користувачів, оскільки AWS забезпечує низку інструментів і послуг для забезпечення безпеки даних, додатків та інфраструктури в хмарному середовищі.

Надано характеристики головним веб-загрозам для хмарного середовища та інфраструктури, а саме: DDoS атаки, злом акаунту, SQL Injection, атаки на додатки, Cross-Site Scripting (XSS), зловмисні завантаження файлів, фішинг-атаки, недоліки налаштування та ін.

Підкреслено, що хмарні постачальники мають широкий спектр послуг, які пропонують можливості зберігання файлів. Сховище Amazon S3 є однією з найбільш впізнаваних послуг у галузі, якою користується значна кількість клієнтів, від маленьких користувачів до великих корпорацій. Однак, неправильна конфігурація Amazon S3 часто використовується для зловмисних цілей, таких як криптовикрадення, електронний скімінг і викрадання даних.

Окремо зазначено проблеми протидії криптоджекінгу і електронному скімінгу, та витоку облікових даних в AWS.

3 ДОСЛІДЖЕННЯ МЕХАНІЗМІВ ТА ЗАСОБІВ ЗАХИСТУ ХМАРНОЇ ІНФРАСТРУКТУРИ AMAZON AWS ВІД ВЕБ-ЗАГРОЗ

3.1 Дослідження служб безпеки AWS

Служби безпеки AWS - це послуги AWS, які в першу чергу надають способи захисту ресурсів в AWS (рис.3.1.).

Управління ідентифікацією та доступом AWS. AWS IAM дозволяє клієнтам безпечно контролювати доступ до своїх ресурсів AWS та AWS користувачів. У двох словах, IAM забезпечує автентифікацію та авторизацію для доступу до ресурсів AWS.

Він підтримує доступ до ресурсів AWS через веб-консоль управління, CLI, або програмно через API та SDK. Він має основні функції для контролю доступу як користувачі, групи, ролі та дозволи, а також розширені функції, такі як ідентифікація для інтеграції із наявною базою даних користувача, якаою може бути Microsoft Active Directory або Facebook, або Google. Можна визначити детальні дозволи для усіх ресурсів, а також використовувати тимчасові дані безпеки для надання доступу до зовнішніх користувачів поза обліковим записом AWS.

Віртуальна приватна хмара AWS. AWS VPC - це IaaS, який дозволяє створювати власну VPN у хмарі. Мережа може бути налаштованою для безпечного підключення до локального центру обробки даних.

Також можна налаштувати брандмауери для всіх ресурсів у VPC на рівні екземпляру та/або на рівні підмережі для управління трафіком передачі та виходу із VPC. VPC має функцію журналу потоків VPC, яка дозволяє збирати інформація щодо IP-трафіку VPC.

Система управління ключами AWS (KMS). AWS KMS - це послуга, яка допомагає керувати ключами, що використовуються для шифрування. Є кілька варіантів KMS, які включають принесення власних ключів та управління KMS поряд із тими, що генеруються AWS. Це повністю керована послуга, яка

інтегрується з іншими служби AWS, такими як AWS CloudTrail, щоб реєструвати всі дії для служб KMS. Ця послуга відіграє важливу роль у захисті даних, що зберігаються програмами[16].

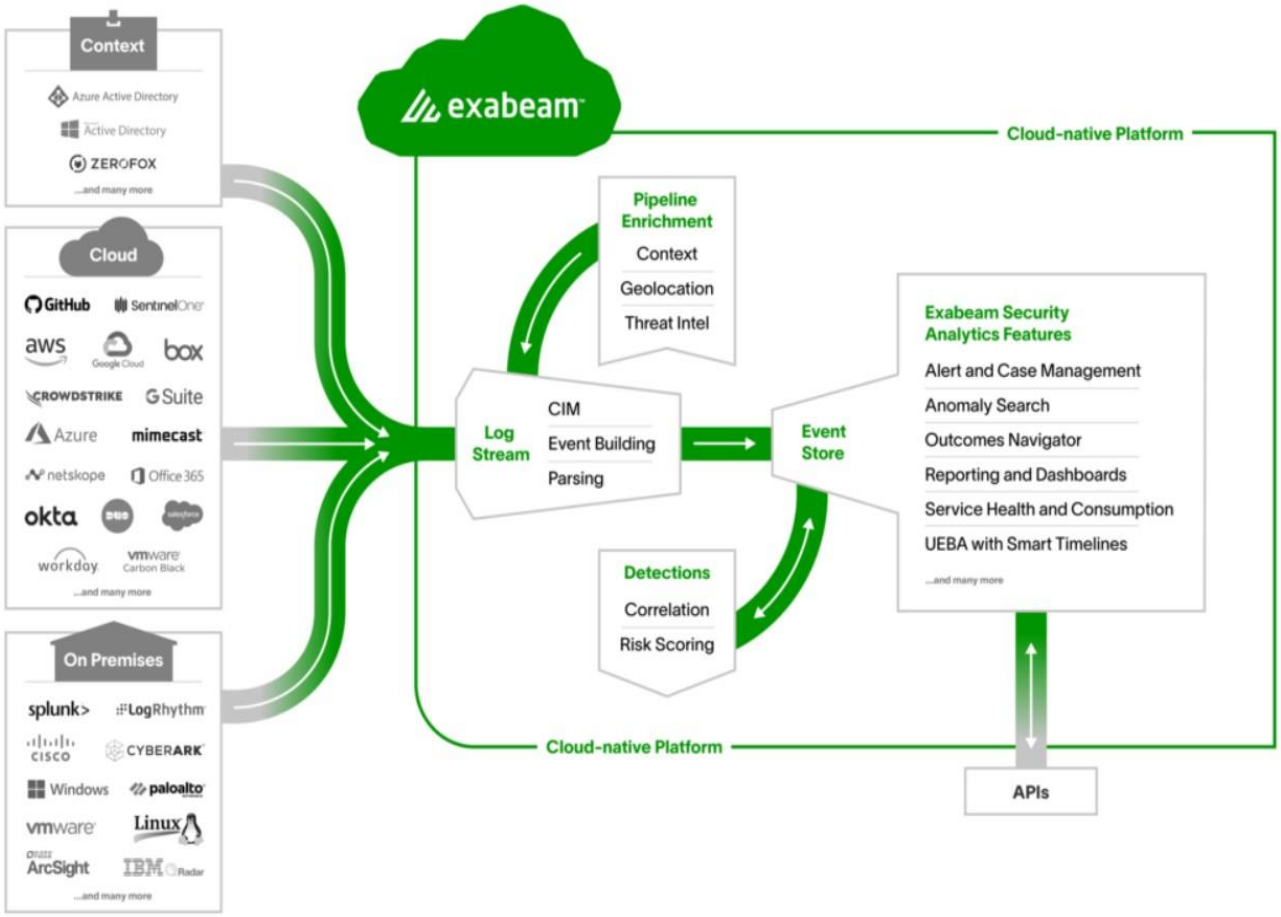


Рис.3.1. Служби безпеки AWS

Shield AWS. AWS shield захищає веб-програми, що працюють на AWS, від керованої розподіленої атаки відмови в обслуговуванні (DDoS). Це повністю керована послуга і має два варіанти, стандартний і вдосконалений.

Стандартна версія щита AWS пропонується всім клієнтам безкоштовно та забезпечує захист від найпоширеніших атак, націлених на програми або веб-сайти AWS. Розширена версія захисту AWS забезпечує вищий рівень захисту та інтеграцію з такими послугами, як брандмауери веб-додатків та доступ до команди відповідей AWS DDoS.

Брандмауер веб-додатків AWS (WAF). AWS WAF - це налаштовуваний брандмауер для веб-додатків, що дозволяє фільтрувати трафік які потрібно

отримати для веб-програм. Це керована послуга, яка може бути налаштовано з консолі керування або через AWS WAF API, щоб можна було мати контрольні пункти безпеки на різних рівнях у програмі (для розробників, інженерів DevOps, аналітиків безпеки тощо).

AWS CloudTrail. Це послуга ведення журналу, яка реєструє всі запити API і входи до облікового запису AWS. Допомагає з дотриманням вимог, аудитом та управлінням. Цей журнал можна проаналізувати, використовуючи засоби аналізу журналів для відстеження історії події. Послуга відіграє дуже важливу роль в автоматизації та безпеці.

AWS CloudWatch. Це служба моніторингу, яка надає показники, сигнали тривоги та інформаційні панелі для всіх послуг AWS, доступних у обліковому записі. Він інтегрується з іншими службами AWS, такими як AutoScaling, Elastic Load Balancer, AWS SNS, та AWS Lambda для автоматичного реагування для метричного порогу перетину.

Він також може збирати та контролювати журнали. AWS CloudWatch може також використовуватись для збору та моніторингу нестандартних показників для ресурсів або програм AWS.

AWS Config. AWS Config - це послуга, яка дозволяє проводити аудит та оцінювати конфігурацію AWS ресурсів. Можна відображати історичну конфігурацію ресурсів AWS, щоб перевірити будь-який інцидент. Це допоможе перевірити відповідність, усунути несправності тощо. Користувач може скористатися цією послугою, щоб переконатися, що ресурси AWS залишаються сумісними та налаштованими відповідно до базової конфігурації. Ця послуга забезпечує постійний моніторинг та безперервну оцінку конфігурації ресурсів AWS.

Артефакт AWS. Ця послуга надає усі документи, пов’язані із дотриманням вимог. AWS Artificat - це портал на замовлення для самообслуговування, присвячений відповідності та аудиту інформація разом із окремими угодами, такими як додаток до бізнесу та нерозголошення домовленості тощо[17].

3.2 Виокремлення основних обов'язків користувачів щодо безпеки

В контексті безпеки сервісів Amazon Web Services (AWS), розподіл відповідальності між AWS та їхніми клієнтами є визначальним. AWS покладає на клієнтів обов'язки забезпечення безпеки усіх використовуваних ними сервісів. Специфічно, клієнти несуть відповідальність за захист усіх елементів, розміщених у хмарному середовищі, включаючи дані, програми, та інші ресурси.

В контексті IaaS (Infrastructure as a Service) служб, клієнти відповідають за захист мережі та екземплярів, тоді як у сфері контейнерних сервісів вони забезпечують захист баз даних. Детальніше, у випадку сервісів інфраструктури AWS, клієнти мають відповідати за:

- Захист даних клієнта,
- Безпеку застосунків клієнта,
- Управління операційною системою,
- Конфігурацію мережі та брандмауера,
- Ідентифікацію користувачів та управління доступом,
- Управління серверами,
- Захист даних (транспортування, зберігання, резервне копіювання),
- Забезпечення високої доступності та автоматичне масштабування ресурсів.

Для контейнерних служб AWS, клієнти несуть відповідальність за:

- Захист даних клієнта,
- Конфігурацію VPC та брандмауера,
- Управління ідентифікацією та доступом,
- Забезпечення високої доступності,
- Захист даних (транспортування, зберігання, резервне копіювання),
- Автоматичне масштабування ресурсів.

Щодо абстрактних служб AWS, клієнтська відповідальність включає: захист даних клієнта; захист даних у стані спокою через власне шифрування; ідентифікація користувачів та управління доступом.

За перехід від інфраструктурних до абстрактних служб AWS, сфера відповідальності клієнтів за безпеку звужується, при цьому більшу частину безпеки забезпечує AWS. Важливо, що інфраструктурні служби AWS, такі як Amazon EC2, Amazon S3, та Amazon VPC, повністю контролюються клієнтами. Це означає, що клієнти мають налаштовувати параметри безпеки для доступу та управління цими ресурсами. Наприклад, в контексті екземплярів EC2, клієнти відповідають за управління операційною системою, оновлення, встановлення, та обслуговування програмного забезпечення та утиліт, а також конфігурацію груп безпеки для кожного екземпляра[18].

3.3 Дослідження особливостей використання AWS CloudTrail

AWS CloudTrail представляє собою службу Amazon Web Services, призначену для забезпечення можливості проведення оперативного аудиту, аудиту ризиків, а також контролю за дотриманням норм і політик у рамках клієнтського облікового запису AWS. Ця служба фіксує дії, здійснені користувачами, ролями чи іншими службами AWS, класифікуючи їх як події в системі CloudTrail.

До цих подій належать дії, виконані через AWS Management Console, AWS Command Line Interface, а також через SDK та API AWS.

CloudTrail автоматично активізується у обліковому записі AWS з моменту його створення, вимагаючи мінімального або відсутнього ручного налаштування. Всі активності, що відбуваються у обліковому записі AWS, систематично реєструються як події CloudTrail, забезпечуючи постійний моніторинг та аудит оперативної діяльності[19].

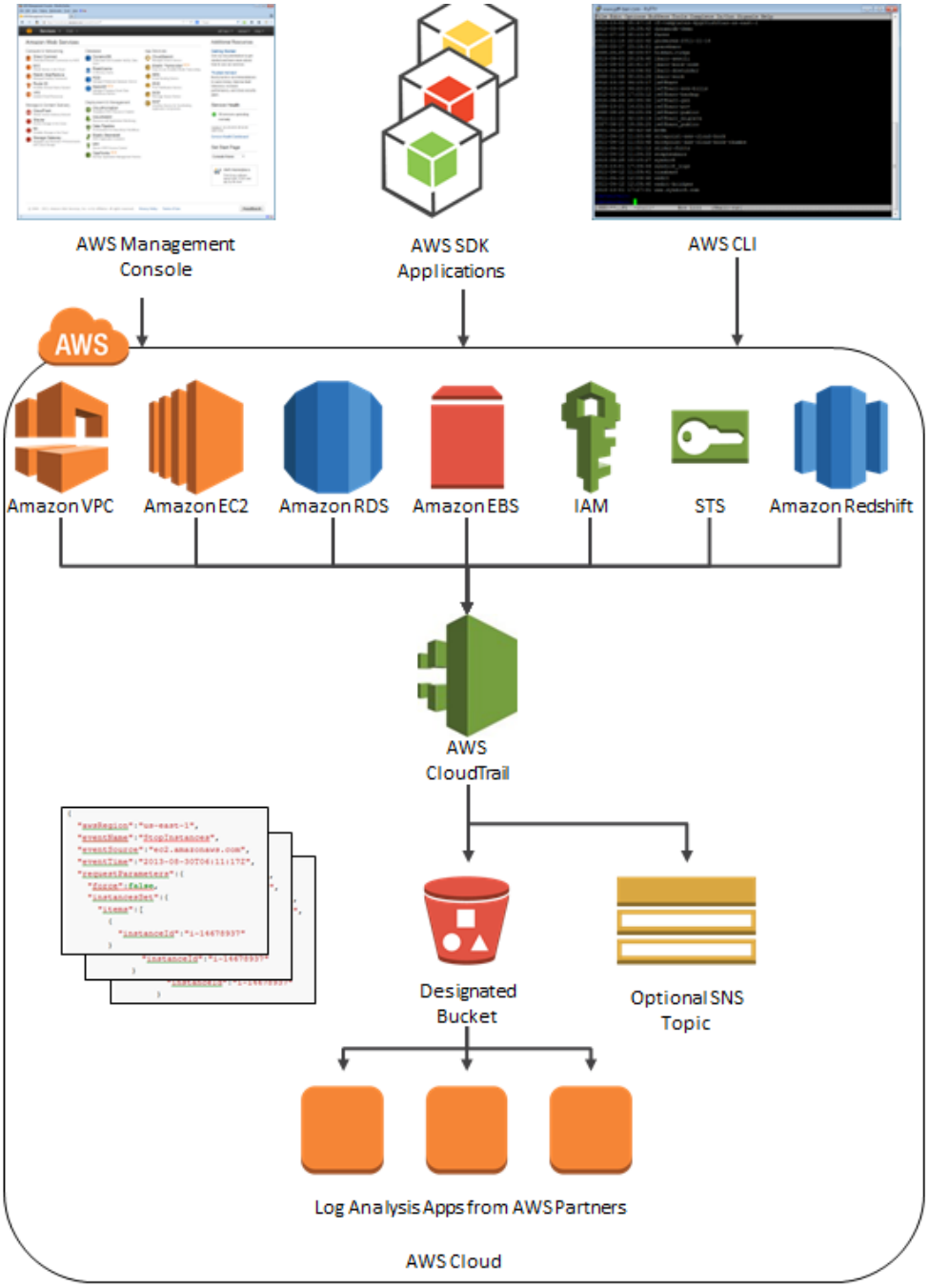


Рис.3.2. Механізм керування CloudTrail

Консоль CloudTrail. Користувач може використовувати та керувати сервісом за допомогою консолі AWS CloudTrail. Консоль надає інтерфейс користувача для виконання багатьох завдань CloudTrail, таких як:

- Перегляд останніх подій та історії подій для вашого облікового запису AWS;
- Завантаження відфільтрованого або повного файлу подій керування за

останні 90 днів;

- Створення та редагування маршрутів CloudTrail;
- Створення та редагування сховищ даних подій CloudTrail Lake;
- Виконання запитів до сховищ даних подій.

Налаштування маршрутів CloudTrail, зокрема:

- Вибір «відра» Amazon S3 для трейлів;
- Встановлення префікса;
- Налаштування доставки в журнали CloudWatch;
- Використання ключів AWS KMS для шифрування даних слідів;
- Увімкнення сповіщень Amazon SNS для доставки файлів журналу на шляхах;
- Додавання та керування тегами для маршрутів.

Налаштування сховищ даних подій CloudTrail Lake, зокрема інтеграція сховищ даних про події з партнерами CloudTrail або з власними програмами, щоб реєструвати події з джерел за межами AWS.

CloudTrail CLI. Інтерфейс командного рядка AWS — це уніфікований інструмент, який можна використовувати для взаємодії з CloudTrail з командного рядка.

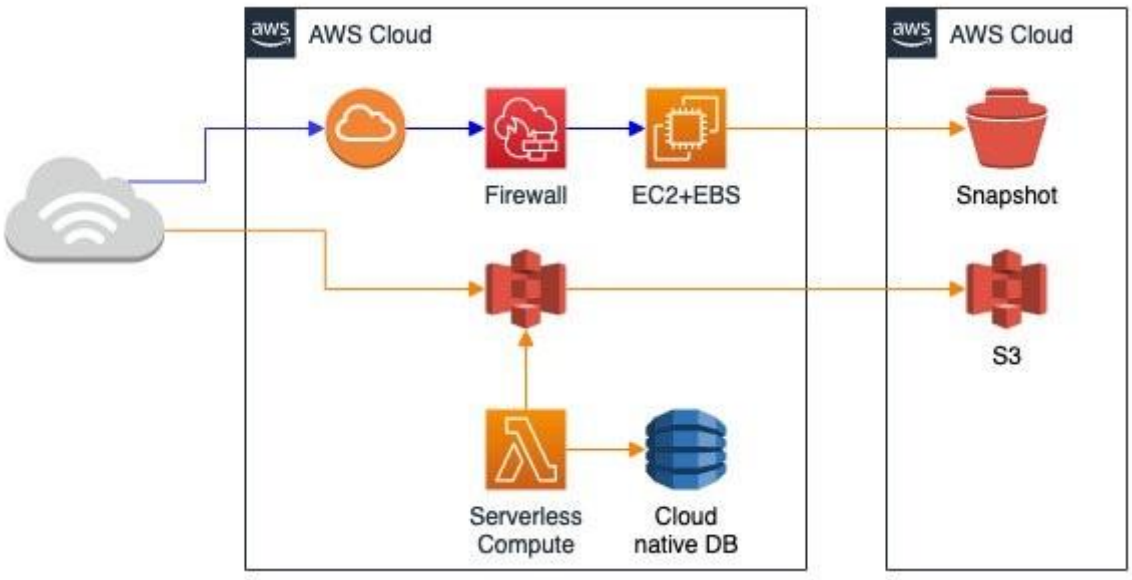


Рис.3.3. Інтерфейс командного рядка AWS

API CloudTrail. Окрім консолі та інтерфейсу командного рядка, також можна використовувати CloudTrail RESTful API для безпосереднього програмування CloudTrail.

AWS SDK. Як альтернативу використанню API CloudTrail можна використовувати один із пакетів SDK AWS. Кожен SDK складається з бібліотек і прикладів коду для різних мов програмування та платформ. SDK забезпечують зручний спосіб створення програмного доступу до CloudTrail. Наприклад, можна використовувати SDK для криптографічного підпису запитів, керування помилками та автоматичного повторення запитів.

Стратегії тегування AWS. Тег — це визначений користувачем ключ і необов'язкове значення, яке можна призначити ресурсам AWS, таким як маршрути CloudTrail, сегменти Amazon S3, які використовуються для зберігання файлів журналів CloudTrail та багато іншого.

Додавши однакові теги до трейлів і сегментів Amazon S3, які використовуються для зберігання файлів журналів для трейлів, можна полегшити керування, пошук і фільтрацію цих ресурсів за допомогою груп ресурсів AWS. Користувач може застосувати стратегії додавання тегів, щоб допомогти послідовно, ефективно та легко знаходити ресурси та керувати ними[20].

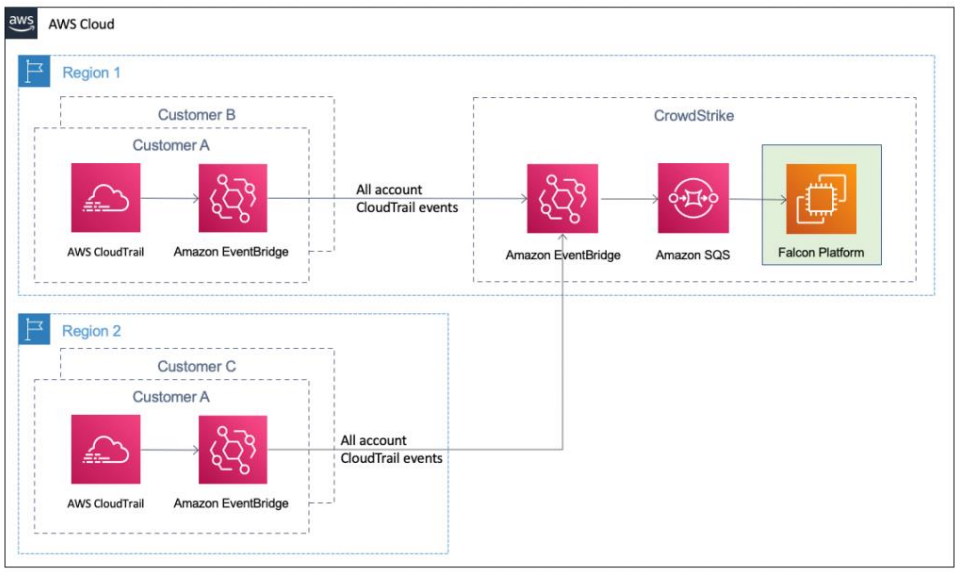


Рис.3.4. Стратегії тегування AWS

AWS Identity and Access Management. AWS Identity and Access Management — це веб-сервіс, який дозволяє клієнтам Amazon Web Services (AWS) безпечно контролювати доступ до ресурсів AWS. Використовуючи IAM, можна централізовано керувати дозволами, які контролюють, до яких ресурсів користувачі CloudTrail мають доступ.

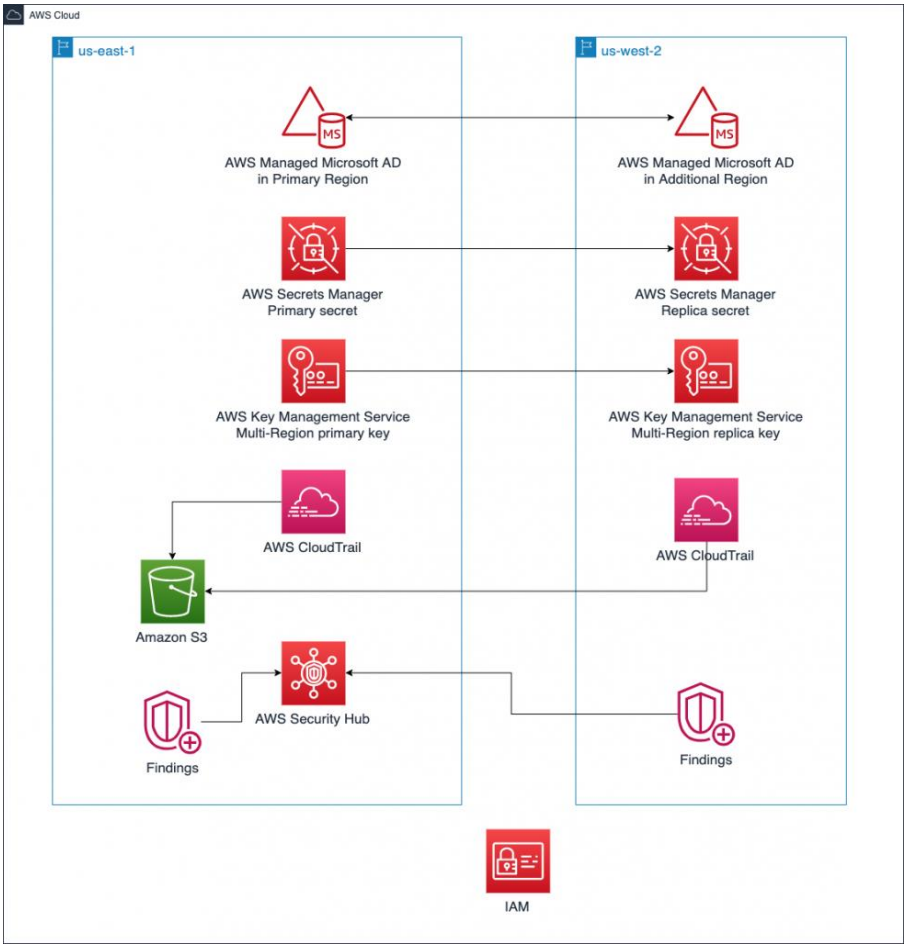


Рис.3.5. AWS Identity and Access Management

CloudTrail Insights. AWS CloudTrail Insights допомагає користувачам AWS визначати та реагувати на незвичні обсяги викликів API або помилки, зареєстровані у викликах API, постійно аналізуючи події керування CloudTrail.

Подія Insights — це запис незвичних рівнів активності API керування записом або незвичайних рівнів помилок, що повертаються під час діяльності API керування. Сторінка подробиць події Insights показує подію як графік незвичайної активності, а також показує час початку та завершення незвичайної діяльності разом із базовим рівнем, який використовується для визначення того, чи є ця

діяльність незвичною. За замовчуванням трейли не реєструють події CloudTrail Insights. У консолі можна реєструвати події Insights під час створення або оновлення сліду. Коли користувач використовує CloudTrail API, можна реєструвати події Insights, редагуючи налаштування існуючого трейлу за допомогою PutInsightSelectors API[20].

Служба маркерів безпеки AWS і CloudTrail. AWS Security Token Service (AWS STS) — це служба, яка має глобальну кінцеву точку, а також підтримує кінцеві точки для певного регіону.

Кінцева точка – це URL-адреса, яка є точкою входу для запитів веб-служб. Наприклад, *https://cloudtrail.us-west-2.amazonaws.com* є регіональною точкою входу в департамент «Захід США (Орегон)» для служби AWS CloudTrail. Регіональні кінцеві точки допомагають зменшити затримку у програмах.

Коли користувач використовує кінцеву точку AWS STS для певного регіону, трейл у цьому регіоні доставляє лише події AWS STS, які відбуваються в цьому регіоні. Наприклад, якщо використати кінцеву точку *sts.us-west-2.amazonaws.com,* слід у *us-west-2* додати лише події AWS STS, які походять із *us-west-2*.

Інтеграція CloudTrail з організаціями AWS. Обліковий запис керування для організації AWS Organizations може додати делегованого адміністратора для керування ресурсами CloudTrail організації.

Користувач може створити організаційний журнал або сховище даних подій організації в керуючому обліковому записі або делегованому обліковому записі адміністратора для організації, яка збирає всі дані подій для всіх облікових записів AWS в організації. Створення журналу організації допомагає визначити єдину стратегію реєстрації подій для організації.

Підтримка шифрування за допомогою KMS. CloudTrail створює файли журналів і надсилає їх у сегмент S3. За замовчуванням файли шифруються за допомогою серверного шифрування (SSE) S3, а потім прозоро розшифровуються, коли користувач їх читає. Завдяки сьогоднішньому запуску користувач тепер може надати ключ KMS для CloudTrail, і він використовуватиметься для шифрування файлів журналу. Як і у випадку з SSE, дешифрування є прозорим і автоматичним,

якщо у користувача є дозвіл читати об’єкт. Тому програми, які читають і обробляють файли журналів, не потребують жодних змін. Просто потрібно надати S3 дозвіл на розшифровку файлів (рис.3.6)

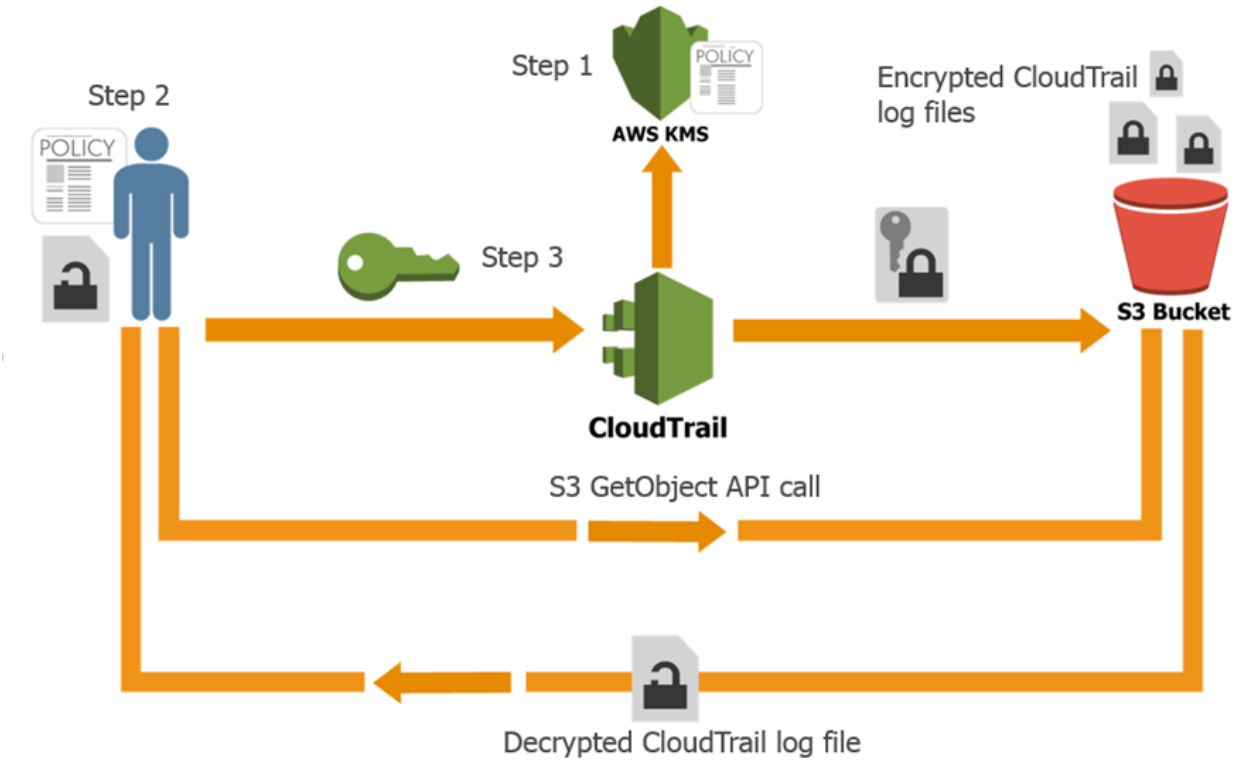


Рис.3.6. Розшифрування файлів CloudTrail

Алгоритм налаштування механізму розшифрування файлів CloudTrail:

1. Створення ключа KMS або використання наявного ключа KMS у тому самому регіоні, що й сегмент S3, куди клієнт повинен отримати файли журналу CloudTrail, і застосування політики KMS-CloudTrail;

2. Застосування дозволів на розшифрування до принципала (користувачі IAM, ролі, групи тощо), які матимуть доступ до файлів журналу CloudTrail;

3. Оновлення наявного трейлу за допомогою ключа KMS із кроку 1 (Можна ввімкнути шифрування під час створення трейлу, якщо використовується CLI).

Перевірка цілісності файлу журналу. Якщо необхідно провести аудит або дослідження безпеки, можна перевірити цілісність файлів журналу CloudTrail, що зберігаються у сегменті S3, і визначити, чи були вони видалені чи змінені після того, як CloudTrail доставив файл журналу у сегмент S3 (очікується, що вони

будуть незмінними). Нова функція перевірки цілісності файлу журналу CloudTrail дозволяє це зробити[21].

Щоб перевірити цілісність файлів журналу, потрібно ввімкнути перевірку файлу журналу. Це можна зробити, встановивши для параметра «Увімкнути перевірку файлу журналу» значення «Так» у розширеному розділі конфігурації трейлу (рис.3.7)

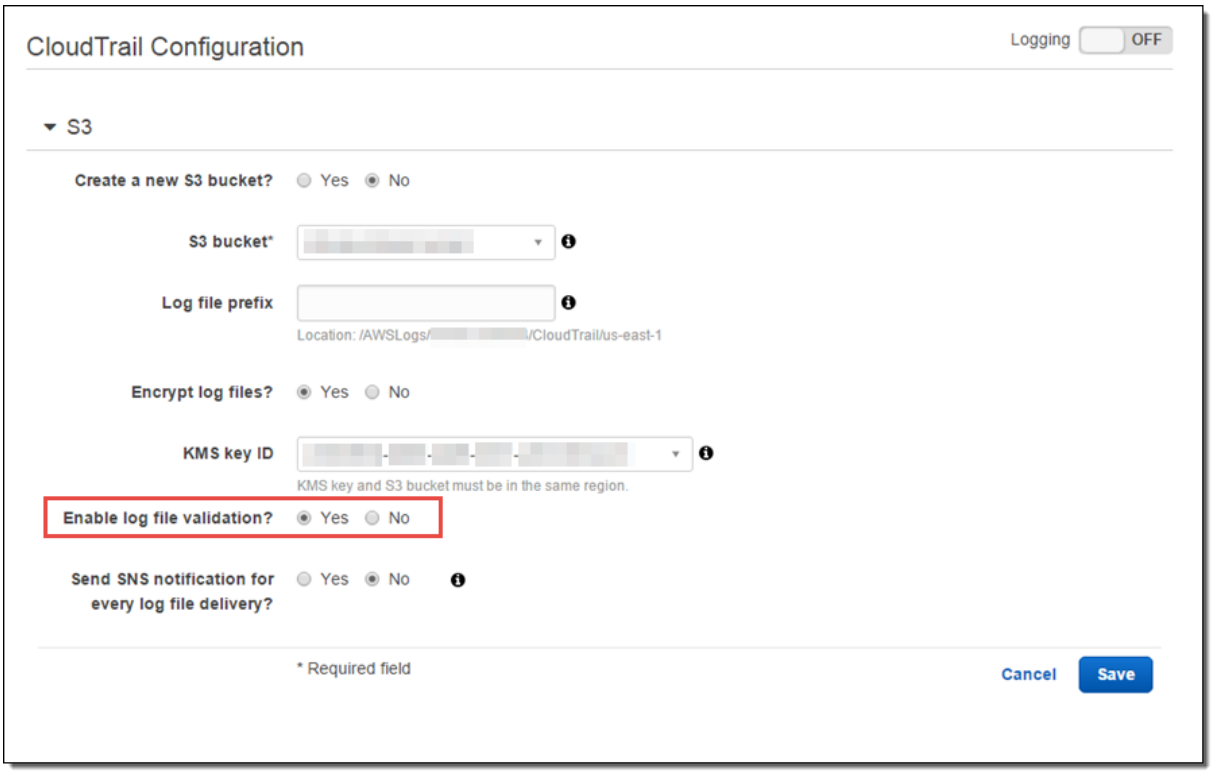


Рис.3.7. Увімкнення перевірки файлів журналу

Щойно було ввімкнено перевірку цілісності файлу журналу, CloudTrail почне щогодини доставляти файли дайджесту в те саме «відро» S3, де отримуються файли журналу CloudTrail, але з іншим префіксом:

- Файли журналу CloudTrail надсилаються до */optional_prefix/AWSLogs/AccountID/CloudTrail/**.
- Файли дайджесту CloudTrail надсилаються до */optional_prefix/AWSLogs/AccountID/CloudTrail-Digest/**.

Цей макет дозволяє програмам, які інтегруються з CloudTrail, обробляти файли журналу без внесення змін. Також можна застосувати різні та детальні дозволи на керування доступом до файлів журналу та файлів дайджесту.

Файли дайджесту містять інформацію про файли журналу, які було доставлено до сегмента S3, хеш-значення для цих файлів журналу, цифрові підписи для попереднього файлу дайджесту та цифровий підпис для поточного файлу дайджесту в розділі метаданих S3.

Щоб перевірити файли журналів CloudTrail, можна скористатися інтерфейсом командного рядка (CLI) AWS і просто запустити таку команду, щоб перевірити файли журналів:

*$ aws cloudtrail validate-logs *
*--trail-arn arn:aws:cloudtrail:us-west-2:111111111111:trail/Trailname *
--start-time 2015-09-24T00:00:00Z --region=us-west-2

Код валідації:

*aws:cloudtrail:us-west-2:111111111111:trail/Trailname between *
2015-09-24T00:00:00Z and 2015-09-25T18:56:41Z
Results requested for 2015-09-24T00:00:00Z to 2015-09-25T18:56:41Z
Results found for 2015-09-24T00:30:26Z to 2015-09-25T18:56:41Z:
43/43 digest files valid
31/31 log files valid

Якщо один або кілька файлів журналу було видалено, буде виведено результат, який виглядає наступним чином:

*Log file s3://mybucket-CTlogs/AWSLogs/111111111111/CloudTrail/us-west-2/2015/09/22/111111111111_CloudTrail_us-west-2_20150922T1720Z_Jy4SwZotr3eTI2FM.json.gz *
INVALID: not found
Results requested for 2015-09-22T00:00:00Z to 2015-09-25T18:42:03Z
Results found for 2015-09-22T00:30:26Z to 2015-09-25T18:42:03Z:
43/43 digest files valid
30/31 log files valid, 1/31 log files INVALID

*Log file s3://mybucket-CTlogs/AWSLogs/111111111111/CloudTrail/us-west-2/2015/09/25/111111111111_CloudTrail_us-west-2_20150925T1845Z_lU58MiCsXyI1U3R1.json.gz *

INVALID: hash value doesn't match

Results requested for 2015-09-24T00:00:00Z to 2015-09-25T21:44:50Z

Results found for 2015-09-24T00:30:26Z to 2015-09-25T21:44:50Z:

45/45 digest files valid

35/36 log files valid, 1/36 log files INVALID

3.4 Виокремлення рекомендацій щодо створення та налаштування безпечного облікового запису в AWS

Реєстрація в AWS та налаштування облікового запису. Спочатку необхідно зареєструвати обліковий запис. Для цього потрібно перейти на сайт *https://aws.amazon.com* та натиснути «Create an AWS account». На цьому кроці відбувається введення реєстраційних даних(рис.3.8.).

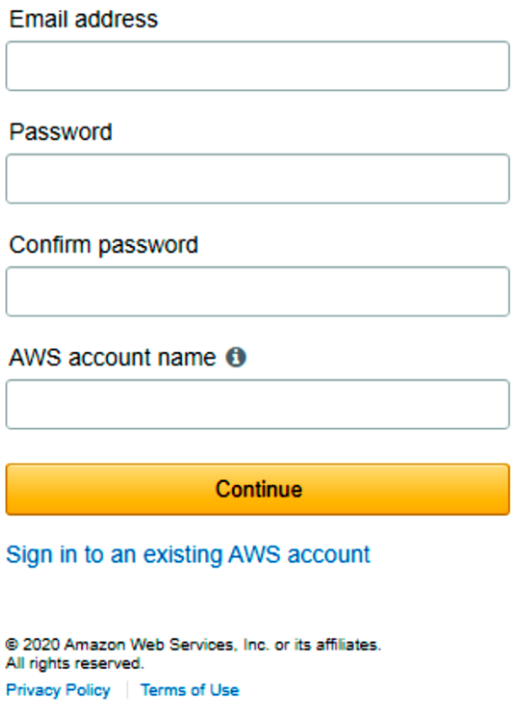


Рис.3.8. Вікно вводу реєстраційних даних

Після вводу необхідної інформації, система переводить до наступного вікна (рис.3.9.) з вводом контактів.

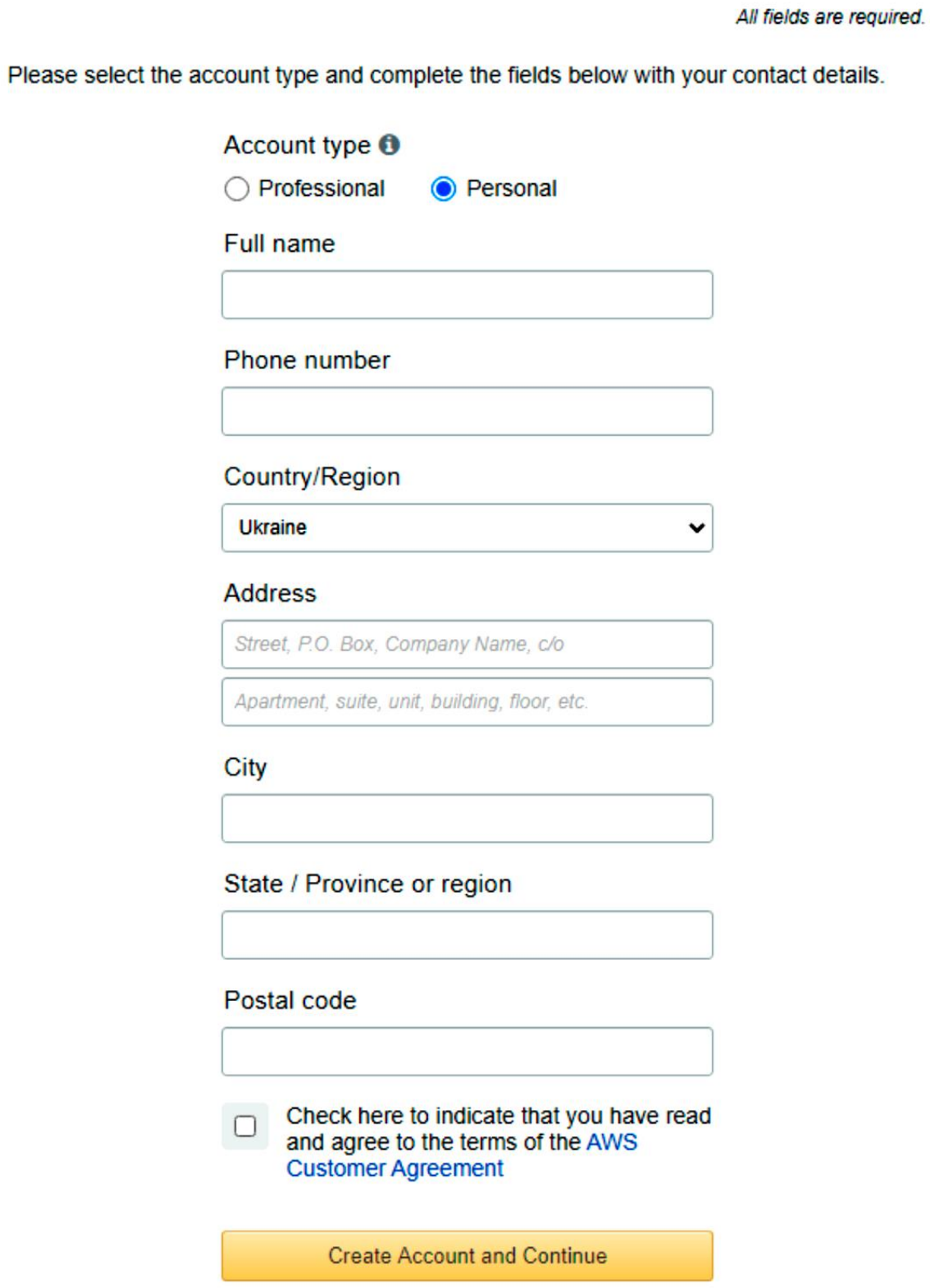


Рис.3.9. Вікно вводу контактів

Після успішної реєстрації здійснюється перехід на головну сторінку Amazon Web Services. На даній сторінці необхідно створити користувача для Terraform з адміністративним доступом до VPC та EC2[21].

Для цього необхідно перейти на вкладинку «Services» та знайти сервіс «IAM». Через інтерфейс IAM відкривається розділ «Users» та необхідно натиснути «Add user» (рис.3.10.). В наступному вікні необхідно ввести ім’я користувача та поставити позначку біля розділу «Programmatic access».

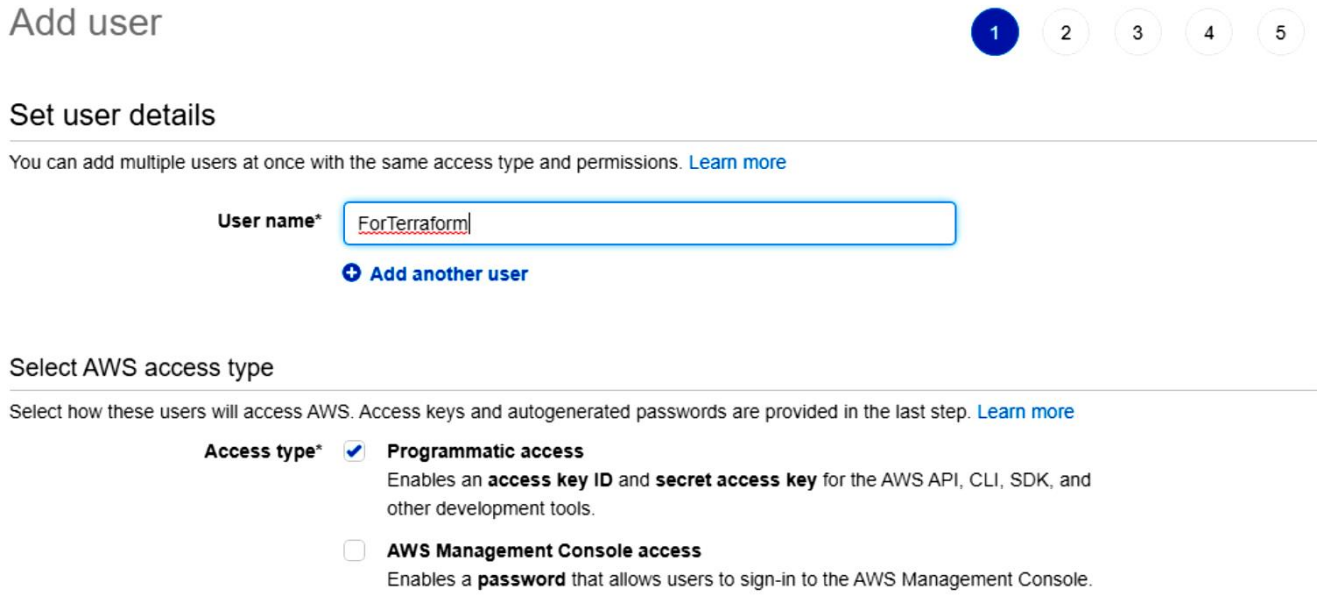


Рис.3.10. Створення користувача через IAM

Наступним кроком необхідно додати користувачу адміністративні права. Для цього необіхдно перейти до «Attach existing policies directly» (рис.3.11) і знайти списки «AmazonEC2FullAccess» та «AmazonVPCFullAccess». Їх необіхдно обрати та здійснити перехід до наступної сторінки.

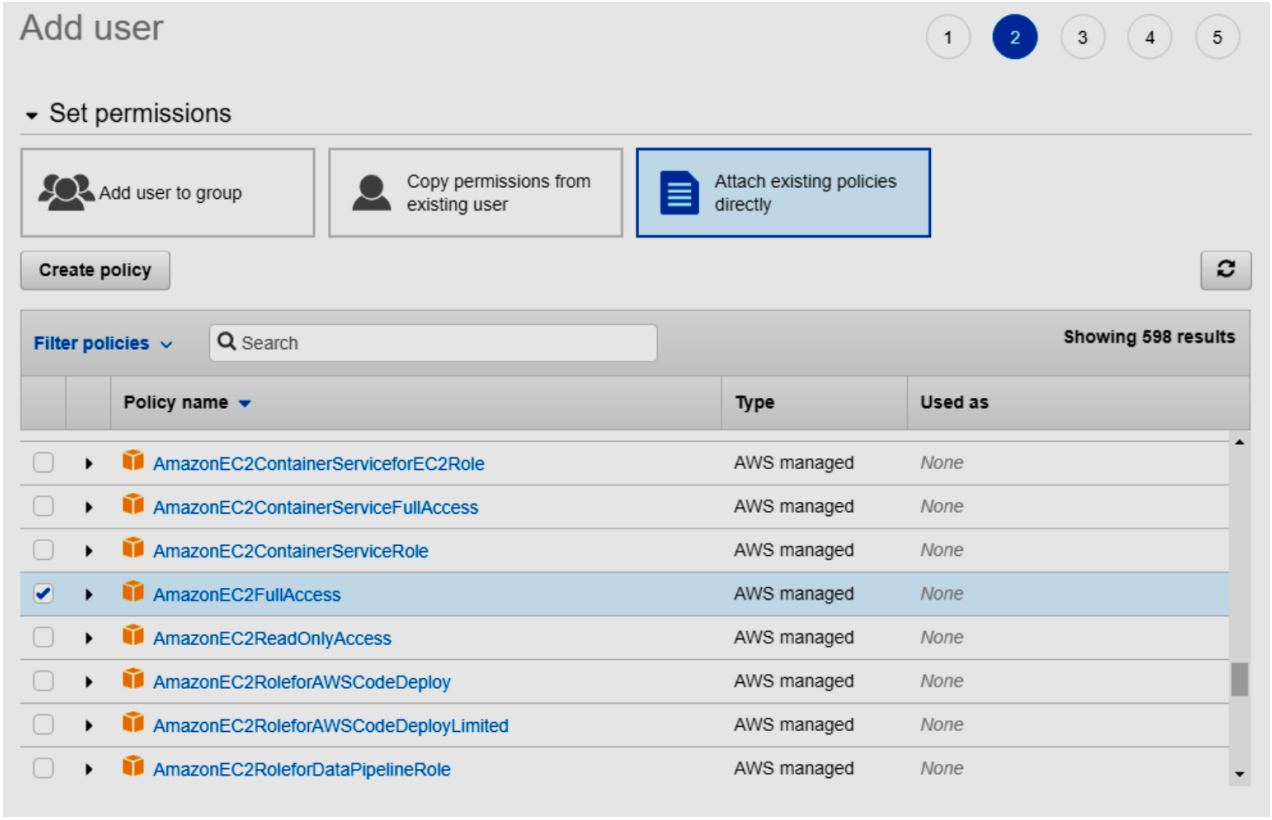


Рис.3.11. Додавання політики доступу

На наступній сторінці додаються теги для зручності (рис.3.12).

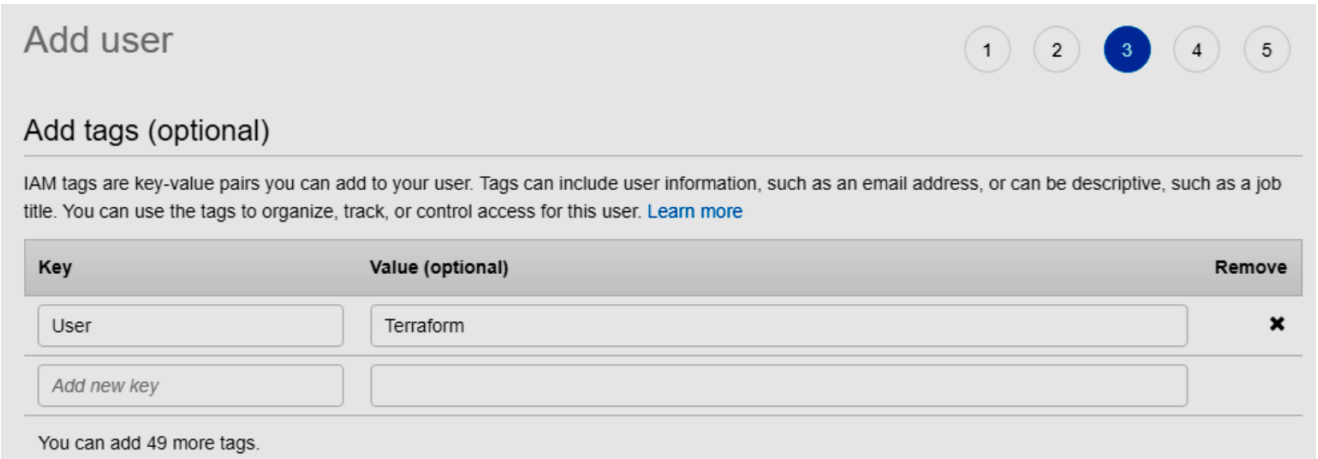


Рис.3.11. Додавання тегів

Наступним кроком AWS дозволяє переглянути всі деталі та права користувача (рис.3.12). Даний перелік інформації потрібно ретельно перевірити, після чого перейти далі, до налаштувань[22].

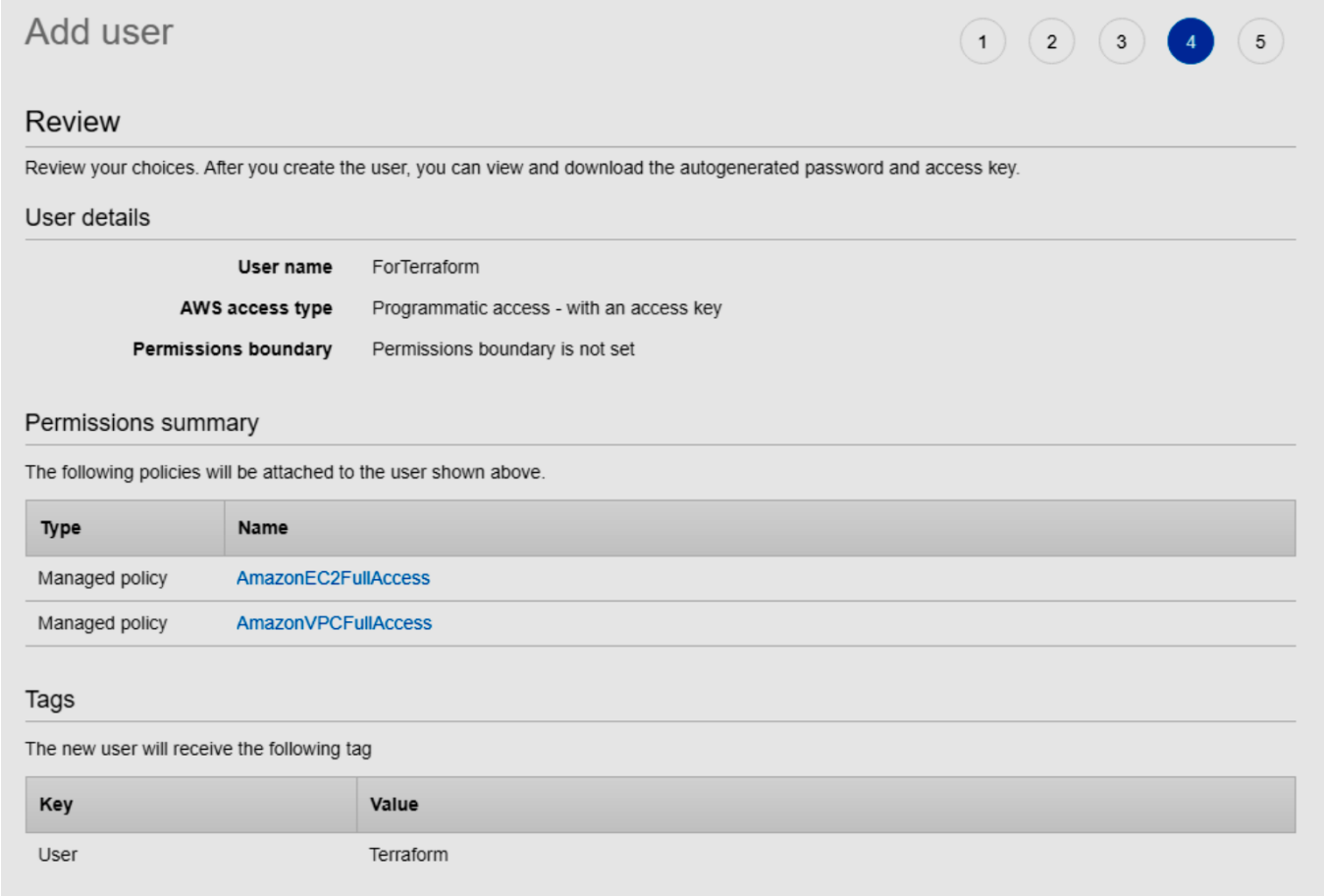


Рис.3.12. Попередній перегляд створеного користувача

Після цього натискається «Create user» та виникає вікно, яке повідомляє про успішне створення нового користувача.

Обов'язково потрібно зберегти «Access key ID» та «Secret access key» (рис.3.13.), оскільки вони знадобляться в майбутньому для керування обліковим записом.

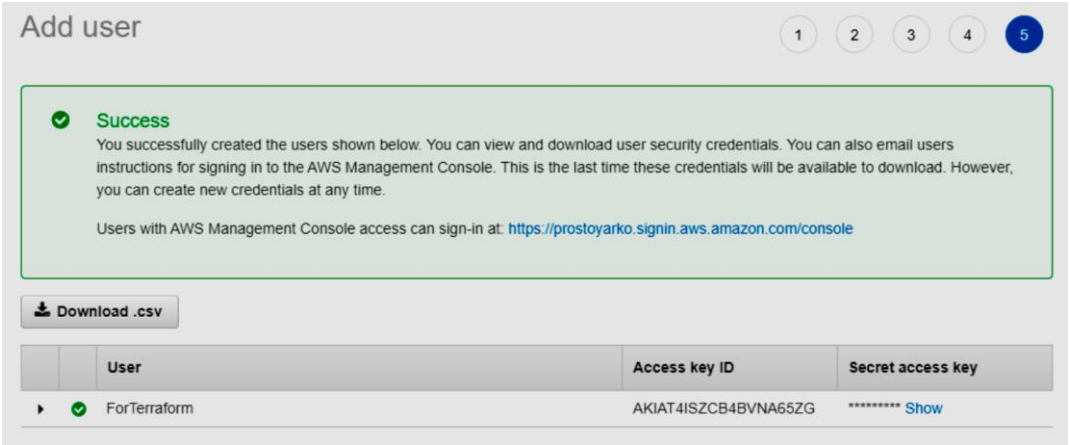


Рис.3.13. Облікові дані користувача

Наступним кроком потрібно створити групу, в яку будуть записуватися лог-файли. Для цього необхідно звернутися до CloudWatch > Log groups > Create log group (рис.3.14.). Зазначається ім’я групи, залишаються значення «Retention setting» в статусі «Never expire». Після чого група вважається створеною.

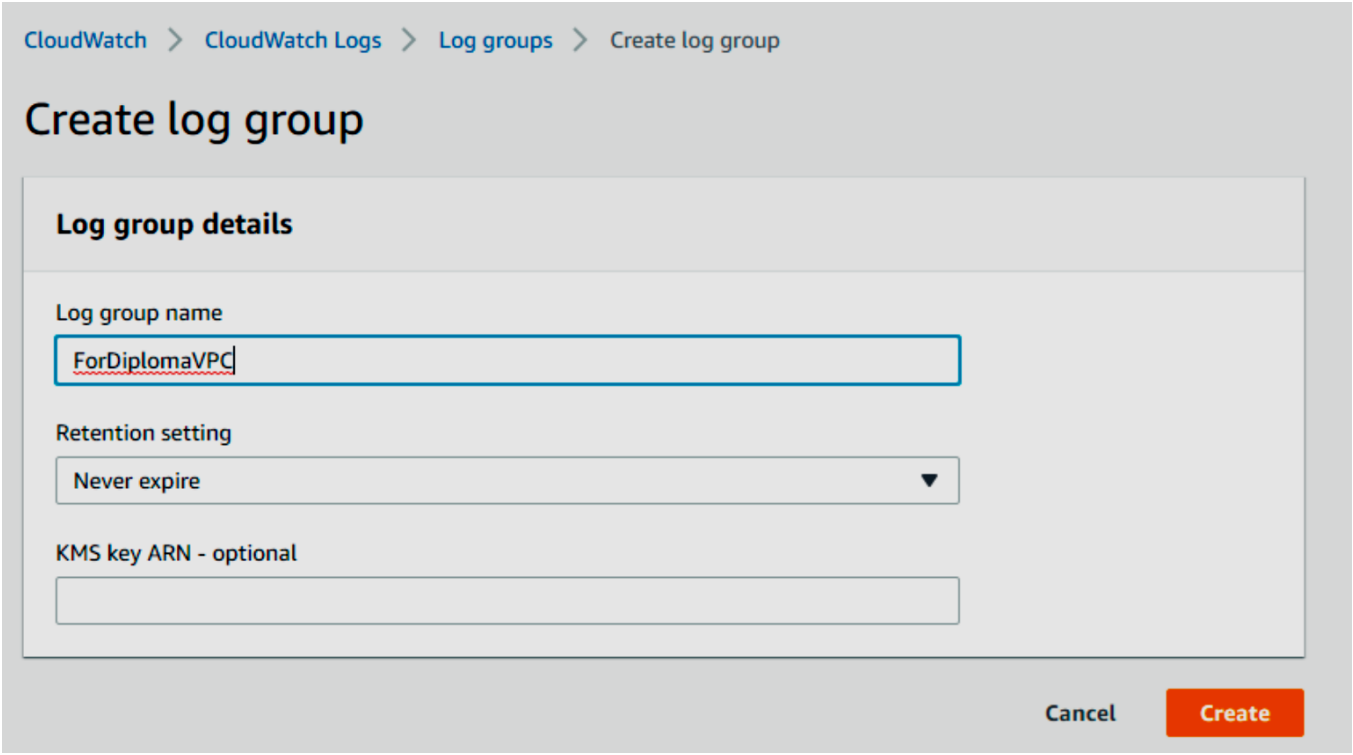


Рис.3.14. Група для лог-файлів

Далі, необхідно перейти до сервісу VPC та відкрити створену мережу. Для цього, у вкладинці «Flow logs» натиснути «Create flow log». Ввести назву для записів, встановити фільтрацію за всіма пакетами з максимальним інтервалом в 10 хвилин та вказати групу для записів даних та роль доступу, які були створені раніше (рис.3.15.).

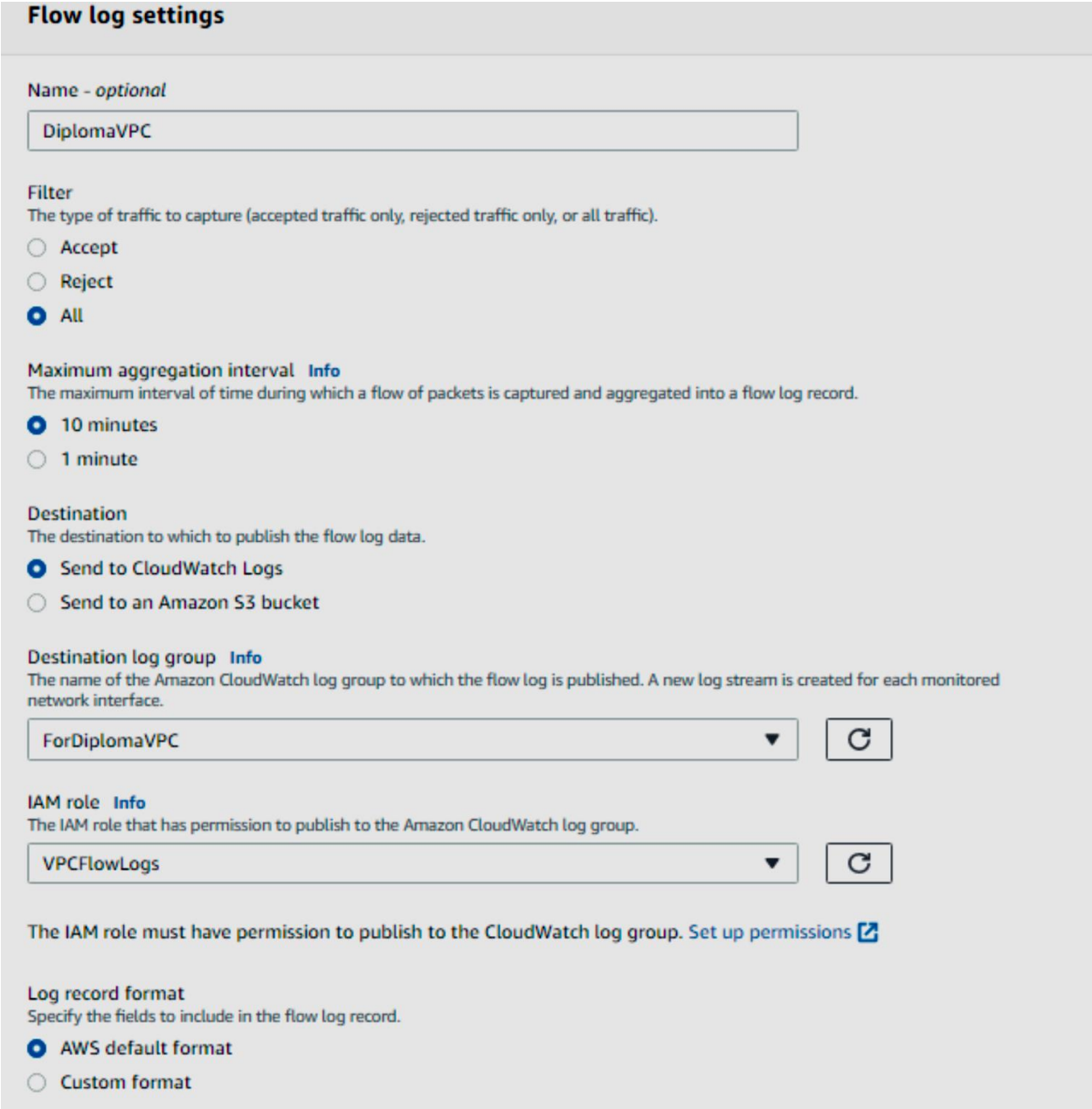


Рис.3.15. Налаштування групи логів

Повернувшись до сервісу CloudWatch, необхідно перейти до відповідної групи та спостерігати за відправленими та отриманими пакетами (рис.3.16).

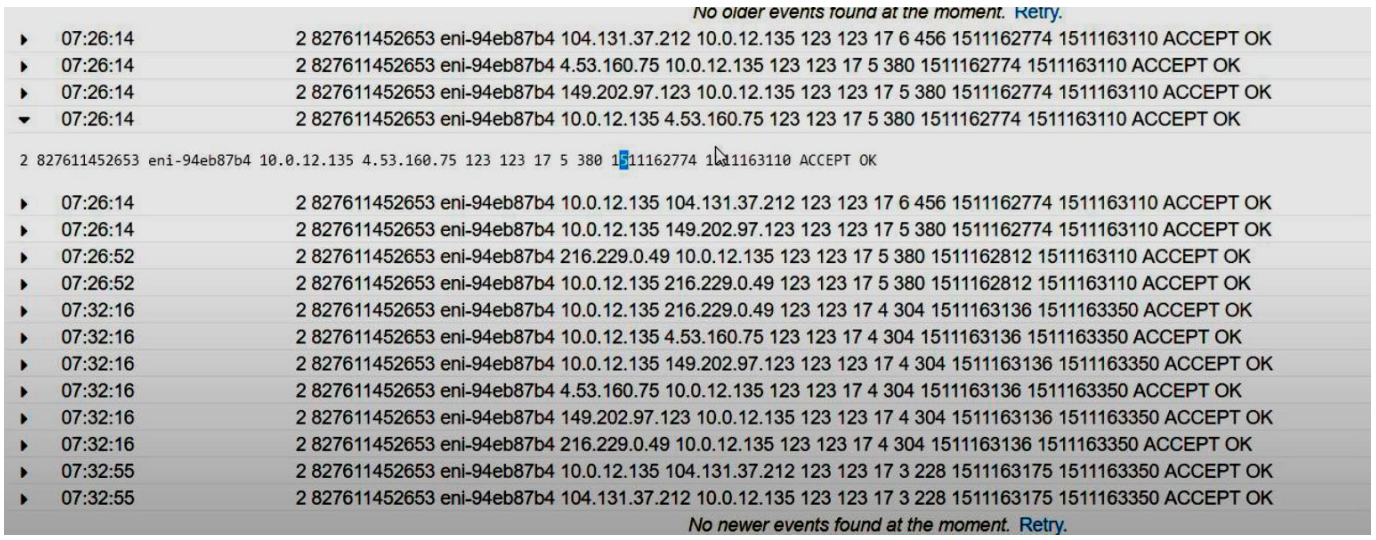


Рис.3.16. Перегляд логів

Запис подій в обліковому записі через CloudTrail. Для коректного налаштування запису подій до CloudTrail необхідно перейти до однойменного сервісу в AWS та натиснути «Create trail». Вказується відповідне ім’я. Разом з записом в CloudTrail буде створено «S3 bucket» до якого записуватимуться події в хмарі (рис.3.17).

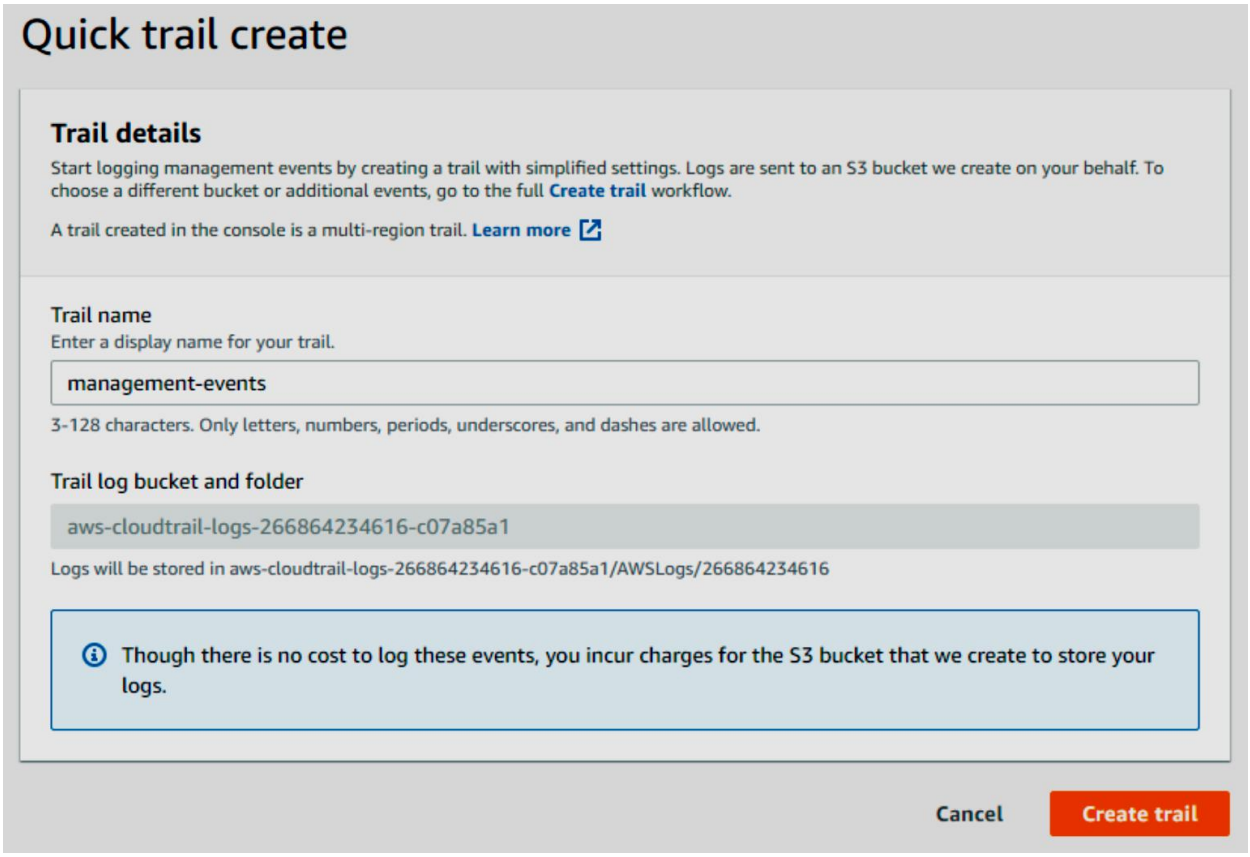


Рис.3.17. Запис подій до CloudTrail

Надалі, вже можна переглядати лог-файли для конкретного регіону[23]. Для цього, перейшовши в S3, потрібно знайти відповідний bucket. В ньому знаходиться файл в форматі *.json*. Після завантаження його, вміст файлу виглядатиме наступним чином (рис.3.18) [24].


```
{"Records":[{"eventVersion":"1.08","userIdentity":
{"type":"AWSService","invokedBy":"cloudtrail.amazonaws.com"},
"eventTime":"2020-11-24T12:24:30Z",
"eventSource":"s3.amazonaws.com","eventName":"GetBucketAcl",
"awsRegion":"eu-central-1","sourceIPAddress":"cloudtrail.amazonaws.com",
"userAgent":"cloudtrail.amazonaws.com",
"requestParameters":{"bucketName":"aws-cloudtrail-logs-266864234616-c07a85a1",
"Host":"aws-cloudtrail-logs-266864234616-c07a85a1.s3.eu-central-1.amazonaws.com","acl":""},
"responseElements":null,"additionalEventData":{"SignatureVersion":"SigV4","CipherSuite":"ECDHE-RSA-AES128-SHA",
"bytesTransferredIn":0,"AuthenticationMethod":"AuthHeader",
"x-amz-id-2":"25LT+gVhk60jFZ/hpNudICY9KxyWsTAiwu4qBgxRwPXGJbN21Xuj6LJ0H44VLQP3h8y4nanXZ4U\u003d",
"bytesTransferredOut":480},"requestID":"695F48645C571D0F",
"eventID":"1bbfd443-f51f-45ab-8cfc-282c33c64ec7","readOnly":true,
"resources":[{"accountId":"266864234616","type":"AWS::S3::Bucket",
"ARN":"arn:aws:s3:::aws-cloudtrail-logs-266864234616-c07a85a1"}],
"eventType":"AwsApiCall","managementEvent":true,"eventCategory":"Management",
"recipientAccountId":"266864234616","sharedEventID":"d47245f1-71e8-44d3-9506-8383b84741a0"}]}
```


Рис.3.17. Перегляд логів

3.5 Розробка рекомендацій щодо забезпечення безпеки в Amazon Web Services (AWS) від веб-загроз

Забезпечення безпеки в Amazon Web Services (AWS) від веб-загроз - це важливий аспект діяльності будь-якої компанії, яка використовує хмарні послуги, тому повинна включати переважну кількість рекомендацій для забезпечення безпеки:

Оцінка загроз і ризиків. Визначення переважної кількості веб-загроз, які є найбільш критичними та тими, що загрожують хмарній інфраструктурі AWS.

Ідентифікація вразливостей. Проведення оцінки вразливостей інфраструктури та додатків в AWS. Виявлення слабких та вразливих складових компонентів, які можуть бути використані зловмисниками.

Захист ідентифікації і аутентифікації. Використання багаторівневої аутентифікації для захисту облікових записів користувачів і адміністраторів. Обмеження доступу до інфраструктури і ресурсів тільки для необхідних осіб.

Моніторинг та логування. Налаштування системи моніторингу та логування для виявлення підозрілих дій і атак. Розгляд використання служби AWS CloudTrail для запису подій і дій в інфраструктурі[25].

Захист даних. Використання шифрування для захисту даних в спокійному та транзитному режимі. Застосування політики управління даними для контролю доступу та ретенції даних.

Безпека додатків. Перевірка додатків на наявність вразливостей і використання патчів для усунення їх. Відокремлення компонентів додатків і ресурсів від інших, для мінімізації ризиків атак.

Захист мережі. Налаштування файерволів та політик безпеки для контролю мережевого трафіку. Використання служби AWS Security Groups для обмеження доступу до ресурсів.

Автоматизація безпеки. Використання інструментів автоматизації, таких як AWS Lambda, для автоматичного реагування на загрози і вразливості.

Навчання та свідомість. Навчання співробітників правилам безпеки в AWS і регулярне їх стажування щодо виокремлення веб-загроз та їх виявлення.

Запасний план і відновлення. Розробка плану відновлення в разі атаки чи компрометації, та регулярне проведення регламентних робіт щодо пом’якшення наслідків та відновлення.

Співпраця з AWS. Співпраця з AWS для забезпечення безпеки інфраструктури і використання найновіших засобів і служб безпеки.

Оновлення та аудит. Регулярне оновлення систем і додатків, а також проведення аудиту безпеки для виявлення вразливостей і усунення їх[26].

Висновки до розділу 3

Приведено огляд служб безпеки AWS, що включають: управління ідентифікацією та доступом AWS, віртуальну приватну хмару AWS, систему управління ключами (KMS), Shield AWS, брандмауер веб-додатків AWS (WAF), AWS CloudTrail, AWS CloudWatch, AWS Config, артефакт AWS, тощо

Виокремлено основні обов'язки користувачів щодо безпеки, а саме: захист даних, безпеку застосунків, управління ОС, конфігурація мережі та брандмауера, ідентифікація користувачів та управління доступом, управління серверами, захист даних (транспортування, зберігання, резервне копіювання), забезпечення високої доступності та автоматичне масштабування ресурсів.

Досліджено особливості використання AWS CloudTrail, та підкреслено, що зазначена служба призначена для забезпечення можливості проведення оперативного аудиту та аудиту ризиків, а також контролю за дотриманням норм і політик у рамках клієнтського облікового запису AWS. Ця служба фіксує дії, здійснені користувачами, ролями чи іншими службами AWS, класифікуючи їх як події в системі CloudTrail. До цих подій належать дії, виконані через AWS Management Console, AWS Command Line Interface, а також через SDK та API AWS.

Виокремлено алгоритм налаштування механізмів розшифрування файлів CloudTrail.

Надано перелік рекомендацій щодо створення та налаштування безпечного облікового запису в AWS (з можливістю доступу до записів усіх подій в обліковому записі через CloudTrail та можливістю перегляду лог-файлів з вмістом файлів)

Розроблено рекомендації щодо забезпечення безпеки в Amazon Web Services (AWS) від веб-загроз. Зазначено, що забезпечення безпеки в AWS від веб-загроз - це важливий аспект діяльності будь-якої компанії, яка використовує хмарні послуги, тому повинна включати переважну кількість рекомендацій для забезпечення безпеки: оцінка загроз і ризиків, ідентифікація вразливостей, захист ідентифікації і аутентифікації, моніторинг та логування, захист даних, безпека додатків, захист мережі, автоматизація безпеки, навчання та свідомість, запасний план і відновлення, співпраця з AWS, оновлення та аудит.

ВИСНОВКИ

В кваліфікаційній роботі отримано наступні наукові та науково-практичні результати:

1) Проаналізовано хмарні технології та хмарну інфрастурктуру, що спрямована на спільне використання обчислювальних ресурсів з метою підвищення ефективності та зниження витрат на адміністрування та інші IT-витрати.

2) Зазначено основні загрози безпеці хмарних інфраструктури, що включають: порушення даних, втрата даних, викрадення облікових записів, незахищені API, відмова в обслуговуванні, зловмисні інсайдери, зловживання хмарними службами, недостатній рівень обережності, спільні технологічні проблеми.

3) Виокремлено, що Amazon Web Service (AWS) пропонує різноманітні сервіси та інструменти, такі як ідентифікація та доступ, шифрування, ведення журналів, нагляд та дотримання норм, для забезпечення безпеки в хмарному середовищі. Ці сервіси AWS дозволяють виконувати широкий спектр завдань для задоволення всіх вимог безпеки, реєстрації користувачів, аудиту та відповідності в хмарному середовищі.

4) Досліджено інфраструктурні ресурси Amazon Web Services (AWS), які являються основою для будь-якої інфраструктури або додатку: Amazon Elastic Compute Cloud (Amazon EC2), Amazon Simple Storage Service (Amazon S3), Amazon Relational Database Service (Amazon RDS), Amazon Virtual Private Cloud (Amazon VPC).

5) Приведено огляд служб безпеки AWS, що включають: управління ідентифікацією та доступом AWS, віртуальну приватну хмару AWS, систему управління ключами (KMS), Shield AWS, брандмауер веб-додатків AWS (W
AWS CloudTrail, AWS CloudWatch, AWS Config, артефакт AWS, тощо

6) Досліджено особливості використання AWS CloudTrail, та підкреслено, що зазначена служба призначена для забезпечення можливості проведення

оперативного аудиту та аудиту ризиків, а також контролю за дотриманням норм і політик у рамках клієнтського облікового запису AWS. Виокремлено алгоритм налаштування механізмів розшифрування файлів CloudTrail.

7) Надано перелік рекомендацій щодо створення та налаштування безпечного облікового запису в AWS. Розроблено рекомендації щодо забезпечення безпеки в Amazon Web Services (AWS) від веб-загроз. Зазначено, що забезпечення безпеки в AWS від веб-загроз - це важливий аспект діяльності будь-якої компанії, яка використовує хмарні послуги, тому повинна включати переважну кількість рекомендацій для забезпечення безпеки: оцінка загроз і ризиків, ідентифікація вразливостей, захист ідентифікації і аутентифікації, моніторинг та логування, захист даних, безпека додатків, захист мережі, автоматизація безпеки, навчання та свідомість, запасний план і відновлення, співпраця з AWS, оновлення та аудит.

ПЕРЕЛІК ПОСИЛАНЬ

1. Peter Mell, Timothy Grance. The NIST Definition of Cloud Computing. National Institute of Standard and Technology US Department of Commerce, Special Publication 800-145. URL: http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf (дата звернення: 10.11.2023).

2. Sichao Wang. An analysis of the pros and cons of moving corporate data into the cloud. URL: https://cloudsecurityalliance.org/wpcontent/uploads/2012/02 (дата звернення 10.11.2023).

3. Vivek Kundra. Federal Cloud Computing Strategy. URL: https://www.whitehouse.gov/sites/default/files/omb/assets/egov_docs/federal-cloud-computingstrategy.pdf (дата звернення 12.11.2023).

4. Daniele Catteddu, Giles Hogben. Cloud Computing: Benefits, Risks and Recommendations for Information Security. ENISA (European Network and Information Security Agency). (дата звернення: 12.11.2023).

5. Wei Z., Yong P., Feng X., Zhonghua D. Modeling and Simulation of cloud computing. IEEE Asia Pacific Cloud Computing Congress (APCloudCC). 2021. Pp 20-24. (дата звернення: 12.11.2023).

6. Kevin McCaney. Amazon cloud service gets approval under fisma. URL: http://gcn.com/articles/2011/09/16/amazon-ec2-cloud-fisma.aspx (дата звернення: 16.11.2023).

7. LaKisha Ladson. The University of Texas at Dallas News Centre. URL: http://www.utdallas.edu/news/2013/3/4-22431_Cloud-Computing-Project-Wins-First-of-its-Kind-Goo_article-wide.html (дата звернення: 16.11.2023).

8. City University London. URL: http://www.city.ac.uk/news/2012/sep/city-universitylondon-wins-european-union-grant-for-cloud-security-research (дата звернення: 18.11.2023).

9. BCS. Data security experts 'to be trained at university', URL: http://www.bcs.org/content/conWebDoc/50515?utm_medium=email&utm_source=BCS+The+Chartered+Institute+for+IT&utm_campaign=2493324_securityspecialmay13&

dm_i=9U7,1HFV0,9QGEZI,51JJQ,1 (дата звернення: 18.11.2023).

10. Cloud Security Alliance. The notorious nine cloud computing top threats in 2021. URL: https://downloads.cloudsecurityallience.org/initiatives/top_threats/The_Notorious_nine_Cloud_Computing_Top_Threats_in_2021.pdf (дата звернення: 20.11.2023).

11. Eric G., John H., James R., Jim R. Steve S. Cloud Computing Roundtable, IEEE Security and Privacy. Vol. 8, No. 6. 2020. Pp 17-23. (дата звернення: 20.11.2023).

12. Nasser S. Abouzakhar. Critical Infrastructure Cybersecurity: A Review of Recent Threats and Violations. 2021. (дата звернення: 22.11.2023).

13. Chris, Brenton. Hypervisor vs. Host-based security, A comparison of the strengths and weaknesses of deploying cloud security with either a hypervisor or agent based model, Cloud Security Alliance. URL: https://cloudsecurityalliance.org/wp-content/uploads/2011/11/hypervisor-vshostbased-security.pdf (дата звернення: 23.11.2023).

14. Scott Ikeda. 2019 Sans Institute Cloud Security Survey Reveals Top Threats, Which Surprisingly Are Not DDoS Attacks. URL: https://www.cpomagazine.com/cybersecurity/2019-sans-institute-cloud-security-survey-reveals-top-threats-which-surprisingly-are-not-ddos-attacks/ (дата звернення: 23.11.2023).

15. Shayan Eskandari, Andreas Leoutsarakos, Troy Mursch, Jeremy Clark. A first look at browser-based Cryptojacking. URL: https://arxiv.org/abs/1803.02887 (дата звернення: 25.11.2023).

16. Yonathan Klijnsma. Spray and Pray: Magecart Campaign Breaches Websites En Masse Via Misconfigured Amazon S3 Buckets. URL: https://www.riskiq.com/blog/labs/magecart-amazon-s3-buckets/ (дата звернення: 25.11.2023).

17. Amazon Web Services . URL: https://docs.aws.amazon.com/cli/latest/reference/s3api/get-bucket-acl.html (дата звернення: 25.11.2023).

18. Dan Salmon. GitHub. S3Scanner. URL:

https://github.com/sa7mon/S3Scanner (дата звернення: 28.11.2023).

19. Sergiu Gatlan. Amazon to Disable S3 Path-Style Access Used to Bypass Censorship. URL: https://www.bleepingcomputer.com/news/security/amazon-to-disable-s3-path-styleaccess-used-to-bypass-censorship/ (дата звернення: 28.11.2023).

20. Amazon Web Services. Introducing Amazon S3 Block Public Access – another layer of protection for your accounts and buckets. URL: https://aws.amazon.com/about-aws/whatsnew/2018/11/introducing-amazon-s3-block-public-access/ (дата звернення: 28.11.2023).

21. Macy Bayern. TechRepublic. The 10 most popular container tools for businesses. URL: https://www.techrepublic.com/article/the-10-most-popular-container-tools-for-businesses/ (дата звернення: 30.11.2023).

22. Docker. Docker security. URL: https://docs.docker.com/engine/security/security/ (дата звернення: 30.11.2023).

23. Rain. Securing Docker Containers. URL: https://0x00secorg/t/securing-docker-containers/16913 (дата звернення: 30.11.2023).

24. Amazon Web Services. Manage IAM User Access Keys Properly. URL: https://docs.aws.amazon.com/en_pv/general/latest/gr/aws-access-keys-best-practices.html#iam-useraccess-keys (дата звернення: 30.11.2023).

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ (Презентація)

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ЗАХИСТУ ІНФОРМАЦІЇ
КАФЕДРА ІНФОРМАЦІЙНОЇ ТА КІБЕРНЕТИЧНОЇ БЕЗПЕКИ

КВАЛІФІКАЦІЙНА РОБОТА
на тему:
«ТЕХНОЛОГІЇ ЗАХИСТУ ХМАРНОЇ ІНФРАСТРУКТУРИ AMAZON AWS ВІД ВЕБ-ЗАГРОЗ»

Керівник:
к.т.н, доцент кафедри
СОБЧУК Андрій

Виконав:
здобувач вищої освіти
групи БСДМ-62
АНДРУЩЕНКО Максим

Київ 2024

ОБ'ЄКТ, ПРЕДМЕТ, МЕТА ТА ЗАВДАННЯ РОБОТИ 2

Об'єкт дослідження – захист хмарної інфраструктури від веб-загроз.

Предмет дослідження – технології захисту хмарної інфраструктури AMAZON AWS від веб-загроз.

Мета роботи – розробка рекомендацій щодо захисту хмарної інфраструктури Amazon AWS від потенційних веб-загроз.

Наукові завдання:
- проаналізувати хмарні технології та хмарну інфраструктуру;
- проаналізувати основні загрози безпеці хмарній інфраструктури;
- дослідити інфраструктуру Amazon Web Service (AWS);
- дослідити служби безпеки AWS;
- розробити рекомендації щодо забезпечення безпеки в Amazon Web Services (AWS) від веб-загроз.

ДОСЛІДЖЕННЯ ІНФРАСТРУКТУРНИХ РЕСУРСІВ AMAZON WEB SERVICES (AWS) 3

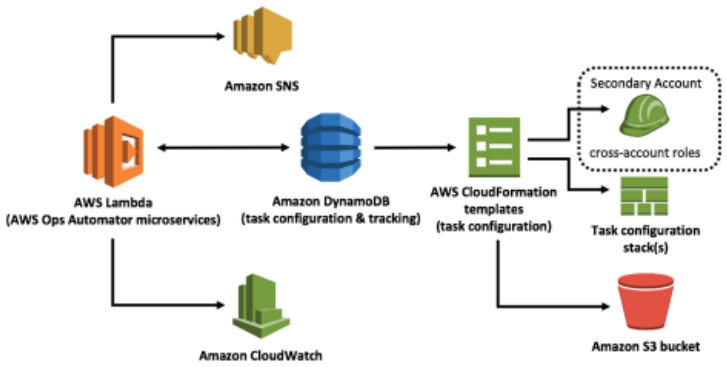


Рис.1. Інфраструктурні ресурси Amazon Web Services (AWS)

ВЕБ-ЗАГРОЗИ У ХМАРНОМУ СЕРЕДОВИЩІ

- DDoS атаки;
- Злам аккаунту (Account Compromise);
- SQL Injection та інші атаки на додатки;
- Cross-Site Scripting (XSS);
- Зловмисні завантаження файлів;
- Фішинг-атаки;
- Недоліки налаштування;
- Зловмисники, що використовують служби хмарних постачальників.

ОГЛЯД ВРАЗЛИВОСТЕЙ РЕАЛІЗАЦІЯ AMAZON S3

Рис.2. Скріншот скомпрометованого японського веб-сайту

Рис.3. Завантажені сценарії з сегмента Amazon S3

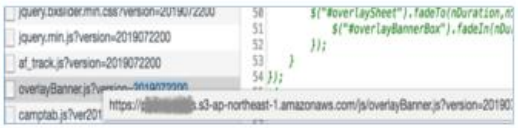


Рис.4. Завантажені файли JavaScript

АНАЛІЗ ВИТОКУ ОБЛІКОВИХ ДАНИХ ДЛЯ ЗАПИСУ AWS

Рис.5. Фрагмент коду з Pastebin, що демонструє реальний API-ключ та ідентифікатор для облікового запису AWS

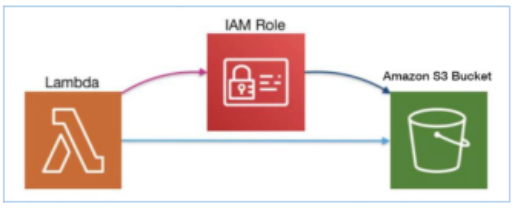


Рис.6. Приклад розподілення ролей в сучасній хмарній інфраструктурі без необхідності викриття облікових даних

ДОСЛІДЖЕННЯ МЕХАНІЗМІВ ТА ЗАСОБІВ ЗАХИСТУ ХМАРНОЇ ІНФРАСТРУКТУРИ AMAZON AWS ВІД ВЕБ-ЗАГРОЗ

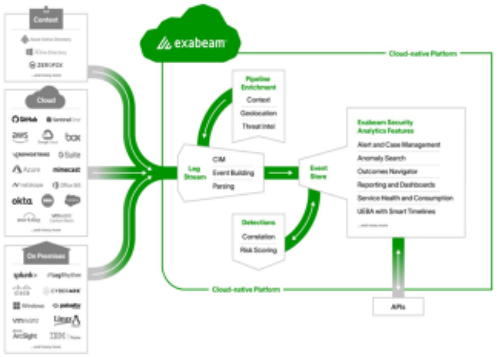


Рис.7. Служби безпеки AWS

- Управління ідентифікацією та доступом AWS.
- Віртуальна приватна хмара AWS.
- Система управління ключами AWS (KMS).
- Shield AWS.
- Брандмауер веб-додатків AWS (WAF).
- AWS CloudTrail.
- AWS CloudWatch.
- AWS Config.
- Артефакт AWS.

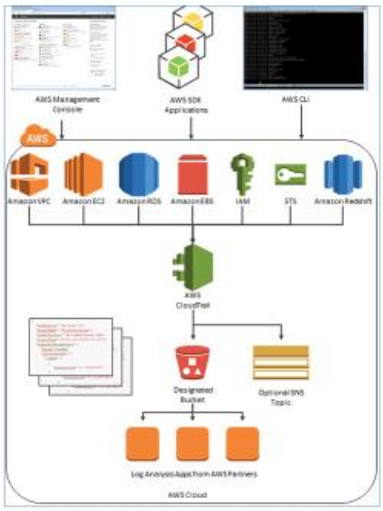


Рис.8. Механізм керування CloudTrail

ДОСЛІДЖЕННЯ ОСОБЛИВОСТЕЙ ВИКОРИСТАННЯ AWS CLOUDTRAIL

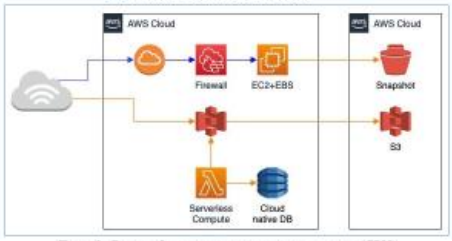


Рис.9. Інтерфейс командного рядка AWS

Рис.10. Стратегії тегування AWS

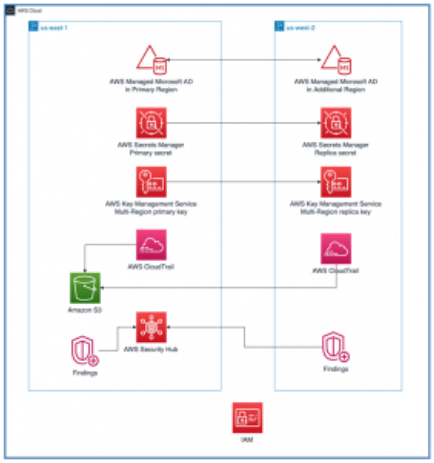


Рис.11. AWS Identity and Access Management

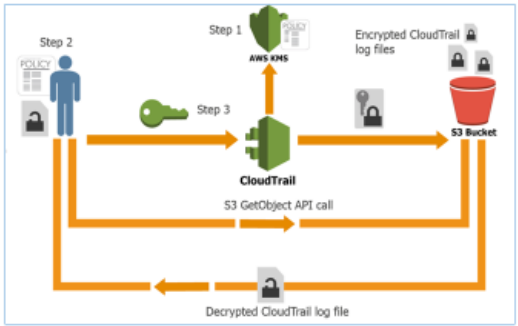


Рис.12. Розшифрування файлів CloudTrail

Рис.13. Вікно вводу реєстраційних даних

Рис.14. Створення користувача через IAM

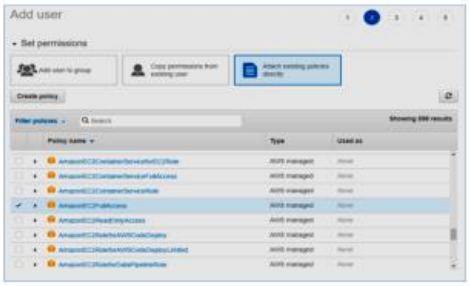


Рис.15. Додавання політики доступу

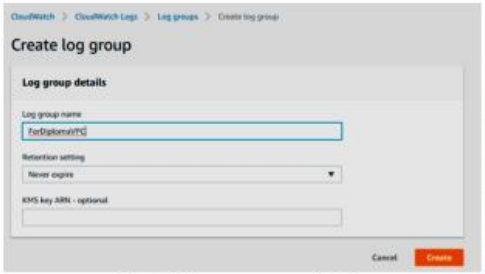


Рис.16. Група для лог-файлів

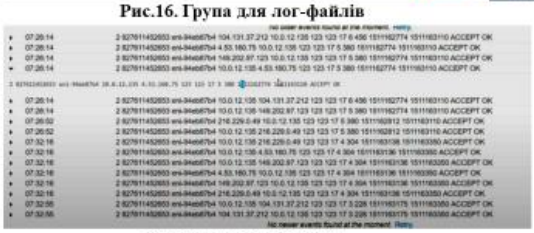


Рис.17. Перегляд логів

Рис.18. Перегляд логів через CloudTrail

РЕКОМЕНДАЦІЇ ЩОДО ЗАБЕЗПЕЧЕННЯ БЕЗПЕКИ В AMAZON WEB SERVICES (AWS) ВІД ВЕБ-ЗАГРОЗ

- ***Оцінка загроз і ризиків.***
- ***Ідентифікація вразливостей.***
- ***Захист ідентифікації і аутентифікації.***
- ***Моніторинг та логування.***
- ***Захист даних.***
- ***Безпека додатків.***
- ***Захист мережі.***
- ***Автоматизація безпеки.***
- ***Навчання та свідомість.***
- ***Запасний план і відновлення.***
- ***Співпраця з AWS.***
- ***Оновлення та аудит.***