УДК 004.5

ЗАБЕЗПЕЧЕННЯ БЕЗПЕКИ СЕРВЕРІВ НА ОСНОВІ AWS: ПРИВАТНІ ТА ПУБЛІЧНІ МЕРЕЖІ

Кхатер Ф. Е.,
здобувач третього (освітньо-наукового) рівня вищої освіти
Приватного вищого навчального закладу
«Міжнародний економіко-гуманітарний університет
імені академіка Степана Дем'янчука» (м. Рівне, Україна)

Науковий керівник: **Джунь Й. В.,**
доктор фізико-математичних наук, професор, завідувач кафедри
математичного моделювання
Приватного вищого навчального закладу
«Міжнародний економіко-гуманітарний університет
імені академіка Степана Дем'янчука» (м. Рівне, Україна)

Анотація. *У статті розглядається питання забезпечення безпеки серверів у хмарних середовищах на прикладі Amazon Web Services (AWS). Дослідження виконано з використанням аналізу доступних інструментів та сервісів AWS з метою ідентифікації оптимальних підходів до захисту серверів через використання приватних та публічних мереж. Основний акцент роботи робиться на аналізі методів та інструментів, доступних на платформі AWS, для забезпечення безпеки серверів та даних у хмарному середовищі.*

Ключові слова: *Хмарні технології, Amazon Web Services (AWS), безпека серверів, приватні мережі, публічні мережі, інформаційна безпека, Virtual Private Cloud (VPC), Availability Zones (AZ).*

Abstract. *This scientific work is dedicated to examining the issue of server security in cloud environments using Amazon Web Services (AWS) as a case study. The research is conducted using an analysis of available AWS tools and services to identify optimal approaches to server protection through the use of private and public networks. The main focus of the work is on analyzing methods and tools available on the AWS platform to ensure server and data security in a cloud environment.*

Keywords: *Cloud technologies, Amazon Web Services (AWS), server security, private networks, public networks, information security, Virtual Private Cloud (VPC), Availability Zones (AZ).*

У сучасній цифровій ері, хмарні технології стали кінцевим рішенням для багатьох компаній, що шукають ефективний та масштабований спосіб забезпечення інфраструктури та послуг. Amazon Web Services (AWS), як провідний постачальник хмарних послуг, забезпечує широкий спектр сервісів,

що включають обчислення, зберігання даних, мережі, аналітику та безліч інших.

Мета статті полягає, щоб разом із зростанням використання хмарних середовищ збільшувався і рівень загроз для безпеки даних та інфраструктури. Забезпечення безпеки серверів у хмарних середовищах, зокрема на платформі AWS, є надзвичайно важливим завданням для забезпечення конфіденційності, цілісності та доступності даних.

У даній роботі проведено аналіз доступних інструментів та сервісів AWS для забезпечення безпеки серверів. Дослідження базувалося на аналізі документації AWS, офіційних джерел інформації та практичного досвіду використання платформи. Використання практичного досвіду сприяло реалізації оптимальних стратегій захисту, враховуючи специфіку конкретних вимог та умов використання. Результати аналізу використовуються для ідентифікації оптимальних підходів до захисту серверів на платформі AWS.

*Використання Приватних Мереж (VPC) та Availability Zones***.** Приватна віртуальна мережа (VPC) у контексті AWS є ключовим елементом для ізоляції серверів та даних від зовнішніх мереж та загроз. Належне налаштування VPC дозволяє контролювати доступ до серверів та здійснювати сегментацію мережі для зменшення ризику витоку інформації.

При створенні VPC важливо враховувати також доступність Availability Zones (AZ). Availability Zones - це фізично відокремлені дата-центри, які забезпечують високий рівень доступності і стійкості до збоїв. Використання кількох Availability Zones дозволяє забезпечити резервне копіювання та відмовостійкість системи. Приведемо приклад створення приватної мережі з використанням Amazon SDK для Node.js та вказанням Availability Zone:

На Рис. 1 вказується Availability Zone для якої створюється VPC та підмережа. Це дозволяє забезпечити географічну розподіленість та стійкість системи до можливих відмов в одному з дата-центрів.

Використання Публічних Мереж (Internet Gateway).

Публічний Інтернет-шлюз (Internet Gateway) забезпечує зв'язок між приватними мережами в AWS та глобальним Інтернетом. Його належна конфігурація та управління дозволяє фільтрувати та контролювати трафік для захисту інфраструктури від зовнішніх атак.

ось приклад створення публічної підмережі (subnet) та підключення до неї інтернет-шлюзу для існуючої мережі (VPC):

На Рис. 2. спочатку створюється публічна підмережа з адресним простором 10.0.1.0/24 та підключається до існуючої мережі (VPC). Потім створюється інтернет-шлюз та приєднується до цієї мережі. Наостанок, встановлюється маршрут для інтернет-шлюзу для направлення всього трафіку з публічної підмережі через цей інтернет-шлюз.

```
const AWS = require('aws-sdk');

// Ініціалізація AWS SDK та об'єкта EC2
const ec2 = new AWS.EC2({ region: 'us-east-1' });

// Функція для створення приватної мережі (VPC)
async function createVPC() {
    try {
        // Створення VPC
        const vpcParams = {
            CidrBlock: '10.0.0.0/16'
        };
        const vpcData = await ec2.createVpc(vpcParams).promise();
        const vpcId = vpcData.Vpc.VpcId;

        // Налаштування власних правил захисту для VPC
        await ec2.modifyVpcAttribute({
            VpcId: vpcId,
            EnableDnsSupport: { Value: true }
        }).promise();
        await ec2.modifyVpcAttribute({
            VpcId: vpcId,
            EnableDnsHostnames: { Value: true }
        }).promise();

        // Створення приватної підмережі
        const subnetParams = {
            VpcId: vpcId,
            CidrBlock: '10.0.1.0/24'
        };
        const subnetData = await ec2.createSubnet(subnetParams).promise();
        const subnetId = subnetData.Subnet.SubnetId;

        console.log("Приватна мережа (VPC) успішно створена з ID:", vpcId);
        console.log("Приватна підмережа успішно створена з ID:", subnetId);
    } catch (err) {
        console.error("Сталася помилка при створенні мережі:", err);
    }
}

// Виклик функції для створення приватної мережі
createVPC();
```

Рис. 1. Приклад створення vpc.

```
const AWS = require('aws-sdk');

// Ініціалізація AWS SDK та об'єкта EC2
const ec2 = new AWS.EC2({ region: 'us-east-1' });

// Функція для створення публічної підмережі та підключення до неї інтернет-шлюзу
async function createPublicSubnetWithInternetGateway(vpcId) {
    try {
        // Створення публічної підмережі
        const subnetParams = {
            VpcId: vpcId,
            CidrBlock: '10.0.1.0/24'
        };
        const subnetData = await ec2.createSubnet(subnetParams).promise();
        const subnetId = subnetData.Subnet.SubnetId;

        console.log("Публічна підмережа успішно створена з ID:", subnetId);

        // Створення інтернет-шлюзу
        const internetGatewayData = await ec2.createInternetGateway().promise();
        const internetGatewayId = internetGatewayData.InternetGateway.InternetGatewayId;

        // Приєднання інтернет-шлюзу до мережі (VPC)
        await ec2.attachInternetGateway({
            InternetGatewayId: internetGatewayId,
            VpcId: vpcId
        }).promise();

        // Створення маршруту для інтернет-шлюзу для публічної підмережі
        await ec2.createRoute({
            DestinationCidrBlock: '0.0.0.0/0',
            GatewayId: internetGatewayId,
            RouteTableId: subnetData.Subnet.RouteTableId
        }).promise();

        console.log("Інтернет-шлюз успішно приєднаний до публічної підмережі");
    } catch (err) {
        console.error("Сталася помилка при створенні підмережі або підключенні інтернет-шлюзу:", err);
    }
}

// Виклик функції для створення публічної підмережі та підключення до неї інтернет-шлюзу
const existingVPCId = 'your_existing_vpc_id'; // Замініть на ID вашої існуючої мережі
createPublicSubnetWithInternetGateway(existingVPCId);
```

Рис. 2. Приклад підключення інтернет шлюзу та створення публічної під мережі.

В результаті отримуємо закриту мережу з доступом в інтернет. Для ефективного забезпечення безпеки серверів на основі AWS рекомендується дотримуватися наступних кращих практик:

– Систематично здійснювати автентифікацію та авторизацію користувачів з використанням міцних методів.

– Проводити регулярне оновлення програмного забезпечення та моніторити активність для виявлення можливих вразливостей.

– Використовувати механізми шифрування для захисту конфіденційної інформації під час передачі та зберігання даних на серверах.

Ефективне забезпечення безпеки серверів на основі AWS вимагає інтеграції приватних та публічних мереж, а також дотримання визначених принципів безпеки та кращих практик управління інформаційною безпекою. На основі проведеного аналізу можна зробити висновок, що для досягнення максимального рівня захищеності в хмарних середовищах AWS необхідно:

– Проактивно налаштовувати мережеві політики: Встановлення правильних мережевих політик, які відображають потреби безпеки вашого додатку або сервісу, дозволяє ефективно управляти доступом до ресурсів та запобігати потенційним загрозам.

– Використовувати механізми ідентифікації та автентифікації: AWS Identity and Access Management (IAM) дозволяє обмежувати доступ до ресурсів мережі лише авторизованим користувачам, що зменшує ризик несанкціонованого доступу.

– Розробляти і реалізовувати стратегії моніторингу та аналізу безпеки: Постійний моніторинг мережевої активності та аналіз безпекових подій дозволяє вчасно виявляти та реагувати на потенційні загрози.

– Створювати та дотримуватися строгих правил захисту даних: Забезпечення конфіденційності та цілісності даних шляхом шифрування, регулярного резервного копіювання та застосування правильних методів управління доступом.

– Постійно оновлювати та підтримувати систему безпеки: Проведення регулярних аудитів безпеки, виявлення та усунення слабких місць, а також оновлення заходів захисту для врахування останніх загроз і тенденцій у сфері кібербезпеки.

Використання зазначених засобів та методів дозволяє забезпечити високий рівень захищеності та надійності інфраструктури в хмарних середовищах AWS. Наукові дослідження та практичний досвід підтверджують, що використання інтегрованих рішень з безпеки дозволяє компаніям ефективно захищати свої дані та інфраструктуру в хмарних середовищах, забезпечуючи високий рівень захисту та відповідність вимогам безпеки даних і регулятивних стандартів.

ЛІТЕРАТУРА

1. Amazon Web Services. (2024). AWS Documentation. Retrieved from https://docs.aws.amazon.com/. **2.** Smith, J. (2023). Securing Your AWS Environment: Best Practices. AWS Blog. Retrieved from https://aws.amazon.com/blogs/security/. **3.** Johnson, A. (2022). AWS Security Best Practices: A Practical Guide. O'Reilly Media. **4.** White, B. (2023). Building Secure and Reliable AWS Architectures. AWS Whitepaper. Retrieved from https://d1.awsstatic.com/whitepapers/architecture/AWS-Security-Whitepaper.pdf.
5. Clark, C. (2024). Advanced Network Security on AWS. IEEE Transactions on Cloud Computing, 12(3), 45-57**. 6.** Kim, D. H., & Lee, S. (2023). Enhancing Cloud Security with AWS Infrastructure: A Case Study. Journal of Cloud Computing, 8(2), 112-125. **7.** Garcia, M. (2022). Understanding Virtual Private Clouds in AWS: A Comprehensive Guide. Springer. **8.** Patel, R. (2023). Internet Gateway Configuration for Secure AWS Deployment. International Journal of Network Security, 10(4), 89-102. **9.** Amazon Web Services. (2023). AWS Security Hub Documentation. Retrieved from https://docs.aws.amazon.com/securityhub/. **10.** Davis, K. (2024). Emerging Threats in Cloud Computing: An Overview. Cloud Security Alliance, 16(1), 23-36.

УДК 37.001.76:81’233

ДО ПРОБЛЕМИ ВПЛИВУ СУЧАСНИХ ТЕХНОЛОГІЙ НА ОСВІТНІЙ ПРОЦЕС

Лясковська Є. О.,
здобувачка першого (бакалаврського) рівня вищої освіти
Приватного вищого навчального закладу
«Міжнародний економіко-гуманітарний університет
імені академіка Степана Дем’янчука» (м. Рівне, Україна)

Науковий керівник: **Смерчко А. А.,**
кандидат філологічних наук, доцент
доцент кафедри іноземних мов
Приватного вищого навчального закладу
«Міжнародний економіко-гуманітарний університет
імені академіка Степана Дем’янчука» (м. Рівне, Україна)

Анотація. *У статті здійснено спробу наукової розвідки актуальної проблеми сьогодення – зв’язку технологій і освіти. Підкреслюється їх взаємна залежність та вплив історичних технологічних проривів на розвиток освіти. Зазначається роль сучасних технологій у сфері освіти, зокрема EdTech-проєктів та штучного інтелекту, що модернізують сучасну освіту і підвищують її якість через інтеграцію інноваційних підходів у навчальний процес. Відзначаєть вплив пандемії на темпи*