

Міністерство освіти і науки України
Державний університет «Житомирська політехніка»
Факультет інформаційно-комп'ютерних технологій
Кафедра комп'ютерної інженерії та кібербезпеки

ПОЯСНЮВАЛЬНА ЗАПИСКА

до випускної кваліфікаційної роботи бакалавра
на тему: «Проект захищеної хмарної інфраструктури Microsoft Azure для роз-
гортання банківського вебдодатку»

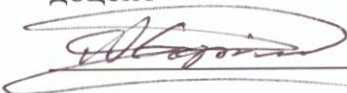
Виконала: студентка 4-го курсу групи КБ-22-1
спеціальності 125 «Кібербезпека»

 Марія КОРОТУН

Керівник: старший викладач кафедри комп'ю-
терної інженерії та кібербезпеки

 Вадим МИКОЛАЙЧУК

Рецензент: завідувач кафедри КІ та КБ, к.т.н.,
доцент

 Андрій ЄФІМЕНКО

Житомир - 2026

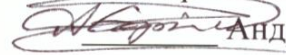
Державний університет «Житомирська політехніка»
Факультет інформаційно-комп'ютерних технологій
Кафедра комп'ютерної інженерії та кібербезпеки
Спеціальність 125 «Кібербезпека»
Освітня програма «Кібербезпека»

ЗАТВЕРДЖУЮ

Завідувач

кафедри комп'ютерної

інженерії та кібербезпеки

 Андрій Єфіменко

02 лютого 2026 р.

ЗАВДАННЯ

на випускню кваліфікаційну роботу бакалавра

Студент: Коротун Марії Олександрівни

Тема роботи: «Проект захищеної хмарної інфраструктури Microsoft Azure для розгортання банківського вебдодатку»

затверджена Наказом університету від 02 лютого 2026 р. № 58/с

Термін здачі студентом закінченої роботи 17 червня 2026 р.

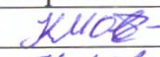
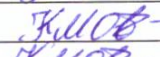
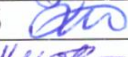
Вихідні дані роботи (зазначається предмет і об'єкт дослідження):

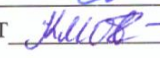
Об'єктом дослідження є процес проектування, розгортання та забезпечення безпеки хмарної інфраструктури в Microsoft Azure для розміщення вебдодатків на базі архітектури Spring Boot та ОС Linux. Предметом дослідження є методи, архітектурні патерни, мережні та безпекові сервіси Azure, а також інструменти комплаєнсу, моніторингу та захисту, що забезпечують високу доступність, захищеність від атак OWASP Top 10 та відповідність нормативним стандартам хмарної інфраструктури.

Обладнання та програмне забезпечення:

1. Azure CLI 2.86.0
2. IntelliJ IDEA 2025.2.5

Консультанти з випускної кваліфікаційної роботи із зазначенням розділів, що їх стосуються

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
1	Головня О.С.	14.03.2026 	14.03.2026 
2	Головня О.С.	07.04.2026 	07.04.2026 
3	Головня О.С.	27.05.2026 	27.05.2026 

Студент  Марія КОРОТУН

Керівник  Вадим МИКОЛАЙЧУК

Календарний план

№ з/п	Назви етапів випускної кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1	Постанова завдання	02.02.2026	Виконано
2	Аналіз отриманого завдання та уточнення вимог до проєкту	20.02.2026	Виконано
3	Аналіз літературних та нормативних джерел з питань хмарної безпеки	06.03.2026	Виконано
4	Аналіз об'єкта дослідження	20.03.2026	Виконано
5	Постановка мети, задач та обґрунтування теми дослідження	03.04.2026	Виконано
6	Проектування архітектурної безпеки хмарної інфраструктури	24.04.2026	Виконано
7	Розробка мережної архітектури та схеми адресації віртуальної мережі	08.05.2026	Виконано
8	Реалізація механізмів захисту вхідного та внутрішнього трафіку	15.05.2026	Виконано
9	Налаштування засобів захисту вебдодатку від атак прикладного рівня	22.05.2026	Виконано
10	Організація безпечного доступу до ресурсів та захисту внутрішніх сервісів	29.05.2026	Виконано
11	Аналіз результатів та формування висновків	02.06.2026	Виконано
12	Оформлення пояснювальної записки та графічних матеріалів	05.06.2026	Виконано
13	Захист	17.06.2026	Виконано

Студент  Марія КОРОТУН

Керівник  Вадим МИКОЛАЙЧУК

АНОТАЦІЯ

Кваліфікаційна робота присвячена проектуванню захищеної хмарної інфраструктури для вебзастосунку в Microsoft Azure. Робота складається зі вступу, трьох розділів, висновків, списку використаних джерел та додатків.

Пояснювальна записка містить 98 сторінок, 84 ілюстрацій та 3 таблиць. Список використаних джерел налічує 31 найменування та розміщений на 3 сторінках.

Метою роботи є проектування, розгортання та обґрунтування захищеної хмарної інфраструктури для вебзастосунку Spring Boot, що функціонує на віртуальній машині Linux під керуванням Ubuntu Server 24.04 у середовищі Microsoft Azure, відповідно до принципів Well-Architected Framework та найкращих практик хмарної безпеки.

Проаналізовано потенційні загрози для вебзастосунку та інфраструктури, зокрема атаки з переліку OWASP Top 10, мережні загрози та ризики несанкціонованого доступу. Розгорнута інфраструктура включає віртуальну мережу з кастомною схемою адресації, private endpoints, network security groups, Azure DNS та вебсервер Nginx, що виконує роль зворотного проксі-сервера. Маршрутизацію вхідного трафіку реалізовано за допомогою Azure Application Gateway з активованим міжмережних екраном вебзастосунків для захисту від вебзагроз та DDoS-атак.

Для основних сервісів було частково застосовано рекомендації Azure Well-Architected Framework і типи політик Azure Policy, а також проаналізовано відповідність стандартам ISO та NIST. Інфраструктуру посилено інструментами Defender for Cloud, а також оцінено доцільність використання Azure Sentinel як SIEM-рішення. Для віртуальної машини описано операційні сервіси, зокрема керування оновленнями, відстеження змін та політику резервного копіювання.

Ключові слова: ХМАРНІ ТЕХНОЛОГІЇ, MICROSOFT AZURE, ЗАХИЩЕНА ІНФРАСТРУКТУРА, SPRING BOOT, APPLICATION GATEWAY, WEB APPLICATION FIREWALL, MANAGED IDENTITY, PRIVATE ENDPOINT, DEFENDER FOR CLOUD, AZURE SQL, MICROSOFT SENTINEL.

					ККІКБ.КР.Б – 125 – 06 – ПЗ				
Змн.	Арк.	№ докум.	Підпис	Дата	Проект захищеної хмарної інфраструктури Microsoft Azure для розгортання банківського вебдодатку	Лім.	Арк.	Акрушів	
Розроб.		Коротун М.О.		16.06.26					
Керівник		Миколайчук В.В.		16.06.26			5	98	
Реценз.		Єфіменко А.А.		16.06.26		ФІКТ, гр. КБ-22-1			
Н. Контр.		Миколайчук В.В.		16.06.26					
Зав.каф.		Єфіменко А.А.		16.06.26					

ANNOTATION

The qualification work is dedicated to designing a secure cloud infrastructure for a web application in Microsoft Azure. The work consists of an introduction, three chapters, conclusions, a list of references, and appendices. The explanatory note contains 98 pages, 84 illustrations, and 3 tables.

The aim of the work is to design, deploy, and substantiate a secure cloud infrastructure for a Spring Boot web application functioning on a Linux virtual machine running Ubuntu Server 24.04 within the Microsoft Azure environment, in accordance with the principles of the Well-Architected Framework and best cloud security practices.

Potential threats to the web application and infrastructure, including OWASP Top 10 attacks, network threats, and unauthorized access risks, are analyzed. The deployed infrastructure includes a virtual network with a custom addressing scheme, private endpoints, network security groups, Azure DNS, and a Nginx web server acting as a reverse proxy. Inbound traffic routing is implemented via Azure Application Gateway with an activated Web Application Firewall to protect against web threats and DDoS attacks.

For the core services, the Azure Well-Architected Framework recommendations and Azure Policy types are applied, and compliance with ISO and NIST standards is analyzed. The infrastructure is reinforced with Defender for Cloud tools, and the feasibility of using Azure Sentinel as a SIEM solution is evaluated. For the virtual machine, operational services are described, including update management, change tracking, and backup policy.

Keywords: CLOUD TECHNOLOGIES, MICROSOFT AZURE, SECURE INFRASTRUCTURE, SPRING BOOT, APPLICATION GATEWAY, WEB APPLICATION FIREWALL, MANAGED IDENTITY, PRIVATE ENDPOINT, DEFENDER FOR CLOUD, AZURE SQL, MICROSOFT SENTINEL.

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Ликолайчук В.В.				
Змн.	Арк.	№ докум.	Підпис	Дата		5

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ.....	7
ВСТУП.....	8
РОЗДІЛ 1. АНАЛІЗ ПРИНЦИПІВ ПОБУДОВИ БЕЗПЕЧНОЇ ХМАРНОЇ ІНФРАСТРУКТУРИ В MICROSOFT AZURE ДЛЯ РОЗМІЩЕННЯ ВЕБДОДАТКУ	10
1.1. Огляд хмарних провайдерів та архітектури Microsoft Azure	10
1.2. Обґрунтування вибору хмарної платформи Microsoft Azure	14
1.3. Аналіз хмарних моделей.....	19
1.4. Методи та засоби забезпечення безпеки хмарної інфраструктури Microsoft Azure	23
1.5. Ключові компоненти для інфраструктури вебдодатку	30
РОЗДІЛ 2. ПРОЄКТУВАННЯ БЕЗПЕЧНОЇ ХМАРНОЇ ІНФРАСТРУКТУРИ ТА АРХІТЕКТУРИ SPRING BOOT ДОДАТКУ	38
2.1. Проєктування архітектури інфраструктури з забезпеченням вимог безпеки	38
2.2. Проєктування безпечної мережної інфраструктури та сервісів Azure для вебдодатку	40
2.3. Проєктна частина Java-додатку	48
РОЗДІЛ 3. ПРАКТИЧНА РЕАЛІЗАЦІЯ ТА РОЗГОРТАННЯ БЕЗПЕЧНОГО ВЕБДОДАТКУ В ХМАРНОМУ СЕРЕДОВИЩІ AZURE	64
3.1. Підготовка та конфігурування хмарного середовища	64
3.2. Налаштування безпечного доступу та ідентифікації	68
3.3. Процес розгортання та експлуатації	76
3.4. Верифікація безпеки та моніторинг стану системи.....	83
ВИСНОВКИ.....	91
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	93
ДОДАТКИ.....	96

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

API – Application Programming Interface
CIS – Center for Internet Security
CRS – Core Rule Set
CSPM – Cloud Security Posture Management
CWP – Cloud Workload Protection
DCR – Data Collection Rules
DDoS – Distributed Denial of Service
DNS – Domain Name System
HTTP – Hypertext Transfer Protocol
HTTPS – Hypertext Transfer Protocol Secure
IaaS – Infrastructure as a Service
IAM – Identity and Access Management
IP – Internet Protocol
ISMS – Information Security Management System
ISO – International Organization for Standardization
JVM – Java Virtual Machine
NSG – Network Security Group
OWASP – Open Web Application Security Project
PaaS – Platform as a Service
RBAC – Role-Based Access Control
SaaS – Software as a Service
SIEM – Security Information and Event Management
SOAR – Security Orchestration, Automation, and Response
SOC – Security Operations Center
SSL – Secure Sockets Layer
TLS – Transport Layer Security
VM – Virtual Machine
VNet – Virtual Network
VPN – Virtual Private Network
WAF – Web Application Firewall
WAF – Well-Architected Framework

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				7
Змн.	Арк.	№ докум.	Підпис	Дата		

ВСТУП

Актуальність теми. Стрімкий розвиток хмарних технологій у сучасному цифровому середовищі радикально змінює підходи до розробки, розгортання та експлуатації вебдодатків. Організації все частіше мігрують до хмарних платформ, щоб забезпечити високу гнучкість, автоматичне масштабування, зниження капітальних витрат та оперативне впровадження інновацій. Водночас перехід до хмари супроводжується значним зростанням кіберзагроз: атаки на вебдодатки, витоки конфіденційних даних та DDoS-атаки стають дедалі складнішими та частішими. У таких умовах створення захищеної інфраструктури перетворюється на критичну вимогу.

У сучасному світі хмарні технології є основою цифрової трансформації бізнесу та державних установ. За даними аналітичних звітів, значна частина організацій вже використовує хмарні сервіси для розміщення критичних вебдодатків. Проте зростання кількості та складності кібератак, зокрема тих, що входять до OWASP Top 10, а також масові DDoS-атаки та спроби компрометації облікових даних, підкреслюють необхідність ретельного проєктування захищених архітектур. Неправильна конфігурація хмарних ресурсів часто стає основною причиною інцидентів безпеки. Тому розробка та впровадження безпечної хмарної інфраструктури з використанням таких сервісів, як Private Endpoints, Managed Identities та Web Application Firewall, є надзвичайно актуальним завданням, що має як теоретичне, так і практичне значення для забезпечення конфіденційності, цілісності та безпечного доступу до інформації.

Метою кваліфікаційної роботи є розробка, обґрунтування та практична реалізація проєкту безпечної хмарної інфраструктури для вебдодатка, створеного на базі Spring Boot, з розгортанням на віртуальній машині Ubuntu Server 24.04 та використанням мережних сервісів, механізмів захисту та управління використаними у роботі сервісами Microsoft Azure відповідно до рекомендацій Well-Architected Framework.

Об'єкт дослідження є процес реалізації проєкту хмарної інфраструктури вебо-рієнтованого програмного рішення, яке розгорнуто в середовищі Microsoft Azure.

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				
Змн.	Арк.	№ докум.	Підпис	Дата		8

Предмет дослідження - методи, інструменти та архітектурні принципи побудови захищеної хмарної інфраструктури для вебдодатка, що охоплюють Virtual Networks, Network Security Groups, Application Gateway, сервіси автентифікації та авторизації на базі Entra ID, Managed Identities, Private Endpoints корпоративні політики безпеки, моніторинг загроз та операційні сервіси Azure.

У процесі виконання кваліфікаційної роботи було розроблено проєкт захищеної хмарної інфраструктури Microsoft Azure для розгортання банківського вебдодатку. Створене комплексне архітектурне рішення базується на системному аналізі сучасних кіберзагроз, вимогах міжнародних стандартів безпеки та принципах методології Azure Well-Architected Framework.

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				9
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 1. АНАЛІЗ ПРИНЦИПІВ ПОБУДОВИ БЕЗПЕЧНОЇ ХМАРНОЇ ІНФРА-СТРУКТУРИ В MICROSOFT AZURE ДЛЯ РОЗМІЩЕННЯ ВЕБДОДАТКУ

1.1. Огляд хмарних провайдерів та архітектури Microsoft Azure

1.1.1. Основні глобальні хмарні провайдери

Сучасний ринок хмарних обчислень характеризується значною конкуренцією, де основними глобальними гравцями є Amazon Web Services, Google Cloud Platform та Microsoft Azure [1]. Кожен із цих провайдерів пропонує широкий спектр послуг, починаючи від базових обчислювальних ресурсів і закінчуючи рішеннями для штучного інтелекту та аналітики великих даних. Проте при виборі платформи для розгортання вебдодатку важливими критеріями є не тільки технічні характеристики сервісів, а й інтеграція з існуючими корпоративними системами, безпека, надійність та зручність управління.

Amazon Web Services є піонером хмарних технологій і найбільшим гравцем на ринку. AWS пропонує найширший спектр сервісів — від віртуальних машин EC2 та баз даних RDS і DynamoDB до спеціалізованих сервісів для машинного навчання, аналітики та Інтернету речей [2]. Серед ключових переваг AWS — висока масштабованість, велика кількість регіонів і дата-центрів, а також набір інструментів для безпеки та управління. Водночас складність конфігурації та управління великою кількістю сервісів може створювати бар'єри для новачків та підвищувати експлуатаційні витрати. Вартість підтримки масштабних рішень на AWS часто перевищує бюджет невеликих і середніх підприємств, що робить платформу менш привабливою для організацій із обмеженими ресурсами.

Google Cloud Platform вирізняється сильним акцентом на аналітику, обробку великих даних та рішення для машинного навчання. Інструменти GCP, такі як BigQuery, TensorFlow і Dataflow, дозволяють ефективно працювати з масивами даних і виконувати складні аналітичні завдання. Це робить платформу особливо цінною для компаній, які займаються науковими дослідженнями, великими даними та інноваційними проектами. Проте кількість регіонів і дата-центрів у GCP обмежена

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				
Змн.	Арк.	№ докум.	Підпис	Дата		10

порівняно з AWS і Azure, що може створювати географічні обмеження для організацій, які потребують локалізації даних або низьких затримок доступу для глобальних користувачів.

Microsoft Azure поєднує в собі гнучкість, корпоративну інтеграцію та високий рівень безпеки [3]. Для організацій, що використовують Windows Server, Active Directory, SQL Server та Microsoft 365, Azure стає природним вибором, оскільки забезпечує безшовну інтеграцію з уже наявними системами [1]. Azure підтримує різні моделі розгортання - IaaS, PaaS та SaaS [4] - і дозволяє запускати як Windows, так і Linux-додатки, що значно підвищує гнучкість при реалізації різноманітних проєктів. Крім того, платформа активно підтримує методологію Well-Architected Framework, що дає змогу будувати безпечну, надійну та масштабовану інфраструктуру з урахуванням кращих практик щодо безпеки, відмовостійкості, оптимізації витрат та експлуатаційної ефективності.

Для наочності порівняння ключових характеристик основних хмарних провайдерів можна представити у вигляді таблиці (табл. 1.1):

Таблиця 1.1

Порівняння провідних хмарних платформ (станом на 2026 рік)

Параметр	AWS	GCP	Microsoft Azure
Глобальна присутність	>33 регіонів, >105 зон доступності	40 регіонів, 121 зона доступності	>60 регіонів, >300 фізичних дата-центрів
Основний фокус	Масштабованість, IaaS, найширший спектр інфраструктурних сервісів	Аналітика великих даних, штучний інтелект, контейнеризація	Глибока інтеграція з корпоративними продуктами (MS Ecosystem, AD, Windows Server)
Підтримка ОС (на рівні віртуальних машин)	Повна підтримка дистрибутивів Linux (зокрема Amazon Linux, Ubuntu) та Windows Server	Оптимізація під дистрибутиви Linux, підтримка Windows Server	Рівноцінна підтримка Linux (Ubuntu, RedHat) та Windows Server; наявність ексклюзивних переваг (Azure Hybrid Benefit)

Інструменти безпеки	AWS IAM, AWS Shield, Amazon GuardDuty, AWS WAF, AWS Key Management Service	Identity-Aware Proxy, Google Cloud Armor, Chronicle, Cloud KMS	Microsoft Entra ID, Microsoft Defender for Cloud, Azure WAF, Azure Policy, Microsoft Sentinel
---------------------	--	--	---

Продовження таблиці 1.1

Параметр	AWS	GCP	Microsoft Azure
Цінова політика	Оплата за фактом (Pay-as-you-go); значні знижки за резервування інстансів (Reserved Instances)	Гнучка система знижок за тривале використання без жорстких зобов'язань	Економічно вигідна для Enterprise завдяки гібридним ліцензіям та Azure Hybrid Benefit (знижки до 85% на Windows Server)

Хоча AWS і GCP пропонують потужні інструменти та сервіси, Microsoft Azure забезпечує більш зручну інтеграцію з корпоративним середовищем, розширені можливості для управління безпекою та відповідність стандартам сучасних організацій, що робить її ваговою платформою для реалізації безпечної хмарної інфраструктури вебдодатку.

1.1.2 Архітектурні компоненти Microsoft Azure

Архітектура Microsoft Azure спроектована з урахуванням принципів масштабованості, безпеки та високої доступності, що робить платформу універсальним середовищем для розгортання сучасних вебдодатків. Відмінною особливістю Azure є модульний підхід: ресурси організовані таким чином, щоб кожен компонент міг виконувати свої функції самостійно, при цьому інтегруючись у єдину екосистему. Це дозволяє забезпечувати високу надійність, гнучкість та контроль безпеки на всіх рівнях інфраструктури [13].

Перш за все, обчислювальні сервіси є основою платформи. Віртуальні машини дозволяють запускати додатки з повним контролем операційної системи,

конфігурацій та програмного забезпечення. Водночас Azure підтримує контейнеризацію та безсерверні сервіси, такі як Azure Kubernetes Service та Azure Functions, що дозволяє реалізовувати динамічне масштабування та автоматичне управління ресурсами без необхідності вручну керувати інфраструктурою [4]. Завдяки цьому можна адаптувати додатки до змінного навантаження, знижуючи витрати та забезпечуючи ефективність роботи системи.

Другим важливим рівнем є мережні сервіси, які забезпечують безпеку та ізоляцію ресурсів. Virtual Network дозволяє створювати приватні мережі для віртуальних машин, баз даних та сховищ, ізолюючи внутрішній трафік від публічного інтернету [18]. Network Security Groups забезпечує детальний контроль над вхідним та вихідним трафіком, дозволяючи реалізовувати принцип мінімального доступу [6]. Для безпечного адміністративного доступу до віртуальних машин Azure пропонує Bastion, який дозволяє підключатися через веббраузер без відкриття SSH чи RDP-портів, тим самим знижуючи ризик несанкціонованого доступу [5].

Важливим компонентом є Application Gateway з активованим Web Application Firewall, який забезпечує балансування навантаження на рівні 7 моделі OSI та захист від найпоширеніших веб-атак, включно з OWASP Top 10 [17]. Azure також інтегрує захист від DDoS, що дозволяє забезпечити стійкість додатку навіть під час масштабних атак на мережному рівні.

Не менш значущою складовою є сховище даних. Azure надає як реляційні бази даних, наприклад Azure SQL Database [10]. Для підвищення безпеки ці ресурси можуть бути підключені через Private Endpoint, що повністю виключає їх із публічного інтернет-простору. Доступ до даних суворо контролюється за допомогою Managed Identity та Role-Based Access Control. Крім того, комплексні механізми захисту та шифрування забезпечують безпеку інформації на рівні системи керування базами даних [20].

Для управління та моніторингу інфраструктури Azure надає потужні інструменти: Azure Policy дозволяє автоматично забезпечувати відповідність корпоративним стандартам та нормам безпеки, контролювати створення ресурсів та автома-

тично виправляти невідповідності. Defender for Cloud виконує постійний моніторинг безпеки всіх ресурсів, аналізує стан захищеності, надає рекомендації щодо управління вразливостей і інтегрується з Azure Sentinel для SIEM, аналізу та автоматизованої реакції на загрози [16]. Для збору, обробки й аналізу метрик та телеметрії використовується Azure Monitor, а детальний аудит дій користувачів та відстеження подій безпеки здійснюється у Log Analytics Workspace за допомогою аналітичних запитів.

1.2. Обґрунтування вибору хмарної платформи Microsoft Azure

1.2.1. Переваги використання Azure для розміщення вебдодатків

У сучасних умовах стрімкого розвитку інформаційних технологій та цифровізації бізнес-процесів питання вибору хмарної платформи для розміщення вебдодатків є критично важливим. Хмарні технології забезпечують не лише гнучкість та масштабованість, а й високий рівень безпеки, надійності та доступності додатків для користувачів з різних регіонів. Вибір Microsoft Azure як платформи для хостингу вебдодатків зумолений низкою стратегічних та технічних переваг, що роблять цю платформу оптимальним рішенням для корпоративного середовища та організацій, що прагнуть до дотримання сучасних стандартів безпеки [1].

Однією з ключових переваг Azure є його тісна інтеграція з корпоративним програмним забезпеченням Microsoft. Для організацій, які вже використовують Windows Server, Active Directory, Microsoft 365 або SQL Server, Azure дозволяє створювати безшовне середовище, де всі сервіси взаємодіють між собою з мінімальною потребою у додатковій конфігурації. Така інтеграція значно спрощує управління користувачами та ролями, дозволяє централізовано здійснювати автентифікацію та авторизацію через Entra ID, а також забезпечує безпечне підключення вебдодатків до баз даних і сховищ за допомогою Managed Identity, що усуває необхідність зберігання секретів у коді додатку [21].

Другою важливою перевагою є широкий спектр обчислювальних і мережних сервісів, що дозволяє реалізовувати архітектури будь-якої складності. Azure підтримує як класичні віртуальні машини, що забезпечують повний контроль над операційною системою, так і контейнеризовані та безсерверні сервіси, які автоматично

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				
Змн.	Арк.	№ докум.	Підпис	Дата		14

масштабуються відповідно до навантаження [4]. Таке поєднання дозволяє гнучко адаптувати вебдодаток до змінюваного трафіку, забезпечуючи оптимальне використання ресурсів та мінімізацію витрат. Особливо важливою є наявність Application Gateway з Web Application Firewall, що забезпечує багаторівневий захист вебдодатків від атак на прикладному рівні, включно з OWASP Top 10 та захистом від DDoS-атак [17, 28].

Ще однією суттєвою перевагою є можливість побудови приватних мереж і безпечного підключення ресурсів. Azure дозволяє створювати Virtual Network із приватними підмережами для вебдодатків, баз даних та сховищ [18]. Network Security Groups забезпечує детальний контроль над вхідним і вихідним трафіком, а Private Endpoints дозволяють підключати ресурси без відкриття публічних IP-адрес [23, 24]. Такий підхід забезпечує високий рівень захисту даних, оскільки всі критичні сервіси залишаються недоступними для зовнішнього інтернету, що відповідає кращим практикам безпечної архітектури.

Не менш важливим аспектом є масштабованість та відмовостійкість платформи. Azure дозволяє розгортати ресурси у різних регіонах світу, забезпечуючи низьку затримку доступу до додатків для глобальної аудиторії [13]. Механізми автоматичного масштабування дозволяють додатку адаптуватися до змінного трафіку без ручного втручання адміністратора. Крім того, Azure підтримує високий рівень відмовостійкості завдяки геореплікації баз даних, резервному копіюванню та балансуванню навантаження, що забезпечує безперервність бізнес-процесів навіть у разі виходу з ладу окремих компонентів [2].

Ще одним стратегічним аргументом на користь Azure є економічна ефективність та гнучка цінова політика. Платформа підтримує різні моделі оплати, включаючи постійне резервування ресурсів для довгострокових проєктів та оплату лише за фактично використані обчислювальні потужності у випадку змінного навантаження. Це дозволяє організаціям оптимізувати витрати на інфраструктуру та забезпечувати ефективне використання ресурсів без надлишкових затрат. Детальний аналіз та порівняння хмарних платформ наведено у Додатку А.

Завдяки вищезгаданим особливостям, використання Microsoft Azure для розміщення вебдодатків дозволяє організаціям реалізовувати сучасні підходи до побудови безпечної, масштабованої та надійної інфраструктури. Платформа забезпечує високий рівень інтеграції, централізоване управління доступом та безпекою, гнучкість у масштабуванні та оптимізацію витрат, що робить її оптимальним вибором для корпоративного середовища та будь-яких проєктів, де критичною є безпека, продуктивність та надійність роботи додатків.

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				16
Змн.	Арк.	№ докум.	Підпис	Дата		

1.2.2. Архітектурні та безпекові особливості платформи

Вибір хмарної платформи для розміщення вебдодатків обумовлюється не лише її функціональними можливостями, а й архітектурною структурою та рівнем безпеки, який вона забезпечує. Microsoft Azure є однією з небагатьох платформ, яка надає комплексну екосистему для побудови безпечної, масштабованої та керованої інфраструктури, що відповідає сучасним корпоративним стандартам.

Архітектура платформи побудована на принципах модульності, ізоляції ресурсів та багаторівневого захисту, що дозволяє організаціям ефективно управляти додатками та даними без ризику витоку або несанкціонованого доступу [13].

На рівні обчислювальних ресурсів Azure надає широкий вибір сервісів. Віртуальні машини дозволяють користувачам запускати додатки з повним контролем над операційною системою та конфігураціями, що особливо важливо для критичних корпоративних сервісів або додатків із специфічними налаштуваннями. Одночас платформа підтримує контейнеризовані та безсервісні сервіси, такі як Azure Kubernetes Service та Azure Functions, які забезпечують автоматичне масштабування, оптимізацію ресурсів та зниження операційних витрат [4]. Це дозволяє вебдодаткам динамічно адаптуватися до змінного навантаження, не вимагаючи ручного втручання адміністратора.

Особливу увагу Azure приділяє мережній безпеці та ізоляції ресурсів. Virtual Network дозволяє створювати приватні підмережі для вебдодатків, баз даних, ізолюючи критичні сервіси від публічного інтернету [18]. Network Security Groups забезпечують контроль доступу на рівні IP-адрес і портів, дозволяючи визначати правила для вхідного та вихідного трафіку [24]. Для безпечного підключення адміністраторів та операторів використовується Azure Bastion, який дозволяє доступ через браузер без відкриття SSH чи RDP портів, що суттєво знижує ризик несанкціонованого доступу.

Ще другим важливим компонентом є Application Gateway з інтегрованим WAF, який забезпечує балансування навантаження на рівні 7 OSI-моделі та захист від атак на вебдодатки, включаючи OWASP Top 10 [17]. WAF дозволяє ефективно захищати вебдодатки від SQL Injection, Cross-Site Scripting, Remote File Inclusion та

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				17
Змн.	Арк.	№ докум.	Підпис	Дата		

інших поширених атак. Комбінація Application Gateway і DDoS Protection забезпечує стійкість додатків навіть при високих навантаженнях або спробах зовнішнього втручання [14].

На рівні зберігання даних Azure забезпечує багаторівневий захист. Azure SQL Database підтримує шифрування даних як у спокої, так і під час передачі, а доступ до ресурсів контролюється через Private Endpoints, що виключає доступ із публічного інтернету [23]. Підключення до ресурсів відбувається за допомогою Managed Identity, що усуває необхідність зберігання облікових даних у коді додатків [21]. Крім того, застосування Role-Based Access Control та вбудованих механізмів безпеки дозволяє детально налаштовувати права доступу, забезпечуючи принцип мінімального доступу та підвищуючи безпеку корпоративних даних [20].

Для контролю відповідності корпоративним стандартам та нормативним вимогам Azure надає Azure Policy, що дозволяє централізовано визначати правила створення ресурсів, обмежувати використання небезпечних конфігурацій та автоматично виправляти порушення політик [2]. Defender for Cloud здійснює постійний моніторинг безпеки інфраструктури, виявляє потенційні вразливості та надає рекомендації щодо їх усунення [15]. Інтегрована система Azure Sentinel забезпечує проактивне виявлення загроз, збір та аналіз логів, а також автоматизовану реакцію на інциденти. Для зберігання та обробки великих обсягів логів використовується Log Analytics Workspace, що дозволяє проводити аудит, відстежувати дії користувачів і аналізувати продуктивність системи [22].

Особлива увага приділяється побудові модульної архітектури. Вебдодаток у Azure розбивається на окремі компоненти: фронтенд, бізнес-логіка, база даних. Кожен модуль розміщується в окремій підмережі та захищається власними правилами доступу, що забезпечує ізоляцію компонентів та мінімізацію ризику компрометації всієї системи у випадку атаки на окремий модуль.

Крім того, Azure підтримує автоматичне оновлення та резервне копіювання ресурсів, що забезпечує високу доступність додатків та стійкість до технічних збоїв. Механізми моніторингу продуктивності за допомогою Azure Monitor, журналювання подій та аналітики дозволяють адміністратору оперативно реагувати на

потенційні проблеми, планувати масштабування та забезпечувати безперервність бізнес-процесів [11].

1.3. Аналіз хмарних моделей

1.3.1. Private, Public та Hybrid Cloud

Хмарні моделі обчислень класифікуються за різними критеріями, одним із яких є тип середовища та спосіб управління ресурсами. Традиційно виділяють три основні моделі: Private Cloud, Public Cloud та Hybrid Cloud, що представлено на рисунку 1.1. Кожна з цих моделей має власні особливості, переваги і обмеження, що безпосередньо впливають на архітектуру вебдодатків, рівень безпеки, масштабованість та ефективність управління інфраструктурою [13].

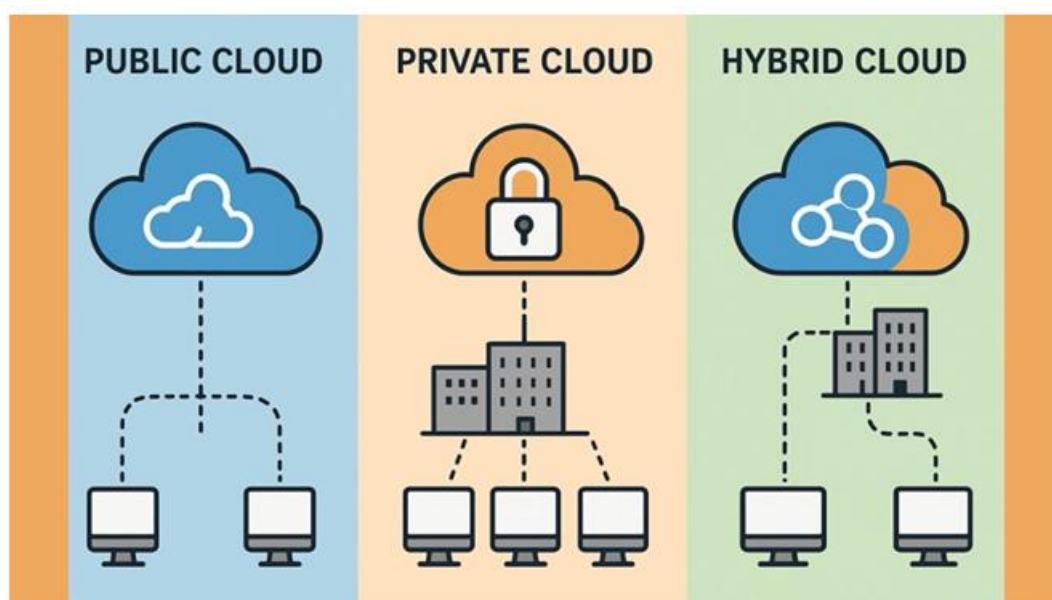


Рисунок 1.1 – Принцип побудови хмарних моделей

Private Cloud передбачає, що всі ресурси виділяються для одного підприємства або організації і знаходяться під повним контролем її ІТ-відділу. Така модель забезпечує високий рівень безпеки та конфіденційності даних, оскільки фізичні та віртуальні ресурси ізольовані від зовнішніх користувачів та інших організацій. *Private Cloud* дозволяє реалізувати індивідуальні політики доступу, контролю та моніторингу, що особливо важливо для організацій із суворими нормативними вимогами, наприклад у фінансовому секторі або державних структурах. З технічної точки зору, приватна хмара надає повний контроль над інфраструктурою, дозволяє

налаштовувати високодоступні кластери, реалізовувати політики резервного копіювання та відмовостійкості, а також впроваджувати індивідуальні протоколи шифрування та автентифікації. Для вебдодатків Private Cloud є оптимальним варіантом у випадках, коли необхідна максимальна конфіденційність, жорсткий контроль за даними та відповідність регуляторним стандартам безпеки [31].

Public Cloud надає ресурси, які керуються і підтримуються стороннім провайдером хмарних послуг. Користувачі орендують обчислювальні потужності, сховища та мережні сервіси за моделлю pay-as-you-go, не беручи на себе управління фізичними серверами та базовим програмним забезпеченням. Публічні хмари, такі як Microsoft Azure, AWS або Google Cloud Platform, пропонують високий рівень масштабованості та швидке розгортання ресурсів, що робить їх особливо пріоритетними для стартапів та компаній із мінливими потребами у навантаженні. Public Cloud дозволяє організаціям зосередитися на розробці та оптимізації вебдодатків, делегуючи провайдеру всі аспекти управління інфраструктурою, включаючи оновлення, резервне копіювання, безпеку та моніторинг. Для вебдодатків Public Cloud підходить тоді, коли пріоритетом є швидке масштабування, доступність для глобальної аудиторії та оптимізація витрат [2]. Узагальнене порівняння Public Cloud та Private Cloud зображено на рисунку 1.2.

Hybrid Cloud поєднує в собі елементи приватної та публічної хмари, надаючи організаціям можливість використовувати обидва типи інфраструктури одночасно. У такій моделі критично важливі дані та внутрішні сервіси розміщуються у приватній хмарі для забезпечення максимального рівня безпеки та контролю, тоді як менш чутливі або динамічно масштабовані ресурси - у публічній хмарі для забезпечення економічності та гнучкості. Гібридна хмара дозволяє реалізовувати стратегії розподілу навантаження, автоматично масштабуючи вебдодаток у публічній хмарі під час пікових навантажень та зберігаючи критичні компоненти у приватному середовищі.

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				20
Змн.	Арк.	№ докум.	Підпис	Дата		



Рисунок 1.2 – Співставлення характеристик Private Cloud та Public Cloud

Особливу увагу слід приділити аспектам безпеки у всіх трьох моделях. У приватній хмарі безпека забезпечується локальним контролем доступу, шифруванням на рівні апаратного та програмного забезпечення та політиками відмовостійкості. У публічній хмарі безпека реалізується через інструменти провайдера, включаючи шифрування даних, контроль доступу, аудит активності та захист від DDoS-атак. Гібридна хмара потребує комплексного підходу до безпеки, що включає захищені канали передачі даних між приватними і публічними компонентами, централізоване управління правами доступу та інтеграцію систем моніторингу безпеки [3].

Для вебдодатків, що розгортаються на Microsoft Azure, ці моделі реалізуються через відповідні сервіси та інструменти: приватна хмара може будуватися на базі виділених Virtual Network та Azure Stack; публічна хмара реалізується через загальнодоступні обчислювальні ресурси, PaaS та безсерверні сервіси; гібридна хмара організовується через інтеграцію приватних мереж із публічними сервісами за допомогою Azure VPN Gateway, ExpressRoute та Private Endpoints.

1.3.2. Моделі надання сервісів: IaaS, PaaS, SaaS

Хмарні обчислення надають організаціям можливість вибору моделей надання сервісів, які визначають рівень контролю над інфраструктурою, гнучкість масштабування та економічну ефективність. Найпоширенішими моделями є Infrastructure as a Service, Platform as a Service та Software as a Service [4].

Кожна з цих моделей має власні характеристики, переваги та обмеження, що безпосередньо впливають на архітектуру вебдодатків, безпеку, управління ресурсами та інтеграцію з іншими сервісами [13].

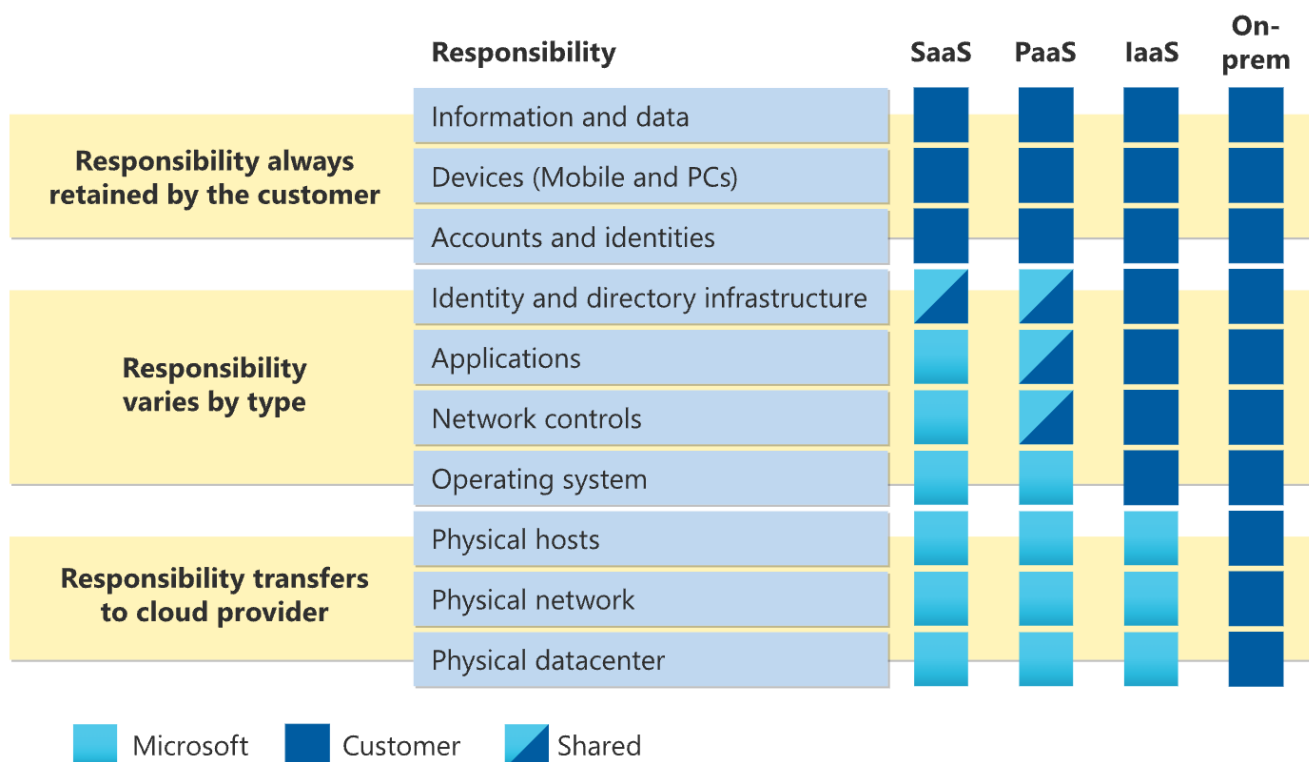


Рисунок 1.3 – Типи хмарних моделей

Infrastructure as a Service забезпечує користувачів віртуалізованими обчислювальними ресурсами, сховищами та мережними компонентами, наданими провайдером хмарних послуг. Користувач самостійно керує операційними системами, встановлює необхідне програмне забезпечення та налаштовує середовище відповідно до потреб додатку. Основною перевагою IaaS є повний контроль над інфраструктурою та можливість реалізації індивідуальних архітектурних рішень, включаючи високі стандарти безпеки та відмовостійкості. Для вебдодатків ця модель оптимальна, коли необхідно забезпечити максимальний контроль над середовищем, базами даних та мережними компонентами.

Platform as a Service пропонує користувачеві вже готову платформу для розробки та запуску додатків без необхідності керувати низькорівневою інфраструктурою. У PaaS організація зосереджується на розробці бізнес-логіки, а провайдер хмари

забезпечує управління операційними системами, патчами, масштабуванням та відмовостійкістю. Це дозволяє прискорити розгортання вебдодатків, автоматично масштабувати ресурси та знизити навантаження на IT-відділ.

У Microsoft Azure PaaS реалізується через сервіси Azure App Service, Azure Functions та Azure Kubernetes Service, що підтримують інтеграцію DevOps-процесів, безперервне оновлення додатків та масштабування відповідно до навантаження [8, 9].

Software as a Service надає кінцевим користувачам готові програмні продукти через інтернет, без необхідності розгортання або адміністрування інфраструктури та платформи. SaaS дозволяє швидко отримати доступ до функціоналу, що надається провайдером, з будь-якого пристрою та місця, зменшуючи витрати на підтримку програмного забезпечення. Для вебдодатків SaaS підходить у випадках, коли потрібне стандартне програмне забезпечення з мінімальною кастомізацією та швидким масштабуванням для глобальної аудиторії.

Для корпоративних вебдодатків часто застосовується комбінований підхід, коли критично важливі компоненти розгортаються на IaaS або PaaS, а менш чутливі сервіси або інтеграційні рішення — через SaaS. Це дозволяє організаціям будувати вебдодатки з оптимальним балансом між контролем, безпекою, масштабованістю та економічною ефективністю. PaaS сервіси автоматично забезпечують оновлення, відновлення після збоїв та захист від DDoS-атак, а SaaS надає готові механізми автентифікації, шифрування та аудиту користувацьких дій [3].

1.4. Методи та засоби забезпечення безпеки хмарної інфраструктури Microsoft Azure

1.4.1. Принципи Azure Well-Architected Framework

Проектування безпечної, надійної та ефективної хмарної інфраструктури вимагає системного підходу, що базується на перевірених архітектурних принципах і кращих практиках. У середовищі Microsoft Azure таким методологічним підґрунтям є Azure WAF - комплекс рекомендацій і принципів, спрямованих на створення якісних хмарних рішень, які відповідають сучасним вимогам безпеки, масштабова-

ності, продуктивності та керованості [2, 13]. Даний фреймворк є універсальним інструментом для оцінки та вдосконалення архітектури вебдодатків, розгорнутих у хмарі. Приклад архітектури побудованої за Well-Architected Framework представлено на рисунку 1.4.

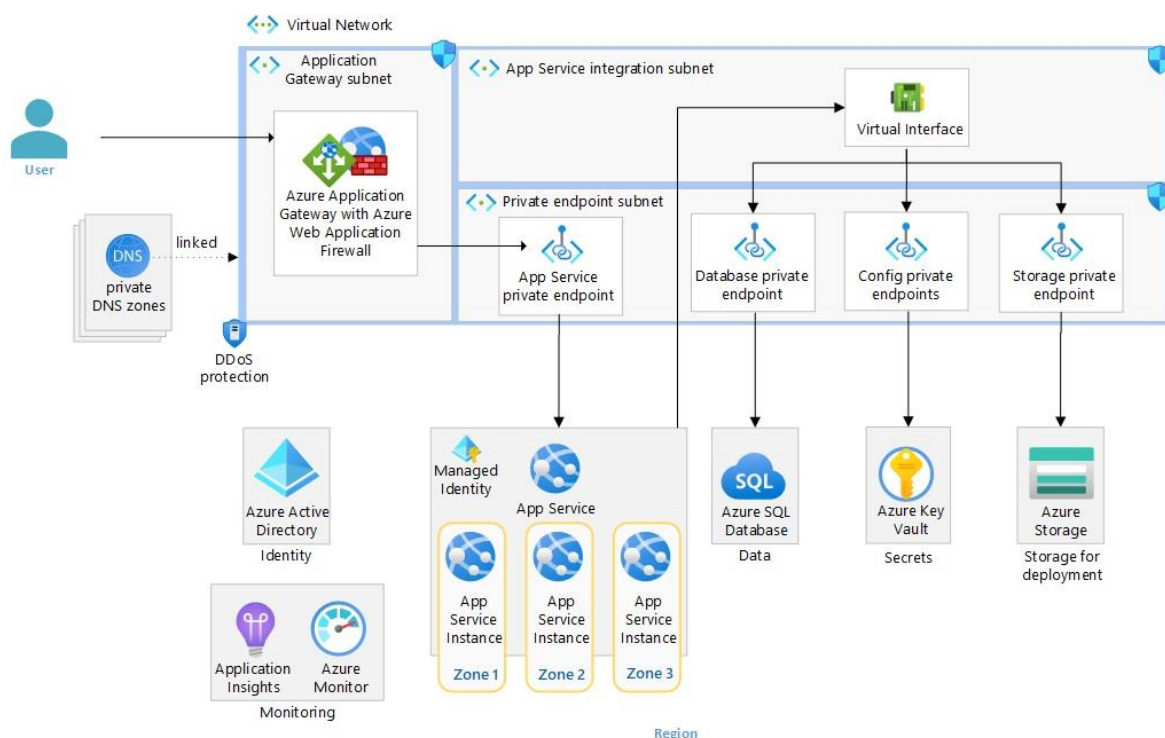


Рисунок 1.4 – Приклад діаграми Azure Well-Architected Framework

Azure Well-Architected Framework ґрунтується на п'яти ключових стовпах: Security, Reliability, Performance Efficiency, Cost Optimization та Operational Excellence [2]. Кожен із цих стовпів охоплює окремий аспект архітектурної якості та разом вони формують цілісну модель проектування хмарної інфраструктури. Застосування принципів WAF дозволяє мінімізувати ризики, пов'язані з безпекою та доступністю сервісів, а також забезпечити стабільну роботу вебдодатків у довгостроковій перспективі.

Першим і фундаментальним стовпом Well-Architected Framework є *безпека*. У контексті Azure безпека розглядається як багаторівнева система захисту, що охоплює ідентифікацію та управління доступом, мережну безпеку, захист даних і постійний моніторинг [14]. Для вебдодатків це означає використання керованих ідентифікацій, мінімізацію прав доступу, застосування мережної сегментації та захист прикладного рівня за допомогою Web Application Firewall [17].

Другим важливим стовпом є *надійність*, яка визначає здатність системи стабільно функціонувати та відновлюватися після збоїв. Well-Architected Framework рекомендує проєктувати архітектуру з урахуванням відмовостійкості, резервного копіювання та планів відновлення після аварій. У середовищі Azure це досягається шляхом використання зон доступності, автоматичного масштабування, резервного копіювання віртуальних машин і баз даних, а також моніторингу стану ресурсів. Для вебдодатків критично важливо забезпечити безперервність доступу до сервісів навіть у разі збоїв окремих компонентів.

Третім стовпом Well-Architected Framework є *ефективність продуктивності*. Цей аспект спрямований на оптимальне використання ресурсів з метою досягнення необхідної продуктивності при мінімальних витратах. Принципи продуктивності передбачають динамічне масштабування, використання керованих сервісів і оптимізацію мережних маршрутів. У випадку вебдодатків, розміщених в Azure, це може включати використання балансувальників навантаження, кешування даних і оптимізацію доступу до баз даних через Private Endpoints [23].

Оптимізація витрат є четвертим стовпом Azure Well-Architected Framework. Хмарна інфраструктура надає гнучкі можливості масштабування, однак без належного контролю це може призвести до неефективного використання ресурсів [4]. Well-Architected Framework рекомендує постійно аналізувати використання ресурсів, застосовувати політики обмеження та використовувати інструменти моніторингу витрат.

Останнім, але не менш важливим стовпом є *операційна досконалість*, яка охоплює процеси управління, моніторингу та вдосконалення системи. Принципи цього стовпа передбачають автоматизацію розгортання, централізований моніторинг, управління змінами та реагування на інциденти. У середовищі Azure це реалізується через використання Azure Monitor, Log Analytics, автоматизованих сценаріїв розгортання та систем сповіщення. Операційна досконалість дозволяє забезпечити стабільну експлуатацію вебдодатків і своєчасне виявлення потенційних проблем.

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				25
Змн.	Арк.	№ докум.	Підпис	Дата		

Важливою перевагою Azure Well-Architected Framework є його тісна інтеграція з інструментами платформи, зокрема Azure Advisor та Defender for Cloud, які дозволяють проводити регулярну оцінку архітектури та отримувати рекомендації щодо її вдосконалення [15]. Це сприяє безперервному підвищенню рівня безпеки, надійності та ефективності інфраструктури.

1.4.2. Підходи до відповідності міжнародним стандартам безпеки

У процесі проектування та експлуатації хмарної інфраструктури особливого значення набуває питання відповідності міжнародним стандартам інформаційної безпеки. Для сучасних вебдодатків, що функціонують у публічному або гібридному хмарному середовищі, дотримання таких стандартів є не лише рекомендацією, а часто й обов'язковою вимогою, зумовленою законодавчими, галузевими або корпоративними нормами. Відповідність стандартам безпеки дозволяє знизити ризики витоку даних, несанкціонованого доступу та порушення цілісності інформації, а також забезпечує довіру з боку користувачів і партнерів.

Microsoft Azure розроблений із урахуванням міжнародно визнаних підходів до інформаційної безпеки та комплаєнсу. Платформа підтримує широкий спектр стандартів і нормативних вимог, серед яких ISO/IEC 27001, ISO/IEC 27017, ISO/IEC 27018, NIST SP 800-53, CIS Benchmarks, а також вимоги, пов'язані з захистом персональних даних [14]. Основою такого підходу є інтеграція механізмів безпеки безпосередньо в архітектуру платформи та надання користувачам інструментів для контролю відповідності на рівні інфраструктури та додатків.

Одним із ключових стандартів у сфері інформаційної безпеки є *ISO/IEC 27001*, який визначає вимоги до ISMS. Azure відповідає цьому стандарту, забезпечуючи захист фізичної інфраструктури, керування доступом, журналювання подій та контроль змін [31]. Для користувачів платформи це означає можливість будувати власні інформаційні системи на базі хмарної інфраструктури, яка вже відповідає базовим вимогам стандарту.

Важливу роль у забезпеченні відповідності міжнародним стандартам відіграють рекомендації *NIST*, зокрема стандарт NIST SP 800-53, який визначає перелік контрольних заходів для захисту інформаційних систем. У контексті Microsoft

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				26
Змн.	Арк.	№ докум.	Підпис	Дата		

Azure ці вимоги реалізуються через багаторівневі механізми безпеки, що включають управління ідентифікацією та доступом, мережну сегментацію, захист даних та постійний моніторинг. Azure Defender for Cloud дозволяє зіставляти конфігурацію ресурсів із вимогами NIST та виявляти відхилення, надаючи рекомендації щодо їх усунення [7].

Ще одним поширеним орієнтиром у сфері безпеки є *CIS Benchmarks*, які містять практичні рекомендації щодо безпечної конфігурації операційних систем, віртуальних машин, мережних компонентів і хмарних сервісів. У Microsoft Azure ці рекомендації можуть застосовуватися через автоматизовані політики безпеки, що дозволяють забезпечити відповідність конфігурацій рекомендованим налаштуванням. Використання CIS Benchmarks сприяє зменшенню поверхні атаки та мінімізації ризиків, пов'язаних із помилками конфігурації.

Центральним інструментом реалізації стандартів безпеки в Azure є *Azure Policy*, який дозволяє централізовано керувати відповідністю ресурсів визначеним вимогам. Політики можуть застосовуватися на рівні підписки або ресурсної групи та виконувати різні функції: від аудиту поточного стану до заборони створення ресурсів із небезпечними параметрами або автоматичного розгортання необхідних компонентів безпеки [2]. Завдяки цьому забезпечується уніфікований підхід до дотримання стандартів ISO, NIST та CIS у межах всієї хмарної інфраструктури.

Додатковим рівнем контролю є *Defender for Cloud*, який виконує функцію постійного моніторингу стану безпеки та відповідності стандартам. Цей сервіс аналізує конфігурації ресурсів, виявляє потенційні вразливості та формує рекомендації щодо їх усунення [15]. Defender for Cloud дозволяє оцінювати рівень відповідності інфраструктури визначеним стандартам та відстежувати динаміку змін безпекового стану системи.

Забезпечення відповідності стандартам безпеки неможливе без належного управління журналами подій та моніторингу. У Azure ці завдання реалізуються через *Log Analytics Workspace* та інтеграцію з системами SIEM, зокрема Azure Sentinel. Збір та аналіз логів дозволяє проводити аудит доступу, виявляти підозрілу активність та забезпечувати доказову базу для перевірок відповідності стандартам.

Це є важливим елементом не лише технічного, а й організаційного аспекту інформаційної безпеки.

1.4.3. Azure Advisor, Azure Policy та Azure Compliance Documentation

Забезпечення високого рівня безпеки та відповідності хмарної інфраструктури сучасним вимогам неможливе без використання інструментів централізованого аналізу, контролю та документального підтвердження виконання стандартів. У середовищі Microsoft Azure такі функції реалізуються за допомогою сервісів Azure Advisor, Azure Policy та комплексу матеріалів Azure Compliance Documentation, які у сукупності формують основу для управління архітектурною якістю, безпекою та регуляторною відповідністю вебдодатків.

Azure Advisor є сервісом рекомендацій, що здійснює безперервний аналіз стану ресурсів у хмарному середовищі та надає пропозиції щодо їх оптимізації. Основна мета Azure Advisor полягає у виявленні потенційних ризиків і неефективних конфігурацій на ранніх етапах експлуатації інфраструктури [1].

У контексті розробки захищеної хмарної інфраструктури для вебдодатку Azure Advisor відіграє важливу роль у підвищенні рівня безпеки. Він може виявляти відкриті мережні порти, надмірні права доступу, відсутність шифрування або застарілі конфігурації ресурсів. Отримані рекомендації дозволяють своєчасно усунути виявлені недоліки та зменшити ймовірність успішних атак. Таким чином, Azure Advisor виконує превентивну функцію, спрямовану на зниження ризиків ще до виникнення інцидентів безпеки. Приклад панелями з рекомендаціями від Azure Advisor зображено на рисунку 1.5 [3].

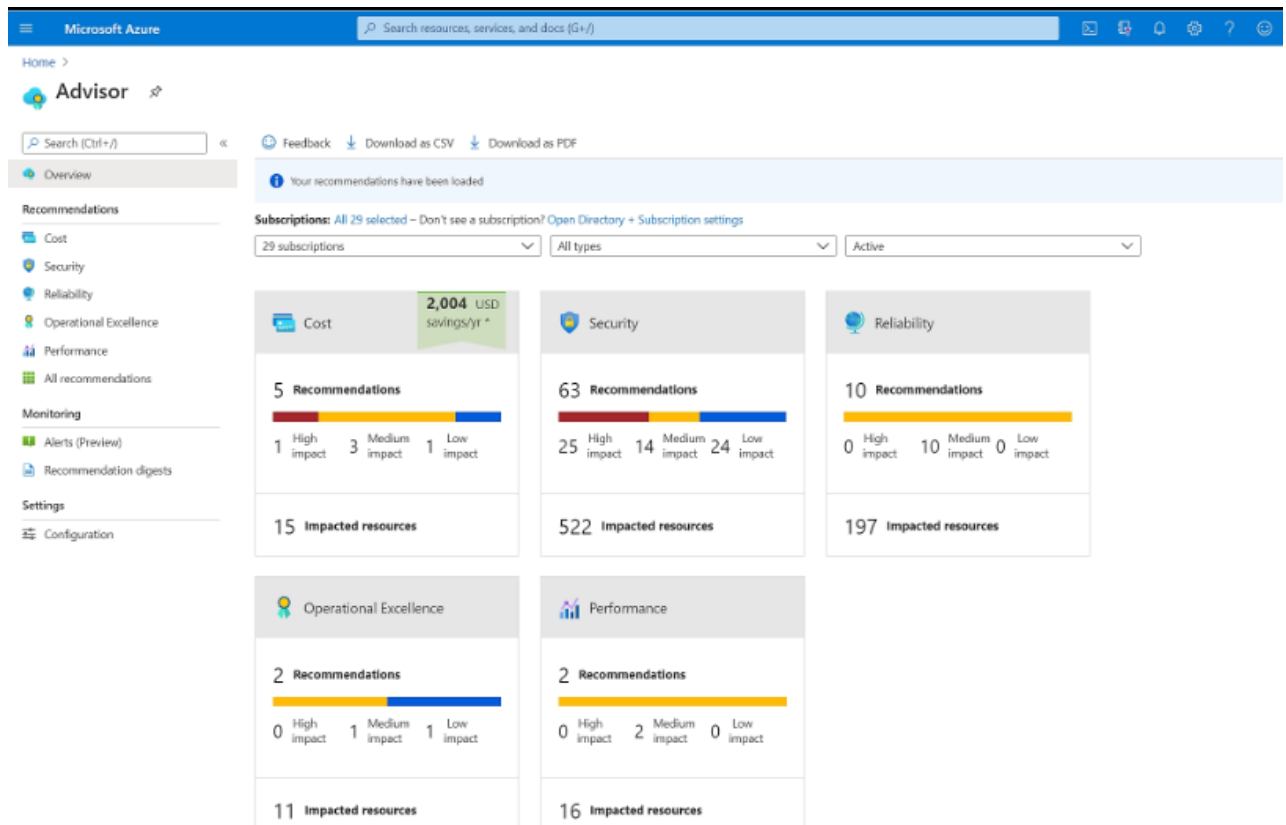


Рисунок 1.5 – Вигляд сторінки Azure Advisor

Для забезпечення обов'язкового виконання правил безпеки використовується *Azure Policy*, який є інструментом централізованого управління конфігураціями хмарних ресурсів. *Azure Policy* дозволяє визначати політики, що регламентують допустимі параметри ресурсів, та автоматично застосовувати їх у межах підписки або ресурсної групи.

Azure Policy підтримує декілька типів політик, зокрема *Audit*, *Deny* та *DeployIfNotExists*. Політики типу *Audit* дозволяють виявляти ресурси, що не відповідають встановленим вимогам, без обмеження їх функціонування. Політики типу *Deny* забороняють створення або зміну ресурсів із небезпечними або небажаними параметрами, що забезпечує жорсткий контроль конфігурацій. Політики *DeployIfNotExists* дозволяють автоматично розгортати необхідні компоненти безпеки, наприклад діагностичні налаштування або журнали аудиту, у разі їх відсутності.

У контексті захищеної хмарної інфраструктури для вебдодатку *Azure Policy* дозволяє забезпечити відповідність архітектури вимогам безпеки та комплаєнсу.

Зокрема, за допомогою політик можна обмежити публічний доступ до ресурсів, вимагати використання Private Endpoints, контролювати налаштування мережних груп безпеки та забезпечувати шифрування даних. Такий підхід мінімізує ризики, пов'язані з людським фактором, та забезпечує єдині стандарти безпеки в межах усієї інфраструктури.

Документальним підґрунтям реалізації стандартів безпеки в Microsoft Azure є *Azure Compliance Documentation* - набір офіційних матеріалів, які описують відповідність платформи міжнародним стандартам, нормативним вимогам і галузевим рекомендаціям [31]. Ці документи містять інформацію про сертифікації, аудиторські звіти, описи контрольних заходів та механізмів захисту, реалізованих у платформі Azure.

Azure Compliance Documentation відіграє важливу роль у процесі підтвердження відповідності хмарної інфраструктури вимогам безпеки. Для організацій, що розробляють і експлуатують вебдодатки, ці документи можуть використовуватися як доказова база під час внутрішніх і зовнішніх аудитів. Вони дозволяють обґрунтувати вибір хмарної платформи та продемонструвати, що базова інфраструктура відповідає міжнародним стандартам.

Комплексне використання Azure Advisor, Azure Policy та Azure Compliance Documentation формує системний підхід до управління безпекою та відповідністю хмарної інфраструктури. У сукупності ці інструменти дозволяють створити захищене, кероване та регламентоване хмарне середовище для розміщення вебдодатків.

1.5. Ключові компоненти для інфраструктури вебдодатку

1.5.1. Azure Application Gateway та Web Application Firewall

У сучасних хмарних архітектурах захист вебдодатку на прикладному рівні є критично важливим, оскільки саме цей рівень найчастіше стає об'єктом цілеспрямованих атак. У розроблюваній інфраструктурі вхідний мережний трафік до вебдодатку не надходить безпосередньо на віртуальну машину, а спочатку обробляється сервісом Azure Application Gateway, який виконує роль

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				30
Змн.	Арк.	№ докум.	Підпис	Дата		

балансувальника навантаження цього рівня моделі OSI. Такий підхід дозволяє реалізувати додатковий рівень захисту, централізовано контролювати доступ та мінімізувати ризики компрометації прикладного рівня.

Azure Application Gateway забезпечує інтелектуальну маршрутизацію HTTP та HTTPS-трафіку на основі параметрів запитів, зокрема URL-шляхів, заголовків та хост-імен. Завдяки роботі на прикладному рівні, сервіс може аналізувати вміст запитів і застосовувати правила безпеки до того, як трафік буде передано на внутрішні ресурси. У контексті даної архітектури Application Gateway розташовується у віртуальній мережі та спрямовує перевірений трафік до вебсервера Nginx, який працює на віртуальній машині з операційною системою Ubuntu Server 24.04 і обслуговує Spring Boot вебдодаток [8, 19].

Однією з ключових функцій Azure Application Gateway є підтримка SSL/TLS termination, що дозволяє завершувати зашифровані з'єднання на рівні шлюзу. Це забезпечує централізоване управління SSL-сертифікатами та зменшує навантаження на вебсервер, оскільки операції шифрування та дешифрування виконуються на рівні шлюзу [30]. SSL-сертифікати можуть зберігатися безпосередньо в Application Gateway або інтегруватися з сервісом Azure Key Vault, що відповідає принципам безпечного управління секретами та ключами. Передача трафіку між Application Gateway та внутрішніми ресурсами може здійснюватися як у зашифрованому, так і в незашифрованому вигляді, залежно від вимог безпеки та архітектурних рішень.

Для захисту вебдодатку від типових атак на прикладному рівні в Azure Application Gateway активується WAF. WAF є спеціалізованим механізмом безпеки, який аналізує HTTP/HTTPS-запити та застосовує набір правил для виявлення і блокування зловмисної активності. Основою роботи WAF є правила, що відповідають рекомендаціям OWASP Top 10, які охоплюють найпоширеніші та найнебезпечніші загрози для вебдодатків [28].

Захист від атак типу SQL Injection реалізується шляхом аналізу параметрів запитів і виявлення підозрілих конструкцій, які можуть бути використані для несанкціонованого доступу до бази даних. WAF також здатний виявляти спроби XSS, що полягають у вставці шкідливих скриптів у вебсторінки з метою

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				31
Змн.	Арк.	№ докум.	Підпис	Дата		

компрометації користувацьких сесій. Окрім цього, правила WAF дозволяють блокувати атаки, пов'язані з некоректною автентифікацією, порушенням контролю доступу, небезпечними десеріалізаціями та використанням вразливих компонентів.

Важливою особливістю Azure WAF є можливість роботи в різних режимах, зокрема Detection та Prevention. У режимі Detection система лише реєструє потенційно небезпечні запити, що дозволяє проаналізувати трафік і налаштувати правила безпеки перед виходом у production. Команди розробки мають можливість переконатися, що WAF не ламатиме функціонал додатку. Режим Prevention забезпечує активне блокування зловмисних запитів у реальному часі, що є доцільним для production середовища після належного тестування.

Окрім захисту від прикладних атак, Azure Application Gateway у поєднанні з WAF відіграє важливу роль у протидії розподіленим атакам відмови в обслуговуванні [6]. Хоча Application Gateway не замінює спеціалізований сервіс Azure DDoS Protection, він забезпечує базовий рівень захисту шляхом фільтрації аномального трафіку та обмеження кількості запитів. Інтеграція з Azure DDoS Protection Standard дозволяє підвищити рівень захисту за рахунок автоматичного виявлення та нейтралізації великих обсягів зловмисного трафіку на мережному рівні.

Отже, використання Azure Application Gateway у поєднанні з Web Application Firewall є ключовим елементом розроблюваної хмарної інфраструктури для вебдодатку. Ці сервіси забезпечують контроль вхідного трафіку, захист від атак OWASP Top 10, базову протидію DDoS-атакам та безпечне управління SSL-з'єднаннями, що дозволяє створити надійне та захищене середовище для функціонування Spring Boot вебдодатку в хмарі Microsoft Azure.

1.5.2. Azure Bastion

У процесі розгортання хмарної інфраструктури критично важливим аспектом є безпечний адміністративний доступ до віртуальних машин. Традиційне підключення через протоколи RDP або SSH із публічних IP-адрес несе високий ризик несанкціонованого доступу та є потенційною точкою проникнення для атак ззовні. Для мінімізації таких ризиків у Microsoft Azure застосовується сервіс Azure Bastion,

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				
Змн.	Арк.	№ докум.	Підпис	Дата		32

який забезпечує безпечний доступ до віртуальних машин без відкриття публічних портів і без використання VPN [12, 19].

Всі сеанси SSH відбуваються через зашифрований канал, що повністю ізолює внутрішні ресурси від публічного інтернету. Завдяки цьому відсутня необхідність у наданні віртуальним машинам публічних IP-адрес, що значно знижує поверхню атаки. Архітектура Azure Bastion представлено на рисунку 1.6.

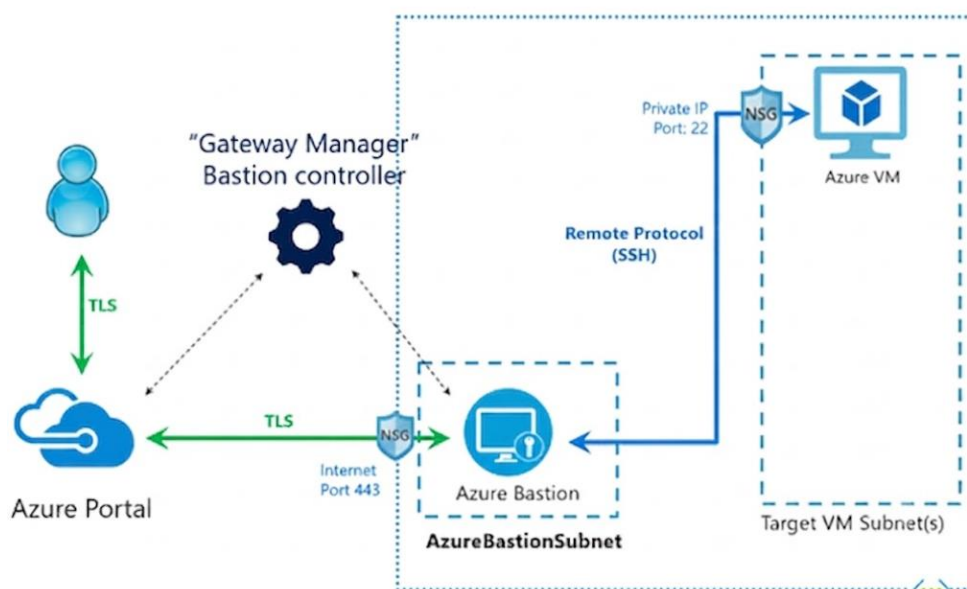


Рисунок 1.6 – Схема вузла Azure Bastion розгорнутий у віртуальній мережі

Функціонально Azure Bastion інтегрується з Virtual Network, у межах якої розгорнуті віртуальні машини. Це дозволяє забезпечити централізований контроль доступу, вести аудит сеансів та застосовувати політики безпеки на рівні підмережі. Всі підключення реєструються в Log Analytics Workspace, що забезпечує можливість моніторингу та розслідування інцидентів [22].

Важливим аспектом використання Azure Bastion є оцінка економічної доцільності, оскільки сервіс тарифікується за час роботи та обсяг підключень. Для невеликих або тестових середовищ може виникати питання доцільності використання Bastion у продуктивному режимі, проте з точки зору безпеки його застосування є оптимальним і відповідає рекомендаціям Well-Architected Framework щодо захисту адміністративного доступу.

1.5.3. Віртуальна мережа та DNS-інфраструктура

Віртуальна мережа у Microsoft Azure є ключовим компонентом хмарної інфраструктури, що забезпечує логічну ізоляцію ресурсів та контроль взаємодії між ними. VNet дозволяє створити приватне середовище, у якому віртуальні машини, бази даних, сховища та інші сервіси можуть обмінюватися даними без необхідності виходу в публічний інтернет, що значно підвищує рівень безпеки [18]. Такий підхід відповідає принципам багаторівневого захисту та зменшує поверхню атаки, оскільки критичні компоненти системи не мають безпосереднього контакту з зовнішнім середовищем.

Важливим елементом побудови VNet є поділ на підмережі, що дозволяє виділити окремі сегменти для різних функцій: вебшари, бази даних, служб моніторингу або внутрішніх адміністративних сервісів. Кожна підмережа може бути захищена за допомогою NSG, які визначають правила доступу до ресурсів на основі IP-адрес, портів і протоколів [24]. Завдяки NSG можна забезпечити принцип мінімального привілею, дозволяючи взаємодію лише між необхідними компонентами та блокуючи будь-які несанкціоновані підключення.

У сучасних архітектурах велике значення має використання Private Endpoints. Вони дозволяють підключати сервіси, такі як Azure SQL Database або Azure Storage, без надання публічного доступу через інтернет [23]. Трафік між компонентами проходить виключно всередині приватної мережі Azure і зашифровується стандартами TLS, що гарантує безпечну передачу даних. Приватні кінцеві точки також спрощують управління доступом та інтегруються з Azure RBAC, забезпечуючи контроль на рівні користувачів та сервісів.

DNS-інфраструктура є невід’ємною складовою організації хмарної мережі. Azure DNS дозволяє централізовано керувати публічними та приватними DNS-зонами, забезпечуючи коректне розв’язання імен ресурсів у межах VNet або для зовнішніх користувачів. Використання приватних DNS-зон для внутрішніх сервісів дозволяє забезпечити ізоляцію імен та уникнути конфліктів з публічними доменами. Для вебдодатку, що розгортається, доцільно використовувати публічні DNS-зони для зовнішнього доступу та приватні зони для внутрішньої маршрутизації між компонентами, наприклад між VM з Nginx, базою даних і сховищем.

Ще одним важливим аспектом є застосування резервованих IP-адрес для ключових ресурсів. Це забезпечує стабільність мережних маршрутів, спрощує конфігурацію SSL/TLS-сертифікатів і дозволяє уникнути непередбачуваної зміни IP при масштабуванні або відновленні ресурсів. У поєднанні з Private Endpoints це гарантує передбачувану маршрутизацію та стабільний зв'язок між компонентами вебдодатку.

Завдяки поєднанню VNet, Private Endpoints, NSG та централізованого управління DNS розроблена інфраструктура відповідає принципам Azure Well-Architected Framework, забезпечуючи безпеку, надійність, ефективність продуктивності та керованість [2, 13]. Така організація мережі дозволяє ефективно інтегрувати вебсервери, бази даних, сховища та сервіси моніторингу в єдине захищене середовище, знижуючи ризики несанкціонованого доступу та підвищуючи стійкість системи до зовнішніх і внутрішніх загроз.

1.5.4. Azure SQL

У сучасних хмарних архітектурах бази даних та сервіси зберігання даних є ключовими компонентами, від яких залежить надійність та безпека вебдодатків. У Microsoft Azure для цих цілей застосовуються керовані сервіси Azure SQL Database та Azure Storage Account, зокрема Blob Storage, що забезпечують високу доступність, масштабованість та інтеграцію з іншими компонентами інфраструктури.

Безпечне розгортання Azure SQL передбачає використання Private Endpoints, завдяки яким підключення до бази даних здійснюється виключно в межах приватної мережі, без публічного доступу через інтернет. Керування доступом реалізується через Entra ID та Role-Based Access Control, що дозволяє дотримуватися принципу мінімальних привілеїв та забезпечує контроль на рівні користувачів і сервісів [21]. Для захисту даних у стані спокою використовується Transparent Data Encryption, а для захисту під час передачі - протоколи SSL/TLS [20].

Моніторинг і аудит активності бази даних здійснюються через Diagnostic Settings та інтеграцію з Log Analytics Workspace, що забезпечує прозорість всіх операцій і дозволяє виявляти потенційні загрози на ранніх етапах. Автоматичне резервне копіювання та можливість відновлення даних у будь-який момент забезпечують стійкість системи до апаратних збоїв або випадкових видалень, що відповідає принципам надійності у рамках Azure Well-Architected Framework.

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				
Змн.	Арк.	№ докум.	Підпис	Дата		35

1.5.5. Сервіси моніторингу та захисту: Defender for Cloud, Azure Sentinel, Log Analytics Workspace

Ефективна безпека хмарної інфраструктури неможлива без комплексного моніторингу та оперативного реагування на інциденти. У Microsoft Azure ключовими інструментами для цих завдань є сервіси Defender for Cloud, Azure Sentinel та Log Analytics Workspace, які дозволяють здійснювати централізоване управління безпекою, збирати телеметрію та впроваджувати автоматизоване реагування на загрози.

Defender for Cloud виступає основним механізмом оцінки безпеки ресурсів Azure та забезпечення відповідності вимогам корпоративних політик. Сервіс здійснює безперервний аналіз стану ресурсів, виявляє конфігураційні ризики та надає рекомендації щодо підвищення безпеки, включно з шифруванням даних, застосуванням Private Endpoints, обмеженням публічного доступу та управлінням правами користувачів [15]. Використання Defender for Cloud дозволяє реалізувати принципи proactive security, забезпечуючи превентивне усунення вразливостей ще до появи потенційних інцидентів.

Для глибшого аналізу загроз та інтеграції з корпоративними процесами реагування використовується Azure Sentinel, який є рішенням класу SIEM + SOAR [16]. Sentinel дозволяє збирати дані з усіх компонентів інфраструктури, включаючи віртуальні машини, мережні ресурси, бази даних і сервіси зберігання, і проводити їх кореляцію для виявлення підозрілої активності. Завдяки інтеграції з Log Analytics, Sentinel може реалізовувати автоматизовані сценарії реагування на інциденти, що значно підвищує оперативність виявлення та локалізації загроз, а також дозволяє створювати централізовані дашборди для керівників і адміністративного персоналу.

Log Analytics Workspace є основним сховищем телеметричних даних, яке об'єднує журнали активності та метрики з усіх компонентів хмарної інфраструктури [11].

Висновок до першого розділу

У першому розділі сформовано теоретичний базис для проектування захищеної інфраструктури вебдодатка в Microsoft Azure згідно з принципами Well-Architected Framework. Обґрунтовано використання моделей IaaS та PaaS, а також визначено ключовий стек безпеки: Azure Application Gateway з WAF для захисту прикладного рівня (L7) від атак OWASP Top 10 та Azure Bastion для безпечного

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				
Змн.	Арк.	№ докум.	Підпис	Дата		36

адміністрування ВМ без публікації SSH-портів. Також окреслено архітектуру безперервного моніторингу подій безпеки на базі Log Analytics Workspace, Microsoft Defender for Cloud та Azure Sentinel, що дозволяє перейти до практичної реалізації та мережевої адресації проєкту.

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				37
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 2. ПРОЄКТУВАННЯ БЕЗПЕЧНОЇ ХМАРНОЇ ІНФРАСТРУКТУРИ ТА АРХІТЕКТУРИ SPRING BOOT ДОДАТКУ

2.1. Проєктування архітектури інфраструктури з забезпеченням вимог безпеки

Для розробки захищеної хмарної інфраструктури вебдодатку на Spring Boot у середовищі Azure було спроектовано архітектуру з врахуванням принципів Azure Well-Architected Framework та політик міжнародних стандартів безпеки ISO 27001 і NIST. Узагальнена архітектура розробленої системи для банківського додатку представлена на рисунку 2.1.

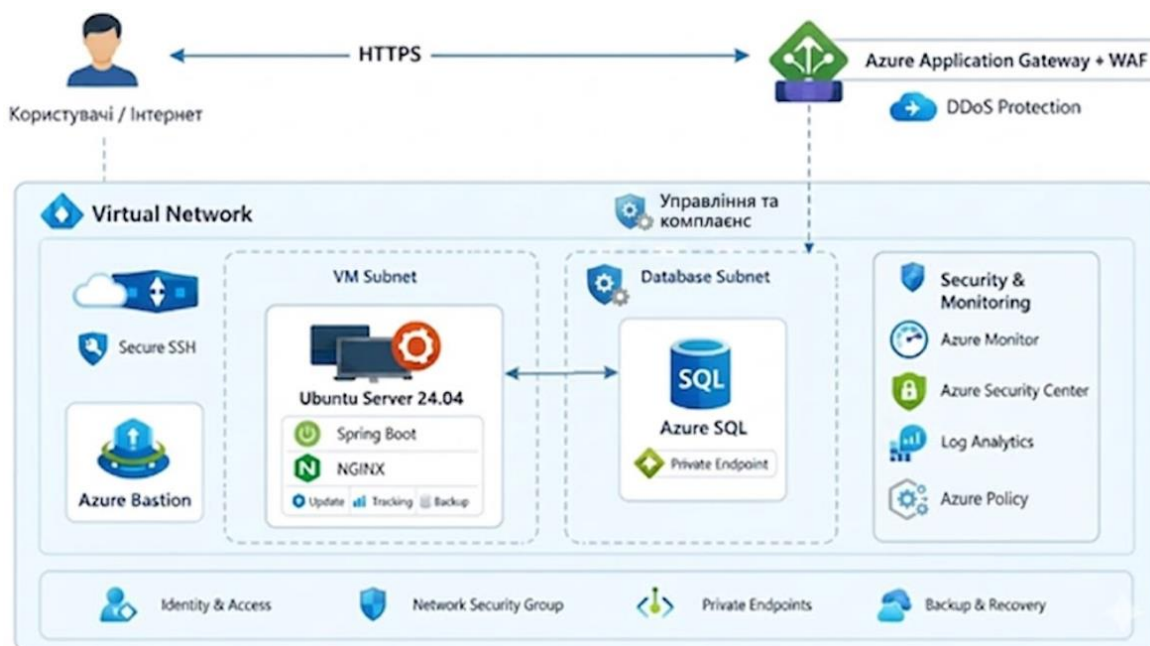


Рисунок 2.1 – Архітектура безпечної хмарної інфраструктури

Основні компоненти архітектури включають:

Віртуальна машина на Ubuntu Server 24.04: віртуальна машина слугує хостом для Spring Boot додатку та вебсерверу Nginx. Розгортання може здійснюватися через Azure Portal, CLI або PowerShell. Доступ до VM здійснюється через Azure Bastion, що забезпечує безпечний SSH-підключення без публічного IP-адреси. На VM налаштовані сервіси для управління операціями: Update Management, Change Tracking та Backup Policy.

Virtual Network та Private Endpoints: VNet забезпечує ізоляцію всіх ресурсів додатку: VM, Azure SQL. Для бази даних використано Private Endpoints, що дозволяє доступ тільки з приватної мережі та дозволених публічних адрес. Резервовані

IP-адреси застосовуються для ключових ресурсів, щоб уникнути конфліктів та спростити налаштування DNS.

Azure SQL Database: доступ до бази даних здійснюється через Managed Identity і Private Endpoint. Додатково застосовано Transparent Data Encryption, firewall та контроль доступу на рівні RBAC.

Azure DNS: Використання Private Endpoints разом із Azure DNS дозволяє ізолювати трафік усередині віртуальної мережі (VNet), зберігаючи стандартні доменні імена для внутрішніх сервісів.

- Керування внутрішніми сервісами: Через Azure Private DNS запити до хмарних ресурсів автоматично перенаправляються на внутрішні приватні IP-адреси (10.0.0.10) замість публічного інтернету.
- Керування DNS-записами додатку: Public Azure DNS використовується для маршрутизації зовнішнього трафіку користувачів до публічних вебдодатків та API.

Інтеграція з доменом pp.ua: Azure DNS підтримує повне делегування та керування безкоштовними доменами у зоні pp.ua. Це дозволяє безкоштовно налаштувати корпоративні або тестові доменні імена як для зовнішнього доступу, так і для внутрішньої розробки в закритому контурі.

Azure Application Gateway з WAF: служить балансувальником навантаження рівня L7 та забезпечує захист від атак OWASP Top 10 через Web Application Firewall. Додатково включено DDoS Protection Standard для захисту від масових атак.

Моніторинг та безпека: всі ресурси підключені до Defender for Cloud для оцінки загроз та рекомендацій щодо hardening. Логи збираються у Log Analytics Workspace, де формується дашборд та налаштовані алерти. Опціонально можна інтегрувати Azure Sentinel для SIEM/SOAR.

Логічна взаємодія компонентів

Потоки даних та шари безпеки: Користувач → HTTPS → Application Gateway → Nginx → Spring Boot додаток → Private Endpoint → Azure SQL Database.

Використовується Managed Identity для безпечного доступу: Spring Boot додаток → Private Endpoint.

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				39
Змн.	Арк.	№ докум.	Підпис	Дата		

Моніторинг та логування: Defender for Cloud + Log Analytics → дашборди та алерти.

2.2. Проєктування безпечної мережної інфраструктури та сервісів Azure для вебдодатку

2.2.1. Управління доступом та політиками безпеки

Мережна архітектура вебдодатку в середовищі Microsoft Azure проєктується з використанням сервісу Azure Virtual Network, який забезпечує логічну ізоляцію ресурсів та контроль мережного трафіку. Основною метою проєктування мережі є створення безпечного адресного простору для розміщення компонентів вебдодатку з урахуванням можливості подальшого масштабування та інтеграції з керованими сервісами Azure.

Віртуальна мережа проєктується з використанням приватного IP-простору відповідно до стандарту RFC 1918. Такий підхід дозволяє уникнути прямого доступу ресурсів з публічного інтернету та забезпечує контроль доступу до них через спеціалізовані мережні компоненти. Адресний простір мережі розподіляється на окремі підмережі відповідно до функціонального призначення ресурсів, що розміщуються в них.

При проєктуванні мережі враховується використання Azure Application Gateway як точки входу для вхідного HTTP/HTTPS трафіку, а також застосування Private Endpoint для безпечного підключення керованих сервісів, зокрема Azure SQL Database. Окремо виділяється підмережа для сервісу Azure Bastion, який використовується для безпечного адміністративного доступу до віртуальної машини без відкриття публічних IP-адрес.

Логічна схема віртуальної мережі передбачає розміщення всіх компонентів вебдодатку в межах однієї Azure Virtual Network з поділом на функціональні підмережі. Вхідний трафік з мережі Інтернет надходить до Azure Application Gateway, який виконує функції балансування навантаження та перевірки HTTP/HTTPS запитів. Після цього трафік перенаправляється до віртуальної машини з розгорнутим вебсервером Nginx та Spring Boot додатком.

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				40
Змн.	Арк.	№ докум.	Підпис	Дата		

Адміністративний доступ до віртуальної машини здійснюється за допомогою сервісу Azure Bastion. Це дозволяє повністю закрити вхідний SSH-трафік (порт 22) з мережі Інтернет на рівні NSG, мінімізуючи поверхню атак. При цьому сам SSH-доступ залишається активним виключно всередині VNet для забезпечення SSH тунелювання від підмережі Azure Bastion до цільової ВМ. Для доступу вебдодатка до бази даних використовується Private Endpoint, який забезпечує передавання трафіку до Azure SQL Database виключно через приватну мережу Azure та ряд довірених публічних адрес.

Таким чином, логічна схема мережі реалізує принцип сегментації та мінімізації поверхні атаки, де кожен компонент має чітко визначене місце та роль у загальній архітектурі.

Для віртуальної мережі вебдодатку обрано адресний простір 10.0.0.0/16, що забезпечує достатній запас IP-адрес для масштабування інфраструктури. Даний простір поділяється на кілька підмереж відповідно до призначення ресурсів, що відображено на таблиці 2.1.

Таблиця 2.1

Схема сегментації хмарної мережі та розподіл IP-адрес

Назва підмережі / Сервіс	Діапазон IP-адрес (CIDR)
Підмережа для Azure Application Gateway	10.0.1.0/28
Підмережа для віртуальної машини з вебдодатком	10.0.2.0/28
Підмережа для Azure Private DNS	10.0.3.0/29
Підмережа для Azure Bastion	10.0.4.0/27

Використання окремої підмережі для Azure Bastion є обов’язковою вимогою сервісу, а мінімальний розмір /27 дозволяє оптимально використовувати адресний простір. При створенні підмереж враховується, що в Azure частина IP-адрес у кожній підмережі є зарезервованою для службових потреб платформи, зокрема адреси мережі, шлюзу та внутрішніх сервісів.

Запропонована схема адресації забезпечує логічну ізоляцію компонентів, контроль мережного доступу та створює основу для безпечного підключення керованих сервісів через Private Endpoint. Високорівнева структура маршрутизації вебтрафіку представлена на рисунку 2.2.

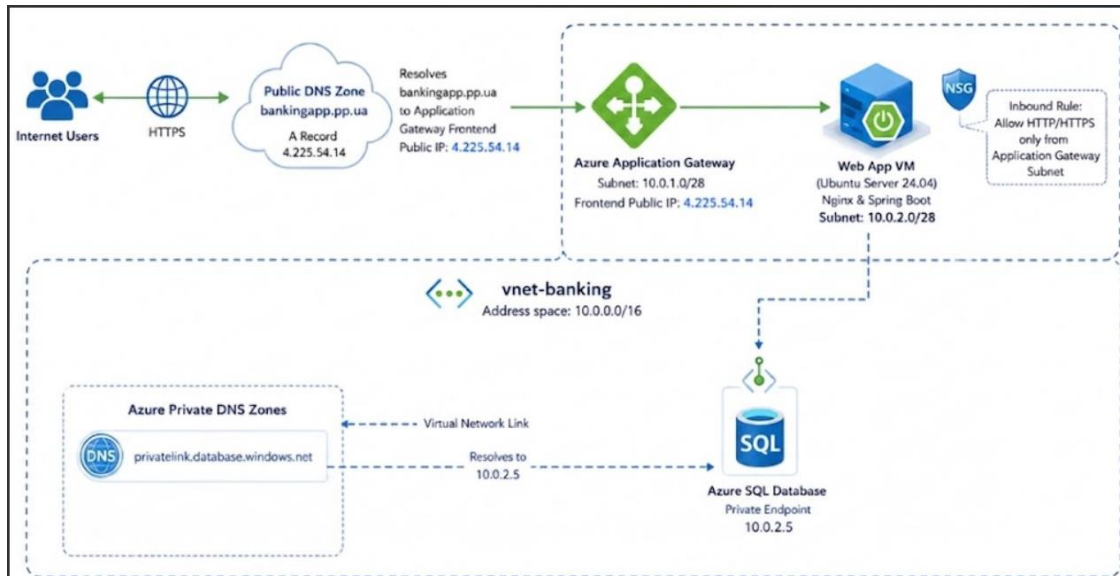


Рисунок 2.2 –Маршрутизація трафіку та resolution DNS для банківського додатку

Для забезпечення безперервного контролю за станом безпеки хмарної інфраструктури та її відповідності міжнародним стандартам безпеки, у проєкті реалізовано механізм автоматизованого управління політиками за допомогою сервісу Azure Policy. Зокрема, було розроблено кастомну Initiative Definition стандарту ISO/IEC 27001:2022, яка об'єднує 10 специфічних правил контролю.

Розгортання зазначеної ініціативи здійснюється декларативно через Azure CLI з використанням шаблонів інфраструктури як коду у форматі JSON. Процес успішного створення розгортання ініціативи політик безпеки на рівні Azure-підписки через командний рядок відображено на рисунку 2.3.

```
kb221_diploma3_kmo [ ~ ]$ nano policies_iso27001.json
kb221_diploma3_kmo [ ~ ]$ az deployment sub create --location swedencentral --template-file policies_iso27001.json
{
  "id": "/subscriptions/1e92f76a-96c0-4261-9d65-0ec110f2bb44/providers/Microsoft.Resources/deployments/policies_iso27001",
  "location": "swedencentral",
  "name": "policies_iso27001",
  "properties": {
    "correlationId": "481ecf3e-fa17-4266-87e1-489e13ca0f6a",
    "debugSetting": null,
    "dependencies": [
      {
        "dependsOn": [
          {
            "id": "/subscriptions/1e92f76a-96c0-4261-9d65-0ec110f2bb44/providers/Microsoft.Authorization/policyDefinitions/custom-sql-public-access",
            "resourceName": "custom-sql-public-access",
            "resourceType": "Microsoft.Authorization/policyDefinitions"
          },
          {
            "id": "/subscriptions/1e92f76a-96c0-4261-9d65-0ec110f2bb44/providers/Microsoft.Authorization/policyDefinitions/custom-nsg-subnet",
            "resourceName": "custom-nsg-subnet",
            "resourceType": "Microsoft.Authorization/policyDefinitions"
          },
          {
            "id": "/subscriptions/1e92f76a-96c0-4261-9d65-0ec110f2bb44/providers/Microsoft.Authorization/policyDefinitions/custom-sql-tls12",
            "resourceName": "custom-sql-tls12",
            "resourceType": "Microsoft.Authorization/policyDefinitions"
          },
          {
            "id": "/subscriptions/1e92f76a-96c0-4261-9d65-0ec110f2bb44/providers/Microsoft.Authorization/policyDefinitions/custom-allowed-vm-skus",
            "resourceName": "custom-allowed-vm-skus",
            "resourceType": "Microsoft.Authorization/policyDefinitions"
          }
        ]
      },
      {
        "id": "/subscriptions/1e92f76a-96c0-4261-9d65-0ec110f2bb44/providers/Microsoft.Authorization/policyDefinitions/custom-allowed-resources",
        "resourceName": "custom-allowed-resources",
        "resourceType": "Microsoft.Authorization/policyDefinitions"
      }
    ]
  }
}
```

Рисунок 2.3 – Результат успішного розгортання кастомної політик безпеки policies_iso27001 через Azure CLI

Після завершення процесу деплою та реєстрації шаблону в хмарі, створена кастомна ініціатива стає доступною для призначення (assignment) та моніторингу безпосередньо в графічному інтерфейсі порталу Microsoft Azure. Опис структури створеної кастомної ініціативи відповідності «ISO/IEC 27001:2022 Custom Compliance» із переліком усіх впроваджених внутрішніх правил та визначеними для них ефектами (Deny, Audit) наведено на рисунку 2.4.

Microsoft Azure

Search resources, services, and docs (G+V)

Copilot

kb221_diploma3_2022@ДЕРЖАВНИЙ УНІВЕРСИТЕТ

Home > Policy | Definitions

ISO/IEC 27001:2022 Custom Compliance

Initiative Definition

Assign initiative

Edit initiative

Duplicate definition

Delete initiative

Essentials

Name

ISO/IEC 27001:2022 Custom Compliance

Definition location

Azure for Students

Description

Поєна кастомна ініціатива відповідності стандарту ISO/IEC 27001:2022 для дипломного п...

Definition ID

/subscriptions/1e92f76a-96c0-4261-9d65-0ec110f2bb44/providers/Microsoft.Authorization/...

Category

Regulatory Compliance

Type

Custom

Version

1.0.0

Policies (10)

Groups (0)

Parameters (2)

JSON

Assignments

Filter by reference ID, policy name or ID...

Type : All selected

Evaluation type : All selected

Edit columns

Export to CSV

Policy	Reference ID	Type	Evaluation type	Default effect
Custom SQL Server public network access disabled	Ctrl_1_SQL_PublicAccess	Custom	Automated	Deny
Custom NSG should be associated with subnet	Ctrl_2_NSG_Subnet	Custom	Automated	Audit
Custom Minimum TLS version 1.2 for SQL	Ctrl_3_SQL_TLS12	Custom	Automated	Deny
Custom Allowed VM SKUs	Ctrl_4_VM_SKUs	Custom	Automated	Deny
Custom Allowed Resource Types	Ctrl_5_Allowed_Resources	Custom	Automated	Deny
Custom Required Tags	Ctrl_6_Required_Tags	Custom	Automated	Audit
Custom Managed disks should be encrypted	Ctrl_7_Disk_Encryption	Custom	Automated	Audit
Custom Azure Backup should be enabled for VMs	Ctrl_8_VM_Backup	Custom	Automated	Audit
Custom Architectural Control: Use Bastion instead of open SSH/RDP	Ctrl_9_No_Direct_Internet_SSH_RDP	Custom	Automated	Audit
Custom Managed Identity for application authentication	Ctrl_10_Managed_Identity	Custom	Automated	Audit

Рисунок 2.4 – Графічне представлення кастомної ініціативи відповідності ISO/IEC 27001:2022 та впроваджених правил безпеки в Azure Portal

2.2.2. Забезпечення безпечного мережного доступу та маршрутизації трафіку

У процесі реалізації хмарної інфраструктури на платформі Microsoft Azure було здійснено комплексне управління доступом та забезпечення політик безпеки, орієнтованих на сучасні вимоги контролю доступу, відповідності стандартам безпеки та мінімізації ризиків конфігураційних невідповідностей. Ключовими компонентами цього процесу були: застосування політик управління конфігураціями через Azure Policy, організація контролю доступу на основі ролей, а також впровадження механізмів безпарольної автентифікації за допомогою Managed Identities. Використані підходи враховують рекомендації Microsoft щодо безпеки хмарної інфраструктури та забезпечують відповідність вимогам міжнародних стандартів ISO/IEC та NIST SP 800-53, які є загальноприйнятими рамками в галузі інформаційної безпеки.

Застосування політик для управління інфраструктурою з Azure Policy

Одним із важливих аспектів забезпечення безпеки хмарної інфраструктури є моніторинг і впровадження політик конфігурації, які гарантують відповідність ресурсів правилам безпеки й корпоративним стандартам. У даній роботі для цього використано ініціативу Microsoft Cloud Security Benchmark, яка є розвитком та розширенням попереднього Azure Security Benchmark, що інтегрує понад сотню контрольних рекомендацій для безпеки хмарних робочих навантажень і забезпечує узгодження з багатьма загальноприйнятими стандартами, такими як CIS Controls, ISO/IEC 27001 і NIST SP 800-53. Обрана ініціатива представляє собою набір політик, призначених для оцінки конфігурацій ресурсів у Azure і моніторингу їх відповідності вимогам безпеки в режимі аудиту без автоматичного примусового усунення невідповідностей. Такий підхід є прийнятним для Azure for Students, де автоматичне виправлення може бути обмежено, а також відповідає принципу уникнення непередбачених змін у робочих системах.

Реалізація політики передбачала призначення Policy Initiative через інтерфейс Azure Portal: у блоці Policy → Assignments → Assign policy було обрано Microsoft

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				
Змн.	Арк.	№ докум.	Підпис	Дата		44

Cloud Security Benchmark, область застосування визначена для конкретної Resource Group, у якій створено віртуальну машину. Під час впровадження політик також аналізовано обмеження функціональних можливостей середовища Azure for Students, що вимагало уточнення підтримки окремих політик у складі ініціативи і врахування їх впливу на результати перевірок. Дані політики обмежували регіони розгортання ресурсів.

2.2.3. Захист та балансування вхідного і міжмережного трафіку

Балансування вхідного трафіку з використанням Application Gateway

Було налаштовано сервіс Azure Application Gateway, який виконує функції рівномірного розподілу вхідних HTTP/HTTPS-запитів між серверними компонентами системи. У процесі налаштування було створено Frontend IP для приймання зовнішніх запитів та налаштовано Backend pool, до якого входить віртуальна машина з додатком, а також Listener-и, що визначають правила обробки та маршрутизації трафіку. Застосування Application Gateway дозволило підвищити доступність вебдодатків, оптимізувати використання обчислювальних ресурсів та зменшити ризик перевантаження серверу.

Захист від атак OWASP Top 10 з використанням Azure Web Application Firewall

Впроваджено механізми захисту вебдодатків від типових загроз прикладного рівня. Для цього створено та налаштовано WAF Policy з використанням набору правил OWASP 3.2, який охоплює основні категорії веб-атак, зокрема SQL Injection, Cross-Site Scripting та інші поширені вразливості. Політика безпеки була асоційована з Application Gateway, що забезпечує автоматичну перевірку всього вхідного трафіку. Додатково налаштовано збір діагностичних даних і журналів безпеки з передачею до Log Analytics workspace, що забезпечує можливість централізованого моніторингу та аналізу інцидентів інформаційної безпеки.

2.2.4. Моніторинг, журналювання та реагування на інциденти

Для централізованого збору та зберігання журналів було використано Log Analytics Workspace, який виступає єдиною точкою агрегації логів з ресурсів Azure.

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				45
Змн.	Арк.	№ докум.	Підпис	Дата		

У межах дослідження працездатності розгорнутої інфраструктури було організовано моніторинг стану бекенд-серверу Application Gateway. Через аналіз таблиці системних метрик AzureMetrics підтверджено коректне функціонування вузла з розміщеним вебдодатком, що ілюструє рисунок 2.5.

AzureMetrics

Time range : Last 24 hours

Show : 1000 results

Add

Simple mode

Results

Chart

TimeGenerated [UTC]	ResourceId	Resource	ResourceGroup	ResourceProvider	SubscriptionId	MetricName	Total	Count
> 6/9/2026, 8:07:00.000 AM	/SUBSCRIPTIONS/1E9276A-96C0-4261-9D65-0EC110F2B844	APPGW-PROD	RG-SWC-BANKING-APP	MICROSOFT.NETWORK	1e9276a-96c0-4261-9d65-0ec110f2bb44	UnhealthyHostCount	0	2
6/9/2026, 8:07:00.000 A...	/SUBSCRIPTIONS/1E9276A-96C0-4261-9D65-0EC110F2B844	APPGW-PROD	RG-SWC-BANKING-APP	MICROSOFT.NETWORK	1e9276a-96c0-4261-9d65-0ec110f2bb44	HealthyHostCount	2	2
TenantId29b895ca-5c23-4f81-b249-7bae9e044538								
SourceSystemAzure								
TimeGenerated [UTC]2026-06-09T08:07:00Z								
ResourceId/SUBSCRIPTIONS/1E9276A-96C0-4261-9D65-0EC110F2B844/RESOURCEGROUPS/rg-swc-banking-app/PROVIDERS/MICROSOFT.NETWORK/APPLICATIONGATEWAYS/APPGW-PROD								
ResourceAPPGW-PROD								
ResourceGroupRG-SWC-BANKING-APP								
ResourceProviderMICROSOFT.NETWORK								
SubscriptionId1e9276a-96c0-4261-9d65-0ec110f2bb44								
MetricNameHealthyHostCount								
Total2								
Count2								
Maximum1								
Minimum1								
Average1								
TimeGrainPT1M								
UnitNameCount								
TypeAzureMetrics								
_ResourceId/subscriptions/1e9276a-96c0-4261-9d65-0ec110f2bb44/resourcegroups/rg-swc-banking-app/providers/microsoft.network/applicationgateways/appgw-prod								

Рисунок 2.5 – Розгорнутий лог однієї метрики з таблиці AzureMetrics в Azure Log Analytics для Application Gateway

Для забезпечення детального аудиту мережної безпеки було активовано збір логів доступу (AGWAccessLogs) та загальної діагностики системних подій (AzureDiagnostics). Це дозволяє відстежувати легітимні запити користувачів, аналізувати структуру HTTP-трафіку та фіксувати невідповідності протоколам (наприклад, запити без обов'язкових HTTP-заголовків), прикладах яких представлені на рисунках 2.6 та 2.7.

AGWAccessLogs

Time range : Last 24 hours

Show : 1000 results

Add

Simple mode

ResultsChart

TimeGenerated [UTC]

↑↓

OperationName

ListenerName

RuleName

BackendPoolName

BackendSettingName

Instanceld

ClientIp

ClientPort

HttpMethod

>

6/9/2026, 6:31:02.000 AM

ApplicationGatewayAccess

https-listener-diploma

https-routing-rule

ubuntu-nginx-pool

http-to-nginx-settings

appgw_1

217.165.22.155

59244

GET

∨

6/9/2026, 6:31:02.000 AM

ApplicationGatewayAccess

https-listener-diploma

https-routing-rule

ubuntu-nginx-pool

http-to-nginx-settings

appgw_1

217.165.22.155

59244

GET

TenantId

29b895ca-5c23-4f81-b249-7bae9e044538

TimeGenerated [UTC]

2026-06-09T06:31:02Z

OperationName

ApplicationGatewayAccess

ListenerName

https-listener-diploma

RuleName

https-routing-rule

BackendPoolName

ubuntu-nginx-pool

BackendSettingName

http-to-nginx-settings

Instanceld

appgw_1

ClientIp

217.165.22.155

ClientPort

59244

HttpMethod

GET

RequestUri

/favicon.ico

UserAgent

Mozilla/5.0 (Linux; Android 10; K) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/148.0.0.0 Mobile Safari/537.36 EdgA/148.0.0.0

HttpStatus

302

HttpVersion

HTTP/2.0

ReceivedBytes

28

SentBytes

222

Коротун М.О.

Миколайчук В.В.

Змн.

Арк.

№ докум.

Підпис

Дата

ККІКБ.КР.Б – 125 – 06 – ПЗ

Арк.

46

Рисунок 2.6 –Розгорнутий лог однієї метрики з таблиці AGWAccessLogs в Azure Log Analytics, який фіксує успішне HTTP-підключення користувача до вебдодатка через Azure Application Gateway

AzureDiagnostics

Time range : Last 24 hours

Show : 1000 results

Add

Simple m...

Results

Chart

TimeGenerated [UTC]	↑↓	ResourceId	Category	ResourceGroup	SubscriptionId	ResourceProvider	
>		6/9/2026, 8:12:13.000 AM	/SUBSCRIPTIONS/1E92F76A-96C0-4261-9D65-0EC110F2BB...	ApplicationGatewayAccessLog	RG-SWC-BANKING-APP	1e92f76a-96c0-4261-9d65-0ec110f2bb44	MICROSOFT.NETWORK
>		6/9/2026, 8:11:49.000 AM	/SUBSCRIPTIONS/1E92F76A-96C0-4261-9D65-0EC110F2BB...	ApplicationGatewayAccessLog	RG-SWC-BANKING-APP	1e92f76a-96c0-4261-9d65-0ec110f2bb44	MICROSOFT.NETWORK
▼		6/9/2026, 8:09:36.379 A...	/SUBSCRIPTIONS/1E92F76A-96C0-4261-9D65-0EC110F2BB...	ApplicationGatewayFirewall...	RG-SWC-BANKING-APP	1e92f76a-96c0-4261-9d65-0ec110f2bb44	MICROSOFT.NETWORK
		TenantId	29b895ca-5c23-4f81-b249-7bae9e044538				
		TimeGenerated [UTC]	2026-06-09T08:09:36.3796507Z				
		ResourceId	/SUBSCRIPTIONS/1E92F76A-96C0-4261-9D65-0EC110F2BB44/RESOURCEGROUPS/RG-SWC-BANKING-APP/PROVIDERS/MICROSOFT.NETWORK/APPLICATIONGATEWAYS/APPGW-PROD				
		Category	ApplicationGatewayFirewallLog				
		ResourceGroup	RG-SWC-BANKING-APP				
		SubscriptionId	1e92f76a-96c0-4261-9d65-0ec110f2bb44				
		ResourceProvider	MICROSOFT.NETWORK				
		Resource	APPGW-PROD				
		ResourceType	APPLICATIONGATEWAYS				
		OperationName	ApplicationGatewayFirewall				
		requestUri_s	/_layouts/15/debug_dev.js				
		Message	Request Missing an Accept Header				
		instanceId_s	appgw_1				
		SourceSystem	Azure				
		clientIp_s	64.62.156.10				
		ruleSetType_s	OWASP CRS				

Рисунок 2.7 –Розгорнутий лог однієї метрики з таблиці AzureDiagnostics в Azure Log Analytics, який фіксує підозрілий запит, що було заблоковано

Особливу увагу приділено моніторингу подій безпеки WAF. За допомогою запитів до таблиці AGWFirewallLogs у Log Analytics Workspace реалізовано відстеження та аналіз спроб кібератак на захищений вебдодаток. Результати логуювання демонструють успішне виявлення сигнатур атаки типу SQL Injection на основі базового набору правил OWASP CRS, що представлений на рисунку 2.8.

AGWFirewallLogs

Time range : Last 24 hours

Show : 1000 results

▼ Add

Simple mode

Results

Chart

TimeGenerated [UTC] ↑↓	OperationName	InstancelId	ClientIp	RequestUri	RuleSetType	RuleSetVersion	RuleId	Message
> 6/9/2026, 6:22:32.000 AM	ApplicationGatewayFirewall	appgw_1	188.163.18.168	/login	OWASP CRS	3.2	942130	SQL Injection Attack: SQL Tautology Detected.
6/9/2026, 6:22:32.000 AM	ApplicationGatewayFirew...	appgw_1	188.163.18.168	/login	OWASP CRS	3.2	942110	SQL Injection Attack: Common Injection Testing Detec...
TenantId	29b895ca-5c23-4f81-b249-7bae9e044538							
TimeGenerated [UTC]	2026-06-09T06:22:32Z							
OperationName	ApplicationGatewayFirewall							
InstancelId	appgw_1							
ClientIp	188.163.18.168							
RequestUri	/login							
RuleSetType	OWASP CRS							
RuleSetVersion	3.2							
RuleId	942110							
Message	SQL Injection Attack: Common Injection Testing Detected							
Action	Matched							
DetailedMessage	Pattern match (?^\s*["'])(["'])(\s*\$) at ARGS.							
DetailedData	{ 'found within [ARGS:username:'1=1'] }							
FileDetails	REQUEST-942-APPLICATION-ATTACK-SQL.conf							
LineDetails	510							
Hostname	bankingapp.pp.ua							

2.3. Проєктна частина Java-додатку

2.3.1. Призначення додатку та функціональні вимоги

Mini Banking App - це навчальний вебдодаток, створений з метою демонстрації сучасних принципів побудови безпечних хмарних систем на платформі Azure з використанням фреймворку Spring Boot. Основне призначення додатку полягає у моделюванні функціоналу банківської системи малого масштабу, що дозволяє користувачам здійснювати базові фінансові операції та управляти своїми рахунками в безпечному середовищі. Додаток покликаний продемонструвати не лише організацію бізнес-логіки та управління даними, але й правильну інтеграцію з хмарними сервісами, включно з Azure SQL Database, із використанням Managed Identity і принципів безпечного доступу через Private Endpoints.

Проєкт розроблено з урахуванням принципів cloud-native архітектури та практик Well-Architected Framework, що передбачає модульність коду, масштабованість та простоту управління середовищем розгортання. Основна цінність додатку полягає у можливості дослідити та реалізувати механізми безпечного зберігання та обробки даних користувачів, забезпечити контроль доступу та реалізувати ролі користувачів, а також продемонструвати інтеграцію з сервісами моніторингу, логування та захисту від загроз, такими як Azure Defender та Web Application Firewall. Головна сторінка додатку відображена на рисунку 2.9.

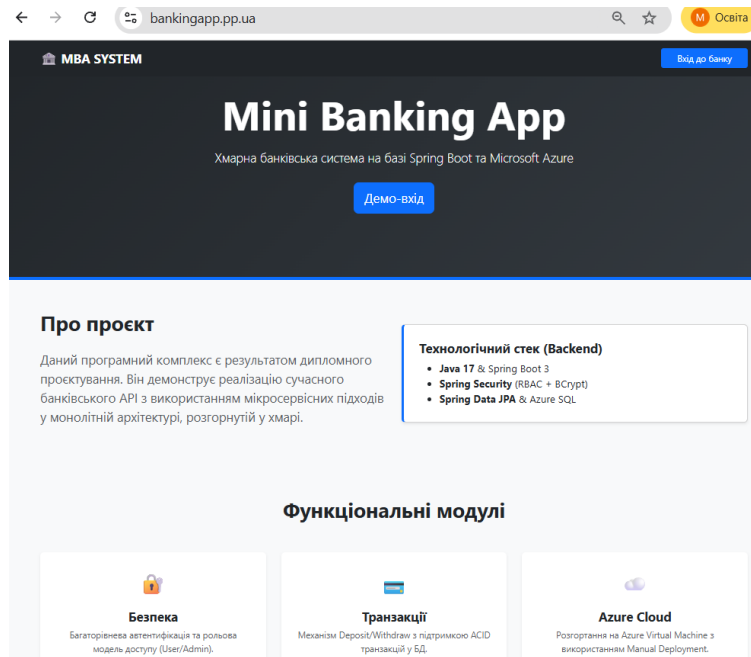


Рисунок 2.9 – Ознайомча сторінка вебдодатку

Функціональні вимоги додатку визначаються його призначенням і охоплюють ключові операції, які користувачі можуть виконувати у системі.

По-перше, додаток забезпечує управління користувачами, що включає реєстрацію, автентифікацію та авторизацію з розмежуванням ролей: звичайний користувач (USER) має доступ лише до власних рахунків та операцій (див. рис. 2.10 , 2.12 та 2.13), тоді як адміністратор (ADMIN) контролює всю систему, включно з управлінням користувачами та переглядом загальної статистики (див. рис. 2.11, 2.14 та 2.15).

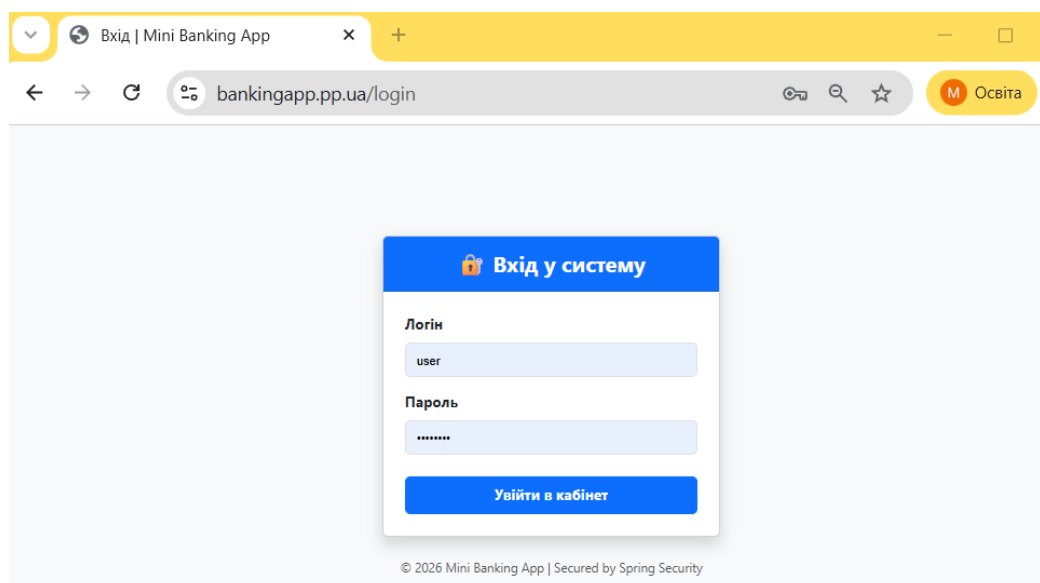


Рисунок 2.10 – Автентифікація звичайного користувача

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				49
Змн.	Арк.	№ докум.	Підпис	Дата		

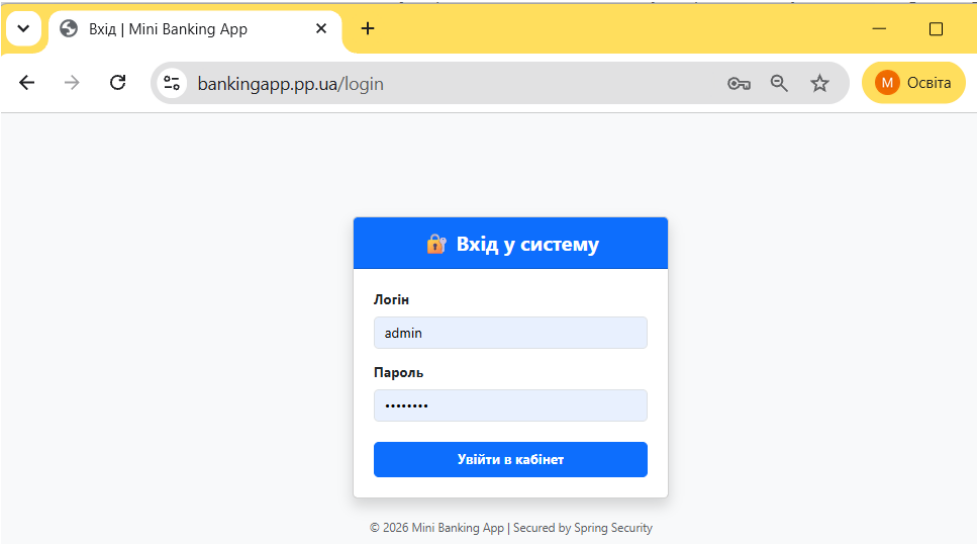


Рисунок 2.11 – Автентифікація адміністратора

По-друге, додаток реалізує роботу з банківськими рахунками, що включає перегляд балансу, історії транзакцій, виконання депозитів, зняття коштів та перекази між рахунками. Цей функціонал покликаний демонструвати роботу бізнес-логіки, включно з перевіркою доступу та дотриманням правил безпеки при виконанні фінансових операцій.

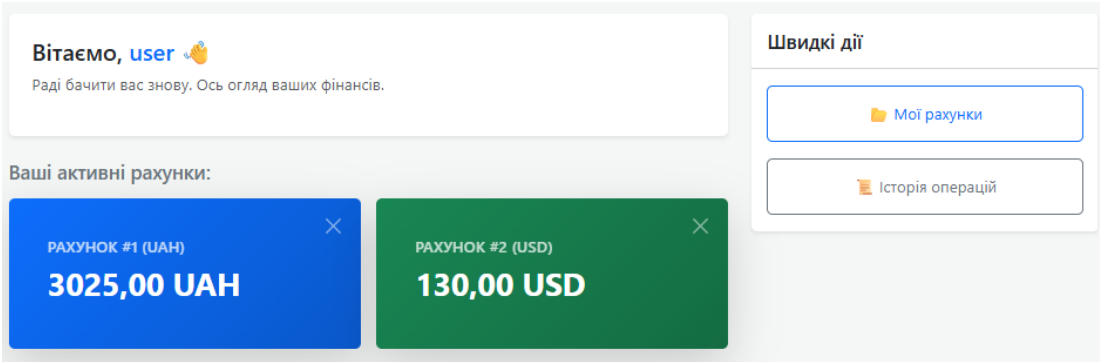


Рисунок 2.12 – Вигляд головної сторінки користувача

MBA BANK Головна Рахунки Транзакції				Вийти
Історія транзакцій				Друк виписки
Дата та час	Тип операції	Рахунок	Сума	
05.06.2026 07:35	Зняття	#5 РАХУНОК #2 (USD)	-30,00	USD
05.06.2026 07:34	Поповнення	#5 РАХУНОК #2 (USD)	+160,00	USD
05.06.2026 07:33	Зняття	#4 РАХУНОК #1 (UAH)	-175,00	UAH
05.06.2026 07:33	Поповнення	#4 РАХУНОК #1 (UAH)	+200,00	UAH
05.06.2026 07:31	Поповнення	#4 РАХУНОК #1 (UAH)	+3000,00	UAH

Рисунок 2.13 – Перелік транзакції користувача

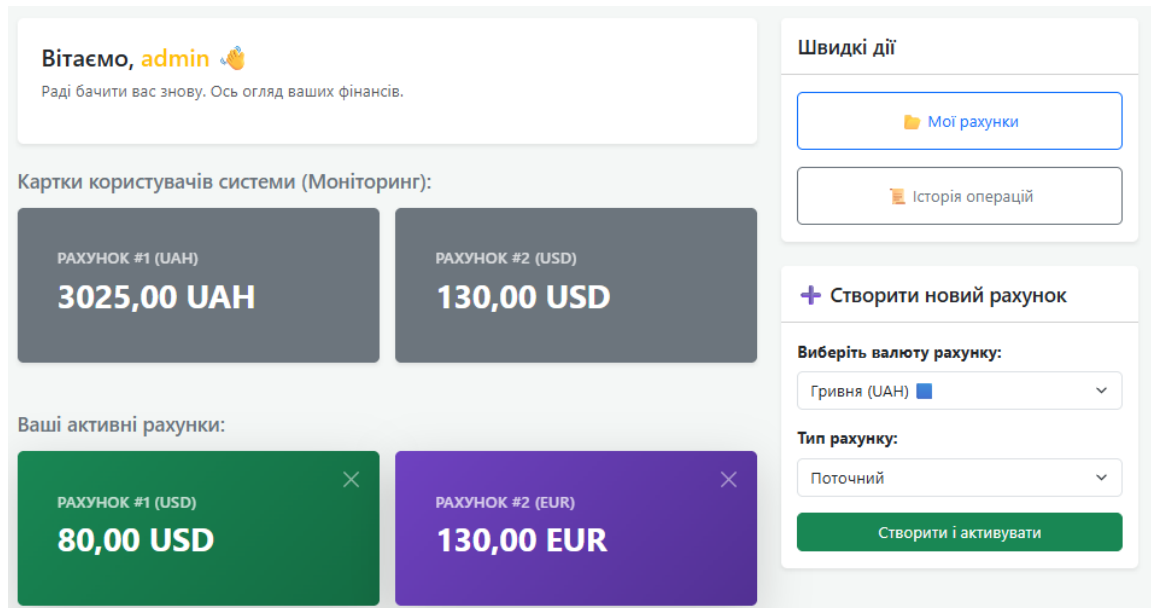


Рисунок 2.14 – Вигляд головної сторінки адміністратора

MBA BANK Головна Рахунки Транзакції Адмін-панель Вийти				
Історія транзакцій Друк виписки				
Дата та час	Тип операції	Рахунок	Користувач	Сума
05.06.2026 07:35	Зняття	#5 РАХУНОК #2 (USD)	user	-30,00 USD
05.06.2026 07:34	Поповнення	#5 РАХУНОК #2 (USD)	user	+160,00 USD
05.06.2026 07:33	Зняття	#4 РАХУНОК #1 (UAH)	user	-175,00 UAH
05.06.2026 07:33	Поповнення	#4 РАХУНОК #1 (UAH)	user	+200,00 UAH
05.06.2026 07:32	Зняття	#2 РАХУНОК #2 (EUR)	admin	-70,00 EUR
05.06.2026 07:31	Поповнення	#4 РАХУНОК #1 (UAH)	user	+3000,00 UAH
05.06.2026 07:28	Поповнення	#2 РАХУНОК #2 (EUR)	admin	+200,00 EUR
05.06.2026 07:26	Зняття	#1 РАХУНОК #1 (USD)	admin	-20,00 USD
05.06.2026 07:25	Поповнення	#1 РАХУНОК #1 (USD)	admin	+100,00 USD

Рисунок 2.15 – Перелік транзакції адміністратора

По-третє, додаток підтримує роботу з файловим сервісом, який дозволяє користувачам завантажувати та зберігати документи, такі як виписки. Доступ до файлів регламентується ролями користувачів, що забезпечує конфіденційність та захист персональних даних (рис. 2.16).

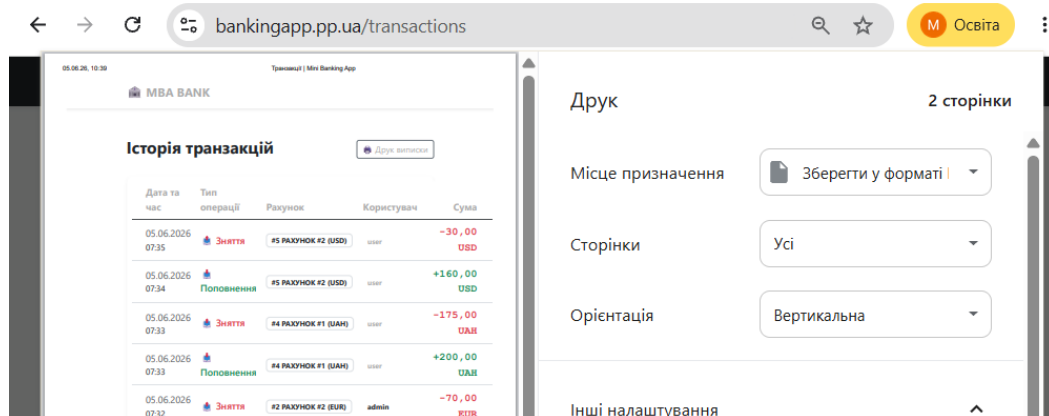


Рисунок 2.16 – Огляд структури параметру «Друк виписки»

Крім того, Mini Banking App передбачає інтерфейс користувача у вигляді вебсторінок для логіну, панелі користувача та адміністративної панелі, що дозволяє зручно управляти рахунками, переглядати транзакції та працювати з документами (рис. 2.17). Використання статичних ресурсів та HTML-шаблонів через Thymeleaf забезпечує динамічне відображення інформації та сучасний інтерфейс користувача.

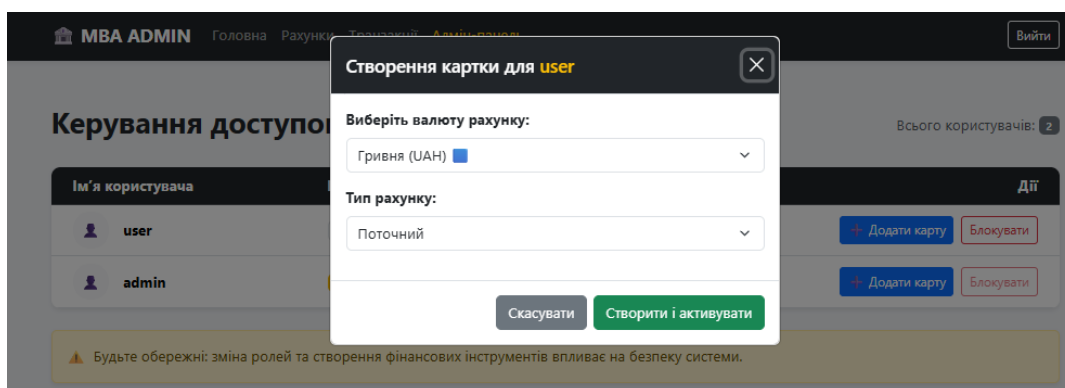


Рисунок 2.17 – Процес створення картки для користувача в «Адмін-панелі»

Особливу увагу приділено безпеці та інтеграції з хмарною інфраструктурою Azure. Додаток підключається до бази даних Azure SQL за допомогою User-Assigned Managed Identity (див. рис. 2.18).

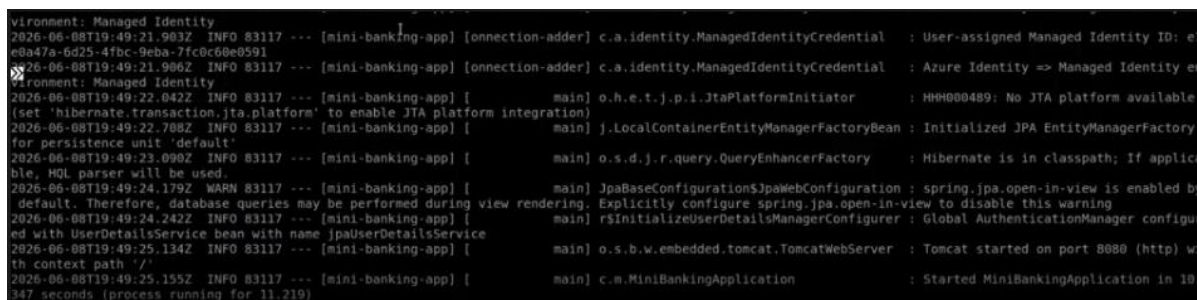


Рисунок 2.18 – Логи ініціалізації Spring Boot додатка із відображенням автентифікації в Azure SQL через User-Assigned Managed Identity

Функціонал закриття картки відрізняється залежно від ролі користувача. Для користувача системи передбачено обов'язкове введення причини закриття картки, що забезпечує фіксацію підстав виконання операції (рис. 2.19). Після завершення процедури система генерує та надсилає електронне повідомлення адміністратору з інформацією про факт закриття картки та зазначену причину (рис. 2.20).

Рисунок 2.19 – Процедура закриття рахунку у користувача

Запит на закриття рахунку: РАХУНОК #2 (USD) Вхідні x

 **kb221_kmo@student.ztu.edu.ua**
кому мені ▼

Повідомлення від системи безпеки MBA BANK.

Користувач закрав свій рахунок.
Інформація про рахунок: РАХУНОК #2 (USD)
Офіційна причина закриття: Не користуюся цим рахунком

Рисунок 2.20 – Зміст листа про закриття рахунку

Для адміністратора механізм закриття картки реалізовано через впливаюче вікно підтвердження в браузері. Перед виконанням операції система відображає запит «Ви впевнені, що хочете видалити цей рахунок?», після підтвердження якого виконується закриття картки (рис. 2.21).

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				53
Змн.	Арк.	№ докум.	Підпис	Дата		

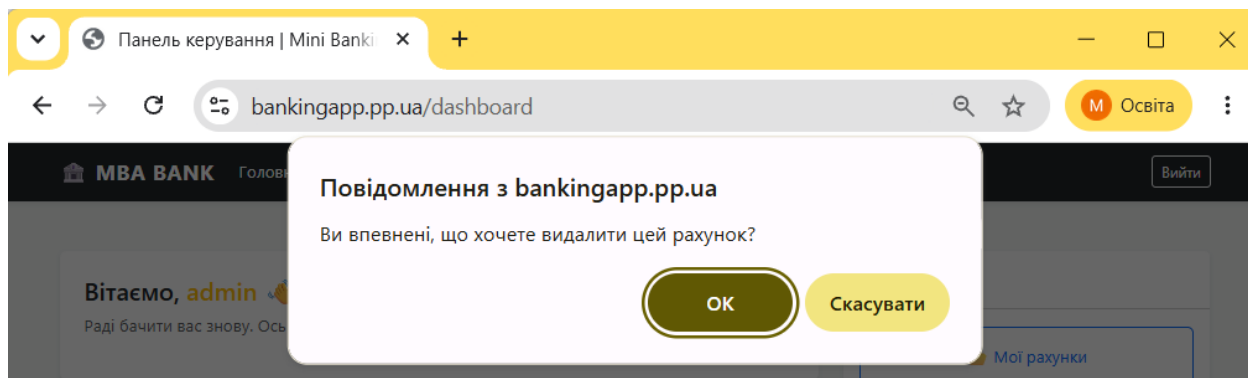


Рисунок 2.21 – Процедура закриття рахунку в адміністратора

Використання Private Endpoints гарантує, що критично важливий трафік між вебсервером додатка та Azure SQL Database повністю ізольований і проходить виключно всередині приватної віртуальної мережі vnet-banking, минаючи публічний Інтернет. Водночас зовнішній периметр додатка захищено на багаторівневій основі:

- WAF забезпечує проактивний захист прикладного рівня (L7) від специфічних веб-атак, таких як SQL-ін'єкції, XSS та інші вразливості з переліку OWASP Top 10;
- Azure DDoS Protection функціонує на мережному та транспортному рівнях, автоматично моніторячи вхідний трафік та блокуючи спроби зломисників порушити доступність сервісу bankingapp.pp.ua шляхом перевантаження каналів зв'язку.

Таким чином, Mini Banking App є комплексним демонстраційним додатком, який поєднує функціональні банківські можливості, безпечну роботу з даними та інтеграцію з хмарними сервісами, що робить його ефективним прикладом для вивчення сучасних підходів до побудови безпечної хмарної архітектури.

2.3.2. Структура проєкту

Вебдодаток побудований на основі Spring Boot, що забезпечує модульність, масштабованість та безпечну інтеграцію з Azure. Проєкт має багаторівневу структуру, де кожен шар відповідає за HTTP-запити, бізнес-логіку, доступ до бази даних та безпеку.

Використання чітких пакетів дозволяє розмежувати функціональні області та спростити тестування. YAML-конфігурації та профілі середовища забезпечують

гнучке розгортання у локальному та продуктивному середовищах з Managed Identity.

Структура включає статичні ресурси, HTML-шаблони та контейнеризацію, що робить додаток готовим до cloud-native розгортання (див. рис. 2.22).

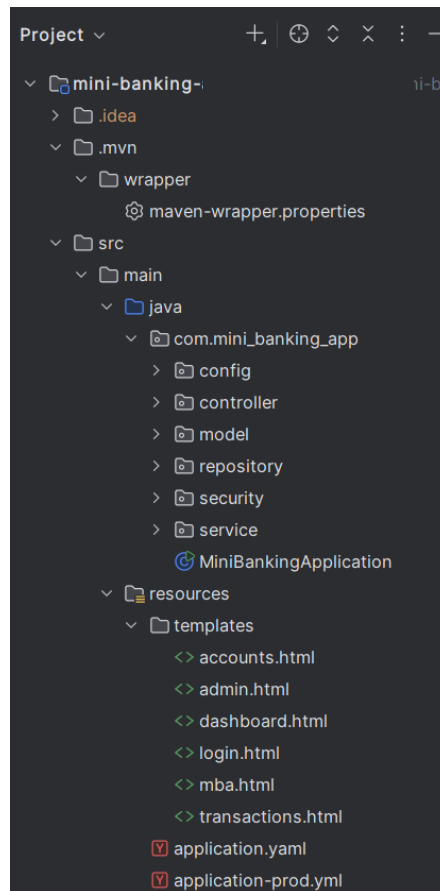


Рисунок 2.22 – Архітектура проекту

src/main/java/com/mini_banking_app/ - головна директорія Java-класів додатку, що містить пакетну структуру згідно з принципами організації коду.

controller/ - містить REST-контролери, які обробляють HTTP-запити користувачів та реалізують взаємодію з сервісним шаром.

service/ - містить бізнес-логіку додатку, включно з обробкою транзакцій, валідацією та управлінням ролями.

repository/ - містить JPA репозиторії для роботи з базою даних, реалізуючи доступ до сутностей.

model/ - містить класові сутності, які відображають структуру даних додатку, наприклад User, Account, Transaction.

security/ - реалізація механізмів безпеки, включно з автентифікацією, авторизацією та налаштуванням ролей.

config/ - містить конфігураційні класи для налаштування підключення до бази даних, Azure Managed Identity, а також CORS та інших параметрів.

src/main/resources/ - директорія ресурсів, що включає конфігураційні файли та статичні ресурси.

application.yml - основний файл конфігурації додатку для середовища prod, містить параметри підключення до бази даних, профілі середовища.

application-prod.yml - специфічні конфігурації для середовищ розробки та продакшена.

templates/ - містить серверні HTML-шаблони, які можуть відображати динамічні дані користувача, наприклад сторінки логіну, панель користувача або історію транзакцій.

2.3.3. Залежності, профілі середовища та вимоги до підключення до бази даних

Для забезпечення коректної роботи Java-додатку Mini Banking App у різних середовищах виконання та його безпечної інтеграції з хмарними сервісами Azure було визначено набір необхідних програмних залежностей, механізм профілів середовища та вимоги до підключення до бази даних. Вибір залежностей здійснювався з урахуванням вимог до масштабованості, безпеки та відповідності рекомендаціям Azure Well-Architected Framework.

У якості основи використовується фреймворк Spring Boot, який забезпечує швидке розгортання додатку та підтримку модульної архітектури. Для реалізації REST-взаємодії застосовується *spring-boot-starter-web*, а для доступу до реляційної бази даних — *spring-boot-starter-data-jpa* у поєднанні з JDBC-драйвером Microsoft SQL Server. Реалізація механізмів автентифікації та авторизації забезпечується за допомогою *spring-boot-starter-security*. Для інтеграції з хмарними сервісами Azure та реалізації безпечної автентифікації без використання облікових даних у коді додатку використовується бібліотека Azure Identity SDK (рис. 2.23 та 2.24).

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				
Змн.	Арк.	№ докум.	Підпис	Дата		56

Project

☐ Gradle - Groovy
 ☐ Gradle - Kotlin
 ☒ Maven

Language

☒ Java
 ☐ Kotlin
 ☐ Groovy

Spring Boot

☐ 4.0.1 (SNAPSHOT)
 ☐ 4.0.0
 ☐ 3.5.10 (SNAPSHOT)
 ☒ 3.5.9

Project Metadata

Group

com

Artifact

mini-banking-app

Name

mini-banking-app

Description

Diploma Bank App

Package name

com.mini-banking-app

Packaging

☒ Jar
 ☐ War

Configuration

☐ Properties
 ☒ YAML

Java

☐ 25
 ☐ 21
 ☒ 17

Dependencies

ADD DEPENDENCIES... CTRL + B

Spring Web

WEB

Build web, including RESTful, applications using Spring MVC. Uses Apache Tomcat as the default embedded container.

Spring Data JPA

SQL

Persist data in SQL stores with Java Persistence API using Spring Data and Hibernate.

Spring Security

SECURITY

Highly customizable authentication and access-control framework for Spring applications.

Validation

IO

Bean Validation with Hibernate validator.

MS SQL Server Driver

SQL

A JDBC and R2DBC driver that provides access to Microsoft SQL Server and Azure SQL Database from any Java application.

Azure Storage

MICROSOFT AZURE

All Storage features are supported, e.g. blob, fileshare and queue.

Рисунок 2.23 – Конфігурація фреймворку Spring Boot

```

<dependency>
  <groupId>com.microsoft.sqlserver</groupId>
  <artifactId>mssql-jdbc</artifactId>
  <version>13.2.1.jre11</version>
  <scope>compile</scope>
</dependency>
<dependency>
  <groupId>com.azure</groupId>
  <artifactId>azure-identity</artifactId>
  <version>1.18.1</version>
  <scope>compile</scope>
</dependency>
<dependency>
  <groupId>com.azure</groupId>
  <artifactId>azure-storage-blob</artifactId>
  <version>12.32.0</version>
</dependency>

```

Рисунок 2.24 – Додавання Azure-залежностей

Базові налаштування визначаються у файлі *application.yml* (рис.2.25), тоді як специфічні параметри середовища - у файлі *application-prod.yml* (рис.2.26). Активізація необхідного профілю здійснюється під час запуску додатку шляхом передачі параметра JVM - *Dspring.profiles.active* (рис. 2.27), що дозволяє змінювати конфігурацію без внесення змін у програмний код.

```

1  spring:
2    application:
3      name: mini-banking-app
4
5    profiles:
6      active: prod
7
8    datasource:
9      url: jdbc:sqlserver://sql-banking-24627.database.windows.net:1433;
10     driver-class-name: com.microsoft.sqlserver.jdbc.SQLServerDriver
11
12    jpa:
13      hibernate:
14        ddl-auto: update
15      show-sql: true
16      properties:
17        hibernate:
18          format_sql: true
19          dialect: org.hibernate.dialect.SQLServerDialect
20
21    thymeleaf:
22      prefix: classpath:/templates/
23      suffix: .html
24      cache: false

```

Рисунок 2.25 – Базові налаштування у файлі application.yml

```

1  spring:
2    datasource:
3      url: jdbc:sqlserver://sql-banking-24627.database.windows.net:1433;
4      username: eb9bd308-2a36-409b-8985-b6f6ad8c41ea
5      driver-class-name: com.microsoft.sqlserver.jdbc.SQLServerDriver
6
7    jpa:
8      database-platform: org.hibernate.dialect.SQLServerDialect
9      hibernate:
10        ddl-auto: update
11        show-sql: false

```

Рисунок 2.26 – Лістинг файлу application-prod.yml

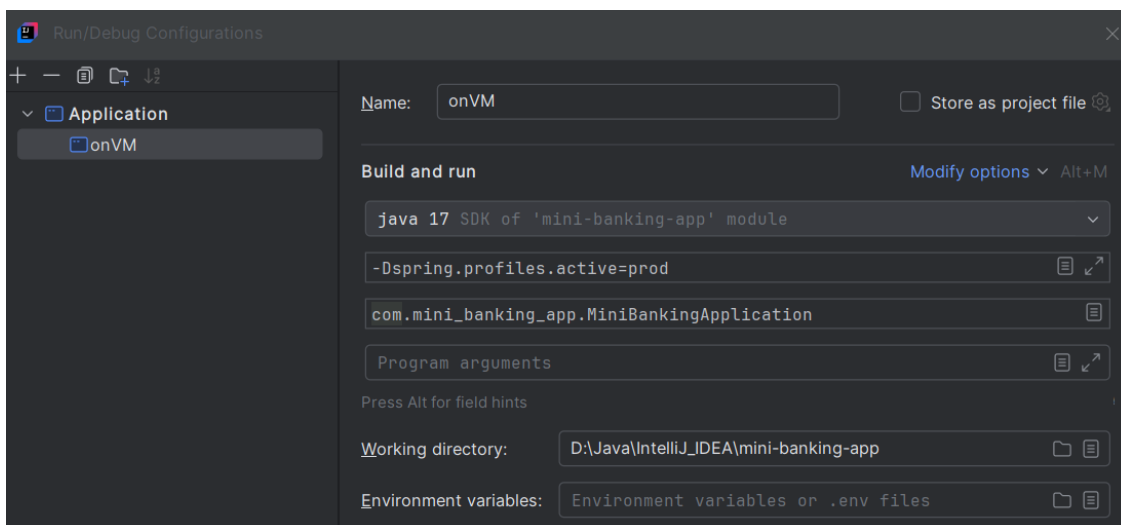


Рисунок 2.27 – Параметри для запуску проекту на віртуальній машині

Підключення до бази даних у production середовищі реалізовано з використанням Azure SQL Database та відповідає сучасним вимогам безпеки. Додаток підключається до бази через логічний SQL Server, ідентифікований доменним ім'ям *sql-banking-20870.database.windows.net*, із зазначенням назви конкретної бази даних у параметрах підключення. Для забезпечення конфіденційності даних передача інформації здійснюється з використанням шифрування TLS 1.2/1.3. Доступ до бази даних реалізовано за допомогою Managed Identity, що дозволяє уникнути зберігання логінів і паролів у конфігураційних файлах та забезпечує централізоване управління доступом через Entra ID.

Крім того, доступ до бази даних обмежено мережею за допомогою Private Endpoint, що гарантує відсутність публічного доступу до сервісу та забезпечує взаємодію додатку з базою даних виключно в межах приватної віртуальної мережі та довірених публічних IP-адрес (рис. 2.28).

Такий підхід підвищує загальний рівень безпеки системи та відповідає рекомендованим практикам побудови захищених хмарних рішень на платформі Azure.

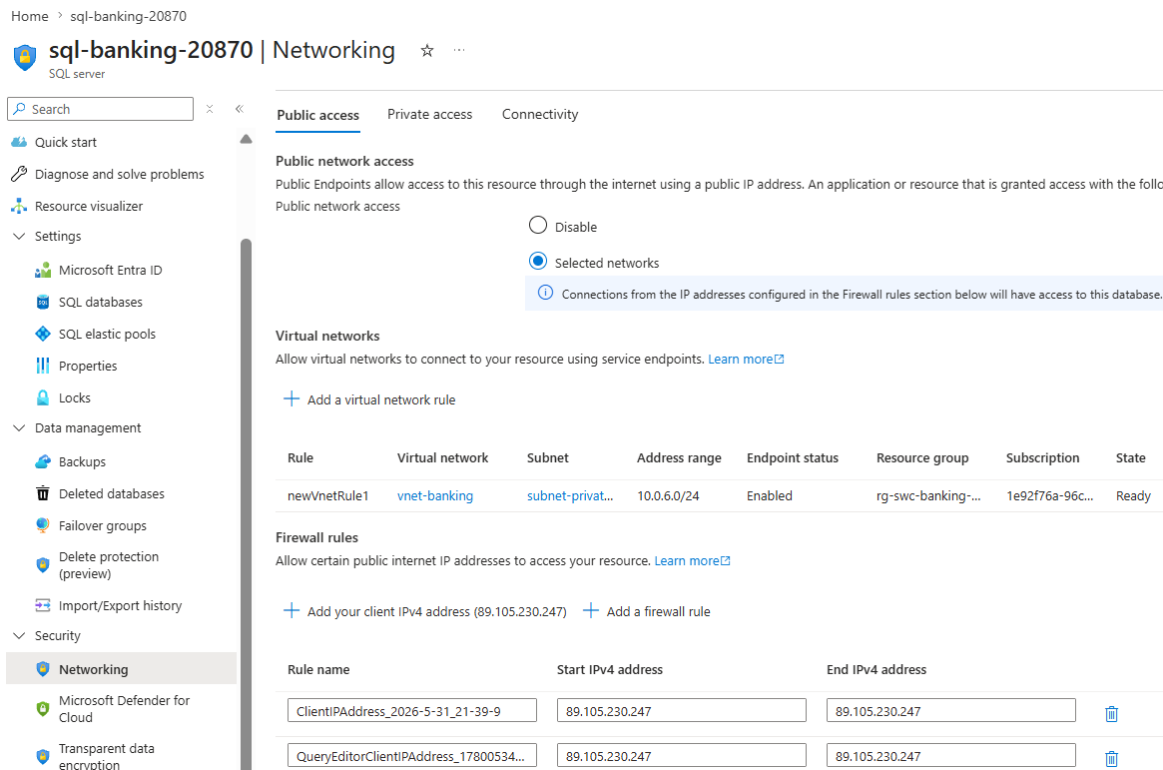


Рисунок 2.28 – Обмеження публічного доступу до бази даних на рівні Azure SQL Firewall

2.3.4. Безпекові вимоги додатку

Проектування та реалізація Java-додатку Mini Banking App здійснювались з урахуванням сучасних вимог до інформаційної безпеки, зокрема принципів *secure by design* та *defense in depth*. Оскільки додаток оперує фінансовими даними користувачів, до нього висувуються підвищені вимоги щодо конфіденційності, цілісності та доступності інформації. Безпекові механізми реалізовано як на рівні програмного забезпечення, так і шляхом інтеграції з керованими сервісами платформи Microsoft Azure.

Однією з ключових вимог є забезпечення надійної автентифікації та авторизації користувачів. Для цього у додатку використовується фреймворк Spring Security, який дозволяє реалізувати централізований контроль доступу до REST-ендпоінтів. Адміністративні функції обмежені доступом тільки для користувачів з роллю ADMIN, тоді як решта функціоналу доступна всім авторизованим користувачам. Такий підхід дозволяє реалізувати принцип мінімальних привілеїв та розмежувати права доступу до критичних ресурсів системи (рис. 2.29).

```
3 import org.springframework.context.annotation.Bean;
4 import org.springframework.context.annotation.Configuration;
5 import org.springframework.security.config.annotation.web.builders.HttpSecurity;
6 import org.springframework.security.config.annotation.web.configurers.AbstractHttpConfigurer;
7 import org.springframework.security.web.SecurityFilterChain;
8
9 import static org.springframework.security.config.Customizer.withDefaults;
10
11 @Configuration 1 usage
12 public class SecurityConfig {
13
14     @Bean no usages
15     @
16     public SecurityFilterChain securityFilterChain(HttpSecurity http) throws Exception {
17         http
18             .authorizeHttpRequests( AuthorizationManagerRequestMat... auth -> auth
19                 .requestMatchers( ...patterns: "/admin/**").hasRole("ADMIN")
20                 .anyRequest().authenticated()
21             )
22             .formLogin(withDefaults())
23             .csrf(AbstractHttpConfigurer::disable);
24
25         return http.build();
26     }
27 }
```

Рисунок 2.29 – Реалізація централізованого контроль доступу до REST-endpoint

Важливою безпековою вимогою є захист конфігураційних даних та облікових даних доступу до зовнішніх ресурсів. У додатку повністю відсутнє зберігання логінів, паролів або ключів доступу у вихідному коді та конфігураційних файлах.

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				60
Змн.	Арк.	№ докум.	Підпис	Дата		

Для автентифікації під час підключення до сервісу Azure SQL Database, використовується Managed Identity. Це дозволяє здійснювати ідентифікацію додатку через Entra ID без необхідності керування секретами, що суттєво зменшує ризик їх витоку та спрощує адміністрування доступу.

Проектом передбачено, що передача даних між усіма вузлами системи має здійснюватися виключно через захищені канали зв'язку. Взаємодія зовнішніх клієнтів із прикладним рівнем базується на протоколі HTTPS із застосуванням сучасних криптографічних протоколів TLS версії 1.2 або вище. Це гарантує захист інформації від атак MITM та забезпечує цілісність пакетів під час транспортування.

На рівні внутрішньої інфраструктури доступ до бази даних Azure SQL проектується із застосуванням механізму примусового шифрування з'єднань. Для посилення мережної ізоляції обрано технологію Azure Private Endpoint, яка дозволяє виключити базу даних із публічного адресного простору мережі Інтернет, спрямовуючи весь трафік через приватний тунель у межах віртуальної мережі.

При проектуванні архітектури Spring Boot додатку закладено механізми протидії критичним загрозам згідно з переліком OWASP Top 10. Основними методами нейтралізації ризиків обрано:

- захист від ін'єкцій: використання Spring Data JPA та параметризованих запитів на рівні репозиторіїв, що унеможливорює експлуатацію SQL-ін'єкцій.
- валідація вхідних даних: застосування декларативної перевірки полів (Bean Validation) для запобігання атакам XSS.
- управління винятками: впровадження централізованого обробника помилок (@ControllerAdvice), що забезпечує надання користувачу лише узагальнених повідомлень без розкриття внутрішньої структури коду або БД.

Ешелонований захист за допомогою хмарних сервісів Додатковий рівень безпеки проектується через інтеграцію додатка з сервісом Azure Application Gateway. Завдяки активації компонента WAF, система отримує можливість аналізувати HTTP-пакети в реальному часі. Проектне рішення передбачає фільтрацію трафіку на мережному периметрі, що дозволяє нейтралізувати шкідливі запити (наприклад, спроби обходу каталогів або brute-force) ще до того, як вони досягнуть обчислювальних ресурсів віртуальної машини.

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				
Змн.	Арк.	№ докум.	Підпис	Дата		61

2.3.5. Використання Managed Identity - архітектурні рішення

У межах проєктування хмарної інфраструктури для вебдодатку Mini Banking App особливу увагу приділено забезпеченню безпечної автентифікації та авторизації між прикладним рівнем та керованими сервісами платформи Azure. Для цього в архітектурі рішення використано механізм Azure Managed Identity, який дозволяє ресурсам автентифікуватися в інших сервісах без необхідності зберігання облікових даних, секретів або ключів доступу в конфігураційних файлах додатку.

Spring Boot додаток, розгорнутий на Azure Virtual Machine, використовує системно призначену керовану ідентичність для доступу до Azure SQL Database, який підключений через Private Endpoint. Процес автентифікації реалізується за допомогою Entra ID, що видає тимчасові маркери доступу на основі ролей, призначених відповідному ресурсу. Таким чином, додаток отримує доступ лише до необхідних сервісів і виключно в межах визначених прав.

Застосування Managed Identity дозволяє повністю усунути потребу у використанні статичних облікових даних, таких як імена користувачів, паролі або connection strings з вбудованими секретами. Це значно зменшує ризики компрометації конфіденційної інформації та відповідає сучасним принципам безпеки. Крім того, ротація токенів доступу виконується автоматично на рівні платформи Azure, що усуває необхідність ручного керування ключами та знижує ймовірність помилок конфігурації.

Висновок до другого розділу

У другому розділі спроектовано безпечну хмарну інфраструктуру Microsoft Azure та розроблено архітектурні рішення для Spring Boot додатка. Створено схему адресації віртуальної мережі vnet-banking з ізольованими підмережами для Azure Application Gateway, вебсервера Nginx та систем Azure Bastion. Мережеву безпеку оптимізовано за NSG, які блокують прямий доступ з Інтернету, та шлюзів Private Endpoints для захисту внутрішньої взаємодії. У програмній частині визначено структуру та залежності Java-додатка, а головним безпековим рішенням стало впровадження безпарольної автентифікації до бази даних Azure SQL через User-Assigned

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				62
Змн.	Арк.	№ докум.	Підпис	Дата		

Managed Identity. Налаштована система централізованого моніторингу та журналювання подій формує надійну базу для фінального розгортання проекту.

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				63
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 3. ПРАКТИЧНА РЕАЛІЗАЦІЯ ТА РОЗГОРТАННЯ БЕЗПЕЧНОГО ВЕБДОДАТКУ В ХМАРНОМУ СЕРЕДОВИЩІ AZURE

3.1. Підготовка та конфігурування хмарного середовища

3.1.1. Створення та налаштування Virtual Network з виділеними підмережами для Application Gateway та Azure Bastion

Процес реалізації розпочався з автоматизації розгортання інфраструктури. Для мінімізації людського фактору та забезпечення ресурсів було використано bash-скрипт з використанням команд Azure CLI.

Першим кроком стала підготовка робочого середовища в хмарному терміналі Azure Cloud Shell (рис.3.1).

```
kb221_diploma3_kmo [ ~ ]$ nano deploy.sh
kb221_diploma3_kmo [ ~ ]$ chmod +x deploy.sh
kb221_diploma3_kmo [ ~ ]$ ./deploy.sh
```

Рисунок 3.1 – Створення, надання прав на запуск та запуск скрипту deploy.sh

Паралельно з підготовкою скриптів розгортання було проведено аналіз доступних конфігурацій та образів віртуальних машин в swedencentral регіоні. Це необхідно для забезпечення оптимального балансу між вартістю хмарних ресурсів у межах лімітів студентської підписки та продуктивністю, що потрібна для стабільної роботи вебсервера Nginx та Spring Boot додатка. За результатами моніторингу доступних варіантів для реалізації проєкту було обрано конфігурацію Standard_B2als_v2 (2 vCPU, 4 GiB RAM), яка повністю задовольняє вимоги тестового середовища (рис. 3.2).

4	4096	Standard_B2als_v2	2	1047552	0
---	------	-------------------	---	---------	---

Рисунок 3.2 – Вибір образу Standard_B2als_v2 у регіоні swedencentral

Скрипт deploy.sh було розбито на блоки коду, які виконуються послідовно. Але з метою логічного групування типів ресурсів, результати виконання блоків скрипту наведені у звіті за логічною схемою.

У першому блоці коду використовується команда `az group create` для створення групи ресурсів. Результат відпрацювання цього блоку представлено на рисунку 3.3.

```

=== 1. Створення групи ресурсів ===
{
  "id": "/subscriptions/1e92f76a-96c0-4261-9d65-0ec110f2bb44/resourceGroups/rg-swc-banking-app",
  "location": "swedencentral",
  "managedBy": null,
  "name": "rg-swc-banking-app",
  "properties": {
    "provisioningState": "Succeeded"
  },
  "tags": null,
  "type": "Microsoft.Resources/resourceGroups"
}

```

Рисунок 3.3 – Результат виконання першого блоку коду для створення RG

Після створення RG rg-swc-banking-app необхідно було створити ізольований мережний простір. Для створення віртуальної мережі було використано команду `az network vnet create`. Результат відпрацювання команди відображено на рисунку 3.4.

```

=== 2. Створення віртуальної мережі ===
{
  "newVnet": {
    "addressSpace": {
      "addressPrefixes": [
        "10.0.0.0/16"
      ]
    },
    "enableDdosProtection": false,
    "etag": "W/\"70882c67-61f3-47b7-91c0-5afd7b0854a7\"",
    "id": "/subscriptions/1e92f76a-96c0-4261-9d65-0ec110f2bb44/resourceGroups/rg-swc-banking-app/providers/Microsoft.Network/virtualNetworks/vnet-banking",
    "location": "swedencentral",
    "name": "vnet-banking",
    "privateEndpointVnetPolicies": "Disabled",
    "provisioningState": "Succeeded",
    "resourceGroup": "rg-swc-banking-app",
    "resourceGuid": "72413ef8-123b-4c83-9566-cf6ba776c24a",
    "subnets": [],
    "type": "Microsoft.Network/virtualNetworks",
    "virtualNetworkPeerings": []
  }
}

```

Рисунок 3.4 – Результат виконання першого блоку коду для створення VNet

Віртуальна мережа vnet-banking з адресним простором 10.0.0.0 /16 була розподілена на підмережі.

В третьому блоці містяться команди `az network vnet subnet create` з різними параметрами для створення підмереж. Першою була створена підмережа subnet-app з адресним простором 10.0.1.0/28, яка буде використовуватись для Application Gateway. Результат відпрацювання команди відображено на рисунку 3.5.

```

=== 3. Створення підмереж ===
{
  "addressPrefix": "10.0.1.0/28",
  "delegations": [],
  "etag": "W/\"ed8a107f-1d74-4c40-a93b-7d8e21955a1d\"",
  "id": "/subscriptions/1e92f76a-96c0-4261-9d65-0ec110f2bb44/resourceGroups/rg-swc-banking-app/providers/Microsoft.Network/virtualNetworks/vnet-banking/subnets/subnet-app",
  "name": "subnet-app",
  "privateEndpointNetworkPolicies": "Disabled",
  "privateLinkServiceNetworkPolicies": "Enabled",
  "provisioningState": "Succeeded",
  "resourceGroup": "rg-swc-banking-app",
  "type": "Microsoft.Network/virtualNetworks/subnets"
}

```

Рисунок 3.5 – Результат виконання блоку коду для створення Subnet subnet-app

Другою була створена subnet-vm з адресним простором 10.0.2.0 /28, яка буде використовуватись безпосередньо для віртуальної машини. Результат відпрацювання команди відображено на рисунку 3.6.

```
{
  "addressPrefix": "10.0.2.0/28",
  "delegations": [],
  "etag": "W/\"085ef100-46da-4d80-9544-2dbb06aa8839\"",
  "id": "/subscriptions/1e92f76a-96c0-4261-9d65-0ec110f2bb44/resourceGroups/rg-swc-banking-app/providers/Microsoft.Network/virtualNetworks/vnet-banking/subnets/subnet-vm",
  "name": "subnet-vm",
  "privateEndpointNetworkPolicies": "Disabled",
  "privateLinkServiceNetworkPolicies": "Enabled",
  "provisioningState": "Succeeded",
  "resourceGroup": "rg-swc-banking-app",
  "type": "Microsoft.Network/virtualNetworks/subnets"
}
```

Рисунок 3.6 – Результат виконання блоку коду для створення Subnet subnet-vm

Третьою було створена підмережа AzureBastionSubnet з адресним простіром 10.0.4.0 /27. Дана службова підмережа використовується для Azure Bastion та обов’язково має мати назву AzureBastionSubnet. Результат відпрацювання команди відображено на рисунку 3.7.

```
{
  "addressPrefix": "10.0.4.0/27",
  "delegations": [],
  "etag": "W/\"f3d9b187-e573-4739-aa1-731826d75572\"",
  "id": "/subscriptions/1e92f76a-96c0-4261-9d65-0ec110f2bb44/resourceGroups/rg-swc-banking-app/providers/Microsoft.Network/virtualNetworks/vnet-banking/subnets/AzureBastionSubnet",
  "name": "AzureBastionSubnet",
  "privateEndpointNetworkPolicies": "Disabled",
  "privateLinkServiceNetworkPolicies": "Enabled",
  "provisioningState": "Succeeded",
  "resourceGroup": "rg-swc-banking-app",
  "type": "Microsoft.Network/virtualNetworks/subnets"
}
```

Рисунок 3.7 – Результат виконання блоку коду для створення Subnet AzureBastionSubnet

Після створення основних компонентів мережі проводиться верифікація стану ресурсів через Azure CLI або Azure Portal. На рисунку 3.8 наведено результат виконання команди `az network bastion create`. Створений Azure Bastion є точкою входу для адміністрування віртуальною машиною та роботи з нею.

```
=== 6. Створення Azure Bastion ===
{
  "disableCopyPaste": false,
  "dnsName": "bst-050f6c9c-82dd-4575-8d1d-083f7d8bb60f.bastion.azure.com",
  "enableFileCopy": false,
  "enableIpConnect": false,
  "enableKerberos": false,
  "enableSessionRecording": false,
  "enableShareableLink": false,
  "enableTunneling": false,
  "etag": "W/\"87b70e5d-1a68-48bb-a2e1-5220a3d77847\"",
  "id": "/subscriptions/1e92f76a-96c0-4261-9d65-0ec110f2bb44/resourceGroups/rg-swc-banking-app/providers/Microsoft.Network/bastionHosts/bastion",
  "ipConfigurations": [
    {
      "etag": "W/\"87b70e5d-1a68-48bb-a2e1-5220a3d77847\"",
      "id": "/subscriptions/1e92f76a-96c0-4261-9d65-0ec110f2bb44/resourceGroups/rg-swc-banking-app/providers/Microsoft.Network/bastionHosts/bastion/ipconfigurations/bastion_ip_config",
      "name": "bastion_ip_config",
      "privateIPAllocationMethod": "Dynamic",
      "provisioningState": "Succeeded"
    }
  ]
}
```

Рисунок 3.8 – Результат виконання блоку коду для створення Azure Bastion

Для керування політиками мережного доступу для підмереж AzureBastionSubnet та subnet-app їх було асоційовано з окремими NSG (рис. 3.9 - 3.10).

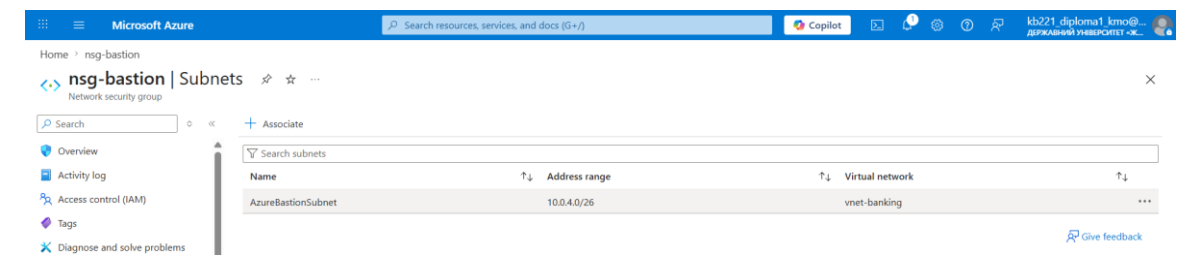


Рисунок 3.9 – Асоціація підмережі AzureBastionSubnet з NSG nsg-bastion

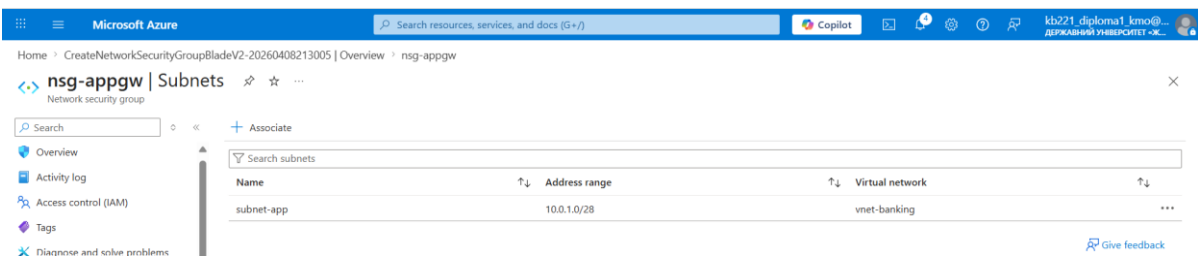


Рисунок 3.10 – Налаштування підмережі subnet-app з NSG nsg-appgw

3.1.2. Розгортання Azure SQL з налаштуванням Private Endpoint

Для створення Azure SQL Server та Azure SQL DB було виконано блоки коду скрипту 7 та 9 з використанням команд az sql server create та az sql db create відповідно. Результат відпрацювання представлено на рисунках 3.11 та 3.12.

```
=== 7. Створення Azure SQL Server ===
{
  "administratorLogin": "sqladminuser",
  "administratorLoginPassword": null,
  "administrators": null,
  "createMode": null,
  "externalGovernanceStatus": "Disabled",
  "federatedClientId": null,
  "fullyQualifiedDomainName": "sql-banking-20870.database.windows.net",
  "id": "/subscriptions/1e92f76a-96c0-4261-9d65-0ec110f2bb44/resourceGroups/rg-swc-banking-app/providers/Microsoft.Sql/servers/sql-banking-20870",
  "identity": null,
  "isIPv6Enabled": null,
  "keyId": null,
  "kind": "v12.0",
  "location": "swedencentral",
  "minimalTlsVersion": "1.2",
  "name": "sql-banking-20870",
  "primaryUserAssignedIdentityId": null,
  "privateEndpointConnections": [],
  "publicNetworkAccess": "Enabled",
  "resourceGroup": "rg-swc-banking-app",
  "restrictOutboundNetworkAccess": "Disabled",
  "retentionDays": -1,
  "state": "Ready",
  "tags": null,
  "type": "Microsoft.Sql/servers",
  "version": "12.0",
  "workspaceFeature": null
}
```

Рисунок 3.11 – Результат виконання блоку коду для створення Azure SQL Server sql-banking-20870

```

=== 9. Створення бази даних ===
{
  "autoPauseDelay": null,
  "availabilityZone": "NoPreference",
  "catalogCollation": "SQL_Latin1_General_CP1_CI_AS",
  "collation": "SQL_Latin1_General_CP1_CI_AS",
  "createMode": null,
  "creationDate": "2026-05-28T14:37:50.867000+00:00",
  "currentBackupStorageRedundancy": "Geo",
  "currentServiceObjectiveName": "Basic",
  "currentSku": {
    "capacity": 5,
    "family": null,
    "name": "Basic",
    "size": null,
    "tier": "Basic"
  },
  "databaseId": "23528735-fb2c-4ffb-a815-2d4c4dbcef51",
  "defaultSecondaryLocation": "swedensouth",

```

Рисунок 3.12 – Результат виконання блоку коду для створення Azure SQL DB db-banking

Наступним кроком налаштування Azure SQL Server була конфігурація Private Endpoint для безпечної комунікації Azure VM з Azure SQL Server. Для реалізації цієї конфігурації було виконано 10 блок коду, який містить команду az network private-endpoint create. Результат спрацювання блоку відображено на рисунку 3.13.

```

{
  "addressPrefix": "10.0.6.0/24",
  "delegations": [],
  "etag": "W/\"bc399cdb-c03a-40fd-aa53-6fb2882ea8c1\"",
  "id": "/subscriptions/1e92f76a-96c0-4261-9d65-8ec110f2bb44/resourceGroups/rg-swc-banking-app/providers/Microsoft.Network/virtualNetworks/vnet-banking/subnets/subnet-private-endpoint",
  "name": "subnet-private-endpoint",
  "privateEndpointNetworkPolicies": "Disabled",
  "privateLinkServiceNetworkPolicies": "Enabled",
  "provisioningState": "Succeeded",
  "resourceGroup": "rg-swc-banking-app",
  "type": "Microsoft.Network/virtualNetworks/subnets"
}

```

Рисунок 3.13 – Результат виконання блоку коду для конфігурації Private Endpoint

3.2. Налаштування безпечного доступу та ідентифікації

3.2.1. Конфігурування Azure Bastion для безпарольного SSH-доступу до Ubuntu Server 24.04.

З метою усунення ризиків, пов'язаних із відкритими портами SSH (22) у публічному просторі, було реалізовано доступ через Azure Bastion. Це дозволяє адмініструвати сервер через захищений SSL-канал без наявності публічної IP-адреси у самої віртуальної машини. На рівні NSG віртуальної машини (vm-diplomaNSG) було налаштовано суворі правила вхідного трафіку (рис.3.14).

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				68
Змн.	Арк.	№ докум.	Підпис	Дата		

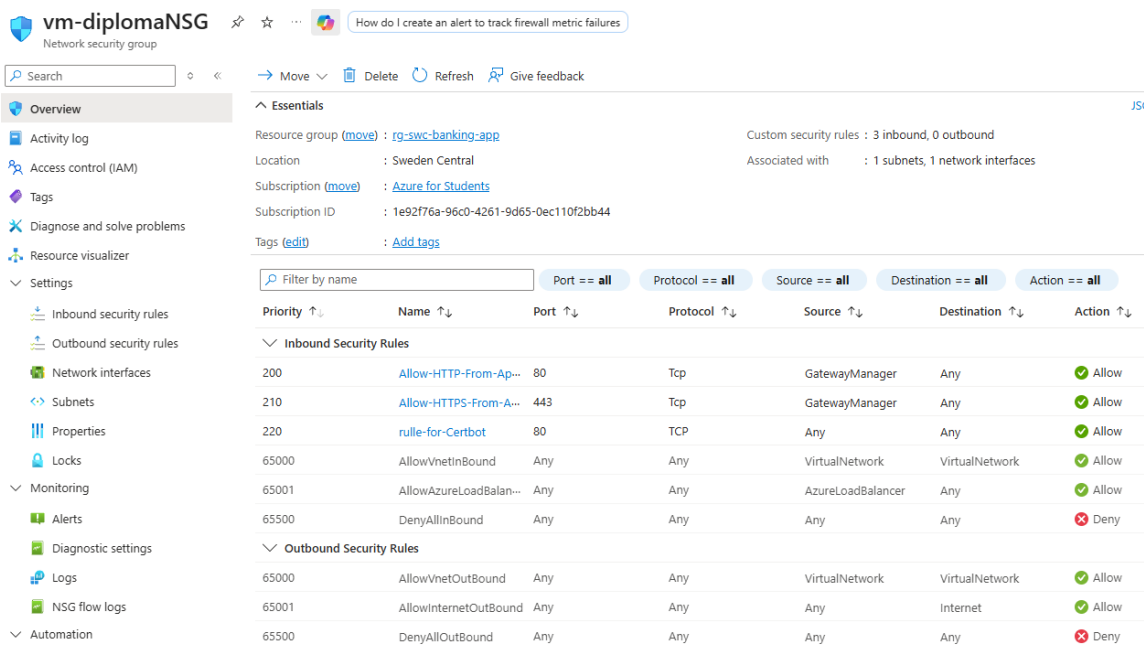


Рисунок 3.14. Конфігурація NSG vm-diplomaNSG

3.2.2. Налаштування Managed Identity для безпечного доступу Azure SQL

Для надання гранульованого доступу віртуальної машини до Azure SQL було створено та призначено vm-diploma Managed Identity до віртуальної машини (рис. 3.15) та до Azure SQL, використовуючи T-SQL запити. Для надання віртуальній машині прав доступу до бази даних було згенеровано унікальний ідентифікатор у межах Entra ID за допомогою Azure CLI.

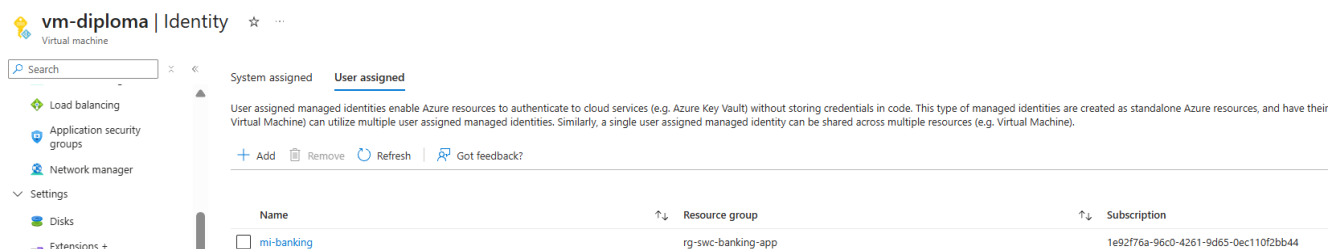


Рисунок 3.15 – User-Assigned Managed Identity призначена до ВМ

Після ізоляції сервера було проведено налаштування бази даних для роботи виключно через механізми хмарної автентифікації. Відмова від SQL-автентифікації на користь Identity-based доступу дозволила усунути вразливості, пов'язані зі зберіганням паролів (див. рис. 3.16).

```

=== 8. Назначения Админа Microsoft Entra ID ===
{
  "administratorType": "ActiveDirectory",
  "azureAdOnlyAuthentication": null,
  "id": "/subscriptions/1e92f76a-96c0-4261-9d65-0ec110f2bb44/resourceGroups/rg-swc-banking-app/providers/Microsoft.Sql/servers/sql-banking-20870/administrators/ActiveDirectory",
  "login": "kb221_diploma3_kmc@student.ztu.edu.ua",
  "name": "ActiveDirectory",
  "resourceGroup": "rg-swc-banking-app",
  "sid": "1e92f76a-96c0-4261-9d65-0ec110f2bb44",
  "tenantId": "d84995f6-c5b5-422d-b123-c1243b2f5125",
  "type": "Microsoft.Sql/servers"
}

```

Рисунок 3.16 – Налаштування Identity-based автентифікації для SQL-сервера

Підключившись до Azure SQL Server було виконано T-SQL команди для надання доступу Managed Identity віртуальної машини до бази даних (рис.3.17).

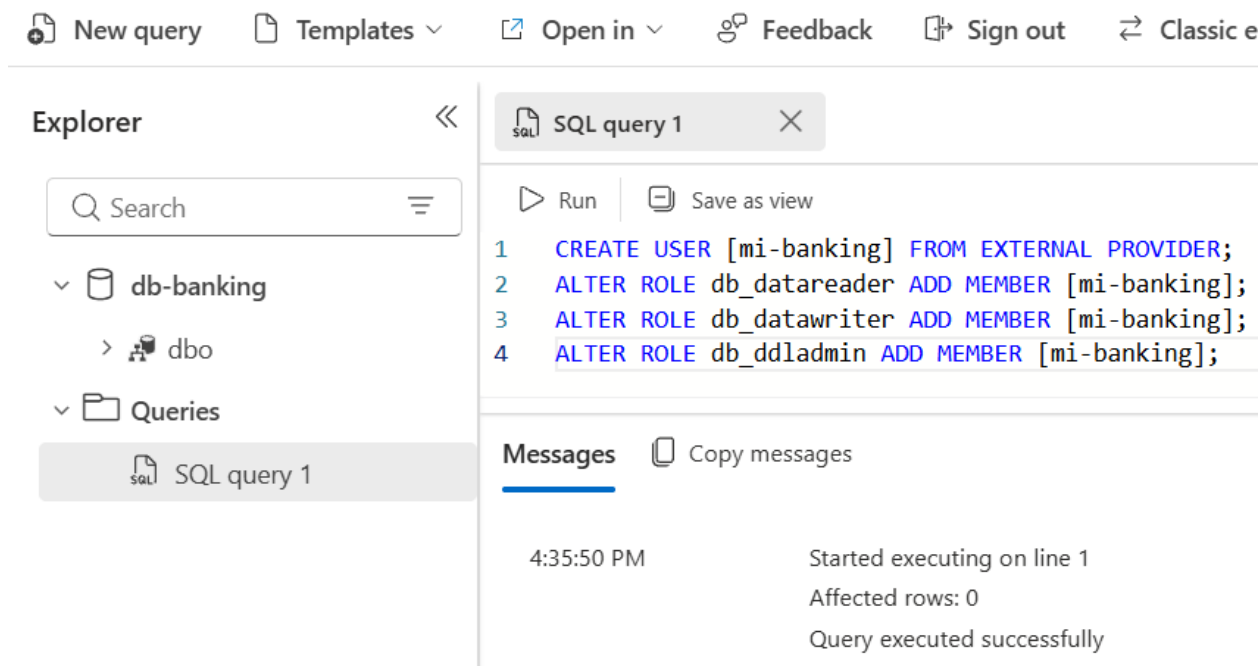


Рисунок 3.17 – Надання Managed Identity доступу до бази даних

3.2.3. Впровадження Azure Application Gateway з активованим WAF для фільтрації вхідного трафіку

Application Gateway виступає в ролі точки входу для користувацького трафіку, забезпечуючи не тільки балансування, а й захист від атак прикладного рівня. NSG Application Gateway nsg-appgw налаштована на прийом стандартного вебтрафіку (HTTP/HTTPS). Крім того, відкрито порти 65200-65535 для сервісу GatewayManager, що необхідно для моніторингу стану шлюзу з боку інфраструктури Azure. Це створює перший ешелон захисту, де весь вхідний трафік аналізується WAF-модулем перед перенаправленням до внутрішньої мережі. На рисунку 3.18 представлені застосовані правила NSG для Application Gateway.

Priority	Name	Port	Protocol	Source	Destination	Action
100	AllowHTTP	80	TCP	Any	Any	Allow
110	AllowHTTPS	443	TCP	Any	Any	Allow
120	AllowGWM	65200-65535	Any	Gateway/Manager	Any	Allow
65000	AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork	Allow
65001	AllowAzureLoadBalancerInBound	Any	Any	AzureLoadBalancer	Any	Allow
65500	DenyAllInBound	Any	Any	Any	Any	Deny
65000	AllowVnetOutBound	Any	Any	VirtualNetwork	VirtualNetwork	Allow
65001	AllowInternetOutBound	Any	Any	Any	Internet	Allow
65500	DenyAllOutBound	Any	Any	Any	Any	Deny

Рисунок 3.18 – Налаштування правил NSG для Application Gateway

Важливим етапом розгортання Application Gateway є його публікація в мережі Інтернет через зрозуміле доменне ім'я з використанням Azure DNS. Налаштування NS дозволяє делегувати управління доменом хмарній інфраструктурі Azure, що необхідно для подальшої прив'язки SSL-сертифікатів та коректної маршрутизації трафіку через Application Gateway. На рисунку 3.19 представлені DNS-записи у зоні bankingapp.pp.ua.

Name	Type	TTL	Value
@	A	3600	4.225.54.14
@	NS	172800	ns1-01.azure-dns.com. ns2-01.azure-dns.net. ns3-01.azure-dns.org. ns4-01.azure-dns.info.
@	SOA	3600	Email: azuredns-hostmaster.microsoft.com Host: ns1-01.azure-dns.com. Refresh: 3600 Retry: 300 Expire: 2419200 Minimum TTL: 300 Serial number: 1

Рисунок 3.19 – Конфігурація DNS-зони для домену застосунку

В налаштуваннях домену bankingapp.pp.ua доменному реєстраторі nic.ua були вказані NS, що використовуються в Azure DNS (рис.3.20).

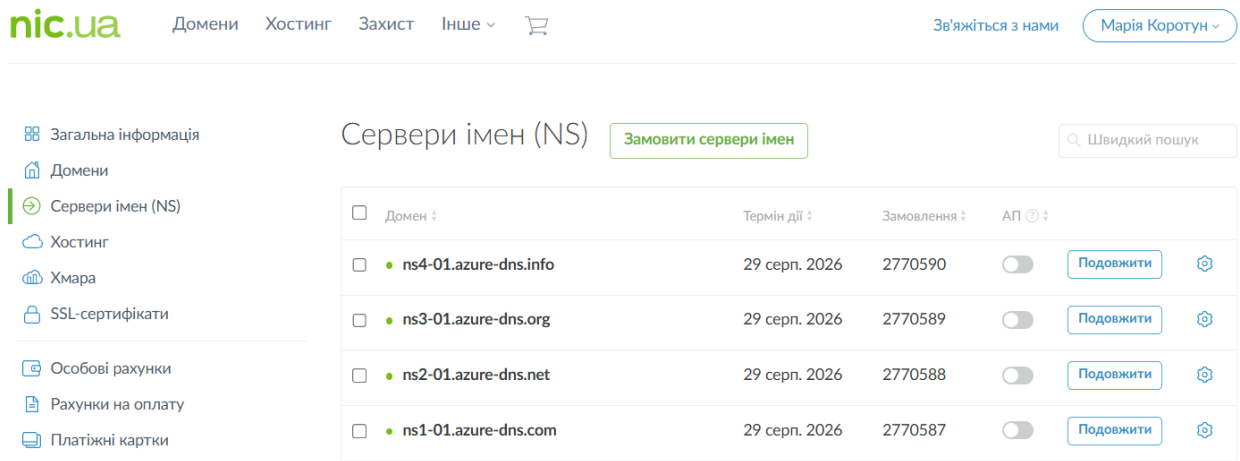


Рисунок 3.20 – Делегування NS-серверів на стороні реєстратора доменних імен nic.ua

Для забезпечення захисту вебзастосунку на рівні протоколу HTTP/HTTPS та запобігання поширеним атакам було розгорнуто Azure Application Gateway. На етапі створення було обрано рівень WAF_v2, який інтегрує функції брандмауера вебзастосунків. Налаштування передбачає роботу в режимі Prevention для моніторингу та блокування шкідливого трафіку.

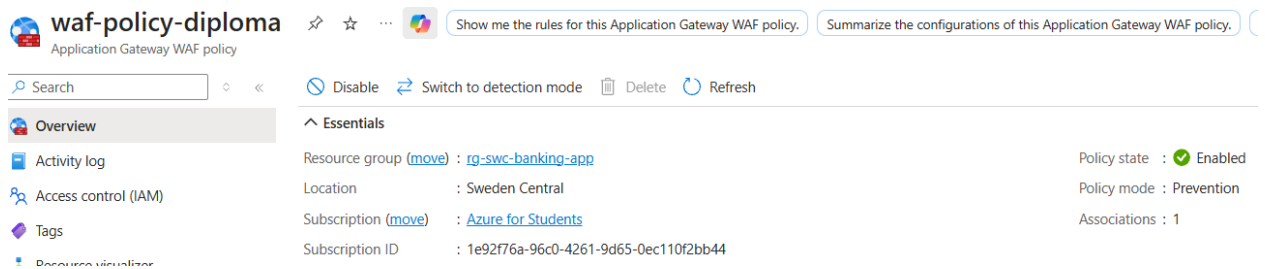


Рисунок 3.21 – Налаштування правила захисту у режимі Prevention

Для підтвердження ефективності розгорнутої політики безпеки Azure WAF було проведено практичне тестування системи на стійкість до сканування чутливих директорій та службових файлів додатка. Під час тестування було виконано серію нелегітимних HTTP-запитів до захищеного додатку bankingapp.pp.ua. Зокрема, було виконано спроби на отримання доступу до службового файлу Git-репозиторію (/.git/HEAD) (рис. 3.22) та файлу автентифікації користувачів (/.htpasswd) (рис.3.23), які відсутні у проєкті, але часто є ціллю для злоумисників.

Завдяки активованому режиму Prevention та базовим правилам OWASP CRS, шлюз програм Azure Application Gateway v2 миттєво ідентифікував ці запити як

аномальні та заблокував їх на рівні мережного периметру, повернувши клієнту стандартну помилку доступу HTTP 403 Forbidden.

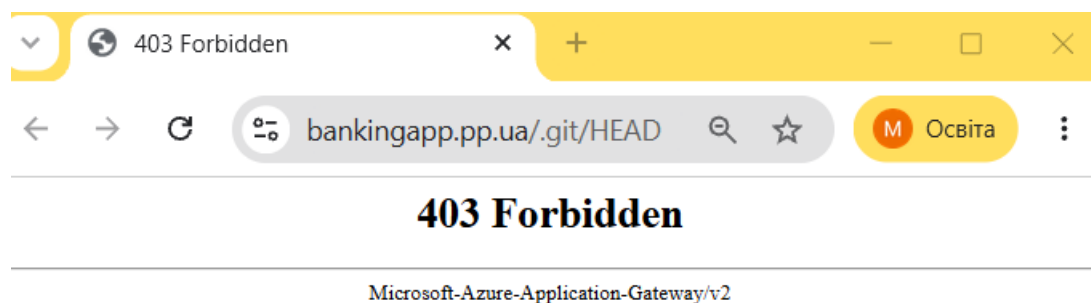


Рисунок 3.22 – Результат блокування сервером Azure WAF спроби несанкціонованого доступу до файлу репозиторію /.git/HEAD

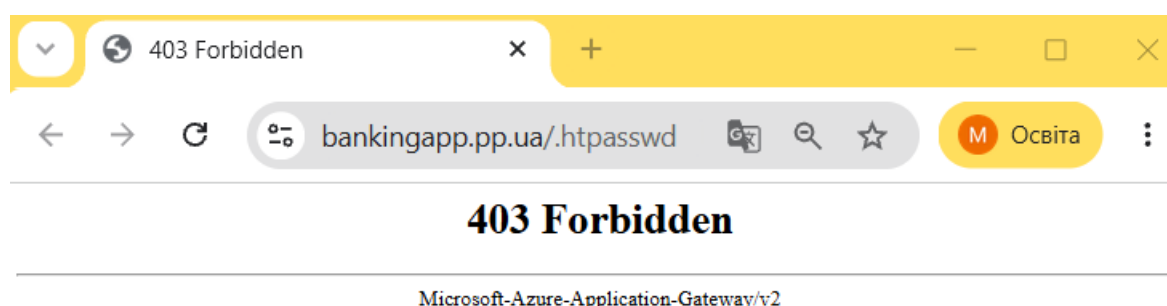


Рисунок 3.23 – Блокування запиту до службового файлу паролів /.htpasswd із поверненням статусу HTTP 403 Forbidden

Для забезпечення комплексного та ешелонованого захисту вебдодатку від розподілених атак, спрямованих на відмову в обслуговуванні, було реалізовано дворівневу систему протидії DDoS-загрозам. Перший рівень захисту розгорнуто на рівні всієї віртуальної мережі шляхом створення та інтеграції спеціалізованого плану захисту. За допомогою інтерфейсу командного рядка Azure CLI було ініційовано створення ресурсу захисту від DDoS-атак у цільовому регіоні розгортання інфраструктури (рис.3.24).

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				73
Змн.	Арк.	№ докум.	Підпис	Дата		

```
kb221_diploma3_kmo [ ~ ]$ az network ddos-protection create \
--resource-group rg-swc-banking-app \
--name AppDdosProtectionPlan \
--location swedencentral
{
  "etag": "W/\"6863d056-8712-406f-a319-a0b6cbb72ed2\"",
  "id": "/subscriptions/1e92f76a-96c0-4261-9d65-0ec110f2bb44/resourceGroups/rg-swc-banking-app/providers/Microsoft.Network/ddosProtectionPlans/AppDdosProtectionPlan",
  "location": "swedencentral",
  "name": "AppDdosProtectionPlan",
  "provisioningState": "Succeeded",
  "resourceGroup": "rg-swc-banking-app",
  "type": "Microsoft.Network/ddosProtectionPlans"
}
```

Рисунок 3.24 – Результат виконання CLI-команди створення плану захисту AppDdosProtectionPlan

Після успішної генерації плану захисту на рівні підписки, його було активовано безпосередньо для периметра хмарної мережі vnet-banking. Це дозволяє здійснювати безперервний моніторинг мережного трафіку, автоматично профілювати стандартні поведінкові паттерни додатку та застосовувати адаптивні алгоритми пом'якшення наслідків широкосмугових DDoS-атак на рівнях L3/L4 OSI. Процес прив'язки створеного плану захисту до віртуальної мережі, що представлено на рисунку 3..25.

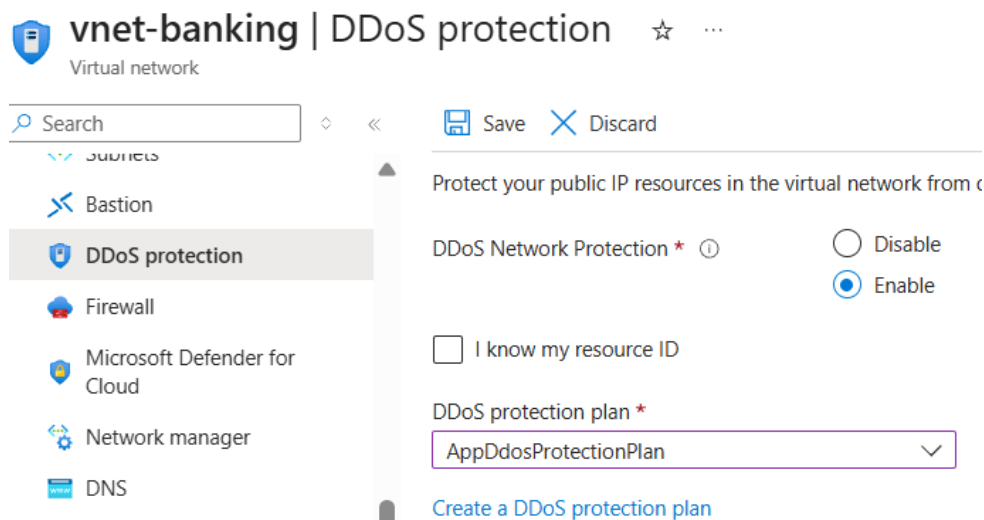


Рисунок 3.25 – Налаштування параметрів DDoS-захисту для віртуальної мережі vnet-banking

Другий рівень протидії загрозам реалізовано на рівні застосунку (L7 OSI) за допомогою кастомних політик брандмауера вебдодатків (WAF Policy). Для запобігання прикладним атакам типу HTTP Flood та нівелювання загрози вичерпання

ресурсів сервера одиночними клієнтами, було спроектовано та впроваджено правило обмеження інтенсивності запитів (Rate Limiting). Конфігурування параметрів користувацького правила DDoSRateLimit із блокуванням трафіку при перевищенні ліміту у 100 запитів за хвилину з однієї IP-адреси (рис.3.26).

Home > waf-policy-diploma | Custom rules

Add custom rule

This is a new version of the WAF policy custom rule experience and contains the latest features. All feedback is appreciated. [Click here to access the previous experience](#)

A custom rule is made up of one or more conditions followed by an action. [Learn more](#)

Custom rule name *

DDoSRateLimit

Enable rule

☒

Rule action

Deny traffic

Priority * ⓘ

20

Rule Type ⓘ

☐ Match

☒ Rate limit

Rate limit duration ⓘ

1 minute

Threshold (requests) * ⓘ

100

Group rate limit traffic by ⓘ

Client address

Deny traffic when all conditions are met

Custom rules require at least one condition.

+ Add condition

Remove condition

☐	Match type	Match variable	Operation	Operator	Value
☐	IP address	Remote address	Does contain		1 selected

Рисунок 3.26 – Інтерфейс конфігурування кастомного правила обмеження інтенсивності запитів у WAF

Розроблене правило було успішно інтегроване в загальну політику безпеки waf-policy-diploma, яка безпосередньо асоційована з Azure Application Gateway. Активація цього правила забезпечує превентивне відсікання аномальних сплесків трафіку до того, як шкідливі HTTP-запити надійдуть на обробку до віртуальної машини з вебдодатком Spring Boot. Фінальний стан і статус працездатності впровадженого правила в структурі політик WAF (рис.3.27).

waf-policy-diploma | Custom rules

Application Gateway WAF policy

Search

Configure a policy with custom-authored rules. Once a rule is matched, the corresponding action defined in the rule is applied to the request. Once such a value for a rule denotes a higher priority. [Learn more](#)

Overview

Activity log

Access control (IAM)

Tags

Resource visualizer

+ Add custom rule

Refresh

Enable rules

Disable rules

Duplicate in this policy

Copy to another policy

Remove

☐	Priority	Name	Rule type	Status	Action
☐	20	DDoSRateLimit	RateLimitRule	Enabled	Block

Рисунок 3.27 – Перелік активованих користувацьких правил у політиці waf-policy-diploma

3.3. Процес розгортання та експлуатації

3.3.1. Збірка артефакту та безпечна доставка JAR-файлу на віртуальну машину через SCP.

Після підтвердження прав було проведено локальну збірку JAR-артефакту та його передачу на сервер.

Успішна компіляція проєкту Maven та використання SCP для доставки файлу забезпечили цілісність коду на віддаленій машині (рис.3.28).

```
[INFO] Replacing main artifact [REDACTED]
[INFO] The original artifact has been renamed to [REDACTED]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 11.687 s
[INFO] Finished at: 2026-04-05T17:57:57+03:00
[INFO] -----
```

Рисунок 3.28 – Успішне завершення фази збірки проєкту

Для відправки файлу по SCP з локальної машини до віртуальної машини було створено тимчасове правило в NSG, що дозволяло доступ по SSH (рис.3.29).

```
PS [REDACTED] \mini-banking-app> scp -i "[REDACTED]\ssh_public_key.pem" target/mini-banking-app-0.0.1-SNAPSHOT.jar azureuser@68.210.96.207:/home/azureuser/
The authenticity of host '68.210.96.207 (68.210.96.207)' can't be established.
ED25519 key fingerprint is SHA256:ml8NUzAKgFWRh4iAYBkATu6AmsUIj3c4kTo6XuIFA84.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])?
Warning: Permanently added '68.210.96.207' (ED25519) to the list of known hosts.
mini-banking-app-0.0.1-SNAPSHOT.jar 100% 80MB 12.6MB/s 00:06
```

Рисунок 3.29 – Передача скомпільованого JAR-файлу на віддалений сервер за допомогою протоколу SCP

Запуск додатка ініціював процес Hibernate, який автоматично створив схему бази даних. Логи Hibernate підтвердили успішне створення таблиць users та accounts, що свідчить про коректне з'єднання з Azure SQL.

```

Hibernate:
    create table accounts (
        id bigint identity not null,
        balance numeric(19,4) not null,
        currency varchar(255),
        type varchar(255),
        user_id bigint not null,
        primary key (id)
    )
Hibernate:
    create table transactions (
        id bigint identity not null,
        amount numeric(19,4) not null,
        created_at datetime2(6) not null,
        type varchar(255),
        account_id bigint not null,
        primary key (id)
    )
Hibernate:
    create table users (
        id bigint identity not null,
        enabled bit not null,
        password varchar(255) not null,
        role varchar(255) not null check (role in ('USER','ADMIN')),
        username varchar(50) not null,
        primary key (id)
    )

```

Рисунок 3.30 – Перевірка результату в Azure SQL: підтвердження створення таблиць через Query Editor

Для захисту вебдодатка та організації ефективного розподілу запитів було реалізовано схему реверс-проксі на базі вебсервера Nginx, розгорнутого на віртуальній машині Azure. Важливим етапом побудови безпечного каналу зв'язку є забезпечення SSL-термінації. Централізоване й безпечне зберігання криптографічних ключів та SSL/TLS-сертифікатів було організовано за допомогою сервісу Azure Key Vault із активованою функцією захисту від видалення. Створення сховища ключів виконано через Azure CLI (рис.3.30).

```
kb221_diploma3_kmo [ ~ ]$ az keyvault create \
--name kv-banking-diploma \
--resource-group rg-swc-banking-app \
--location "Sweden Central" \
--enable-purge-protection true
{
  "id": "/subscriptions/1e92f76a-96c0-4261-9d65-0ec110f2bb44",
  "location": "swedencentral",
  "name": "kv-banking-diploma",
  "properties": {
    "accessPolicies": [],
    "createMode": null,
    "enablePurgeProtection": true,
    "enableRbacAuthorization": true,
  }
}
```

Рисунок 3.30 – Створення сховища Azure Key Vault із увімкненим захистом від очищення

Після успішного розгортання сховища всередині Azure Key Vault було ініційовано процес генерації та імпорту довіреного SSL/TLS-сертифіката під назвою ssl-cert для забезпечення шифрування трафіку за протоколом HTTPS (рис. 3.31).

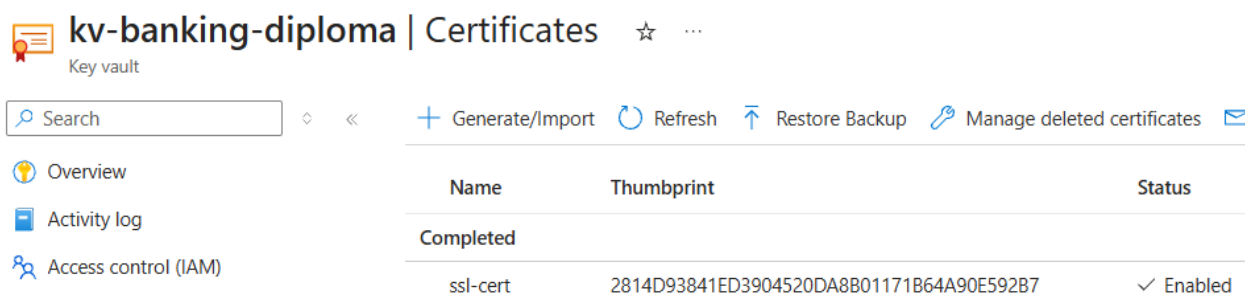


Рисунок 3.31 – Стан успішно згенерованого SSL-сертифіката у вкладці Certificates сховища Azure Key Vault

Для того щоб віртуальна машина з вебсервером Nginx могла динамічно зчитувати сертифікат безпеки зі сховища, за допомогою Azure CLI було створено прив'язку Managed Identity віртуальної машини та призначено їй роль Key Vault Secrets Officer (рис.3.32).

```

kb221_diploma3_kmo [ ~ ]$ VM_PRINCIPAL_ID=$(az vm show -g rg-swc-banking-app -n vm-diploma --query "identity.principalId" -o tsv)
kb221_diploma3_kmo [ ~ ]$ az role assignment create \
  --role "Key Vault Secrets Officer" \
  --assignee-object-id $VM_PRINCIPAL_ID \
  --assignee-principal-type ServicePrincipal \
  --scope "/subscriptions/1e92f76a-96c0-4261-9d65-0ec110f2bb44/resourceGroups/rg-swc-banking-app/providers/Microsoft.KeyVault/vaults/4b-4892-9c63-f63aa127606d",
  {
    "condition": null,
    "conditionVersion": null,
    "createdBy": null,
    "createdOn": "2026-06-08T14:53:08.905922+00:00",
    "delegatedManagedIdentityResourceId": null,
    "description": null,
    "id": "/subscriptions/1e92f76a-96c0-4261-9d65-0ec110f2bb44/resourceGroups/rg-swc-banking-app/providers/Microsoft.KeyVault/vaults/4b-4892-9c63-f63aa127606d",
    "name": "70cd2b81-8b4b-4892-9c63-f63aa127606d",
    "principalId": "3e2c6a30-c4cc-4c58-95bb-bcb4e1776d28",
    "principalType": "ServicePrincipal",
    "resourceGroup": "rg-swc-banking-app",
    "roleDefinitionId": "/subscriptions/1e92f76a-96c0-4261-9d65-0ec110f2bb44/providers/Microsoft.Authorization/roleDefinitions/b86a2b46-4c9c-4720-b662-8c76d771bf52",
    "scope": "/subscriptions/1e92f76a-96c0-4261-9d65-0ec110f2bb44/resourceGroups/rg-swc-banking-app/providers/Microsoft.KeyVault/vaults/4b-4892-9c63-f63aa127606d",
    "type": "Microsoft.Authorization/roleAssignments",
    "updatedBy": "0775a014-0d1a-4ce7-aaca-be9831bbec87",
    "updatedOn": "2026-06-08T14:53:09.132655+00:00"
  }

```

Рисунок 3.32 – Призначення ролі Key Vault Secrets Officer для ідентифікатора віртуальної машини

З метою автоматизації випуску та оновлення сертифікатів через утиліту Certbot було розгорнуто Service Principal в Entra ID з правами DNS Zone Contributor, що дозволяє виконувати перевірку володіння доменом за допомогою DNS-01 challenge (рис. 3.32).

```

kb221_diploma2_kmo [ ~ ]$ az ad sp create-for-rbac --name "certbot-dns-azure" --role "DNS Zone Contributor" --scopes "/subscriptions/8e739f50-9476-4b19-98b2-67cacc7e38/resourceGroups/rg-spc-banking-app/providers/Microsoft.Network/dnszones/bankingapp.pp.ua"
Found an existing application instance: (id) 357ac7f9-8c98-49f9-a8da-0dc0e9ac175d. We will patch it.
Creating 'DNS Zone Contributor' role assignment under scope '/subscriptions/8e739f50-9476-4b19-98b2-67cacc7e38/resourceGroups/rg-spc-banking-app/providers/Microsoft.Network/dnszones/bankingapp.pp.ua'
The output includes credentials that you must protect. Be sure that you do not include these credentials in your code or check the credentials into your source control. For more information, see https://aka.ms/azadsp-cli
{
  "appId": "0ab931e6-3c8d-4a4a-a7d0-c6883f25f1d3",
  "displayName": "certbot-dns-azure",
  "password": "3o.BQ~86g~T_IGI08TDgyp4ABysF3h_pkgZS3dww",
  "tenant": "d84995f6-c5b5-422d-b123-c1243b2f5125"
}
kb221_diploma2_kmo [ ~ ]$

```

Рисунок 3.32 – Створення сервісного принципалу з правами доступу до керування DNS-зоною інфраструктури

Наступним кроком стало делегування доменного імені bankingapp.pp.ua на публічну IP-адресу інфраструктури. Додавання А-запису в Azure DNS Zone було виконано за допомогою командного інтерфейсу платформи. Коректність внесення змін та публікації ресурсних записів для доменного імені перевірено безпосередньо через вебкерування Azure Portal у розділі Recordsets (див. рис. 3.32).

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				79
Змн.	Арк.	№ докум.	Підпис	Дата		

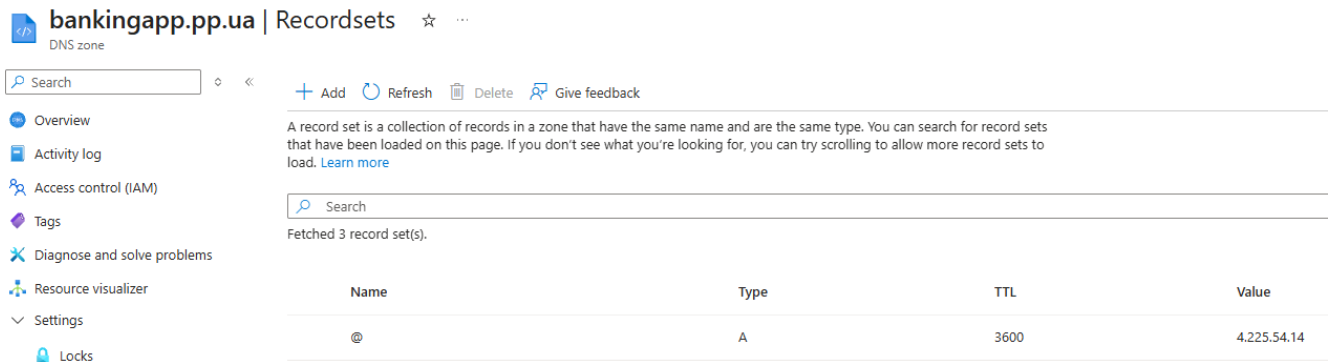


Рисунок 3.32 – Перевірка конфігурації А-запису для домену bankingapp.pp.ua в інтерфейсі Azure DNS zones

Фінальним етапом архітектурного налаштування став конфігураційний файл самого вебсервера Nginx. У файлі конфігурації віртуального хосту додатка було прописано директиву `map` для коректного визначення реальної IP-адреси клієнта через заголовок `X-Forwarded-For`. Також налаштовано проксіювання вхідного трафіку з порту 80 на локальний порт вебдодатка із передачею відповідних HTTP-заголовків, що забезпечує прозору SSL-термінацію на рівні проксі-сервера (рис. 2.33).



Рисунок 3.33 – Конфігураційний файл віртуального хосту Nginx у текстовому редакторі GNU nano 7.2

3.3.2. Налаштування системних сервісів операційної системи: Backup, Update Management та Change Tracking

Безпека на рівні хоста забезпечується через постійний моніторинг цілісності системи. Для забезпечення цілісності середовища виконання було розгорнуто розширення `ChangeTracking-Linux`. Цей інструмент у режимі реального часу відстежує зміни у системних файлах, реєстрах та встановленому ПЗ, що дозволяє виявити несанкціоноване втручання або випадкові зміни конфігурації. Також встановлено `Azure Monitor Agent` для збору та передачі логів безпеки (рис. 3.34).

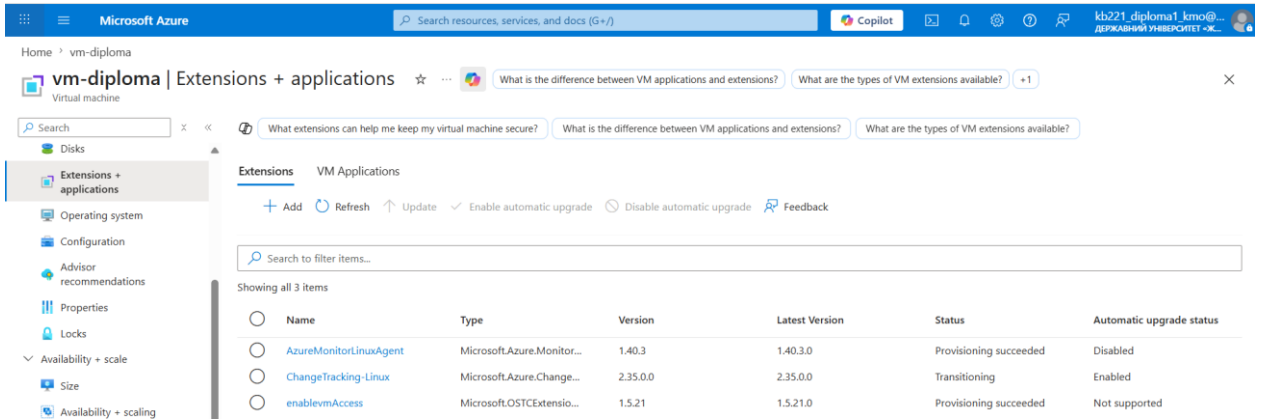


Рисунок 3.34 – Стан розгортання розширень віртуальної машини для моніторингу змін та управління безпекою

Важливим аспектом експлуатації є підтримка актуального стану системного ПЗ та безпека вузлів. Для віртуальної машини vm-diploma було активовано механізм автоматичного оцінювання та керування оновленнями (рис. 3.35). Налаштування включає періодичну перевірку наявності патчів та вибір стратегії оркестрації (Azure Managed - Safe Deployment), що дозволяє автоматизувати процес інсталяції критичних оновлень безпеки без прямого втручання адміністратора (див.рис. 3.36).

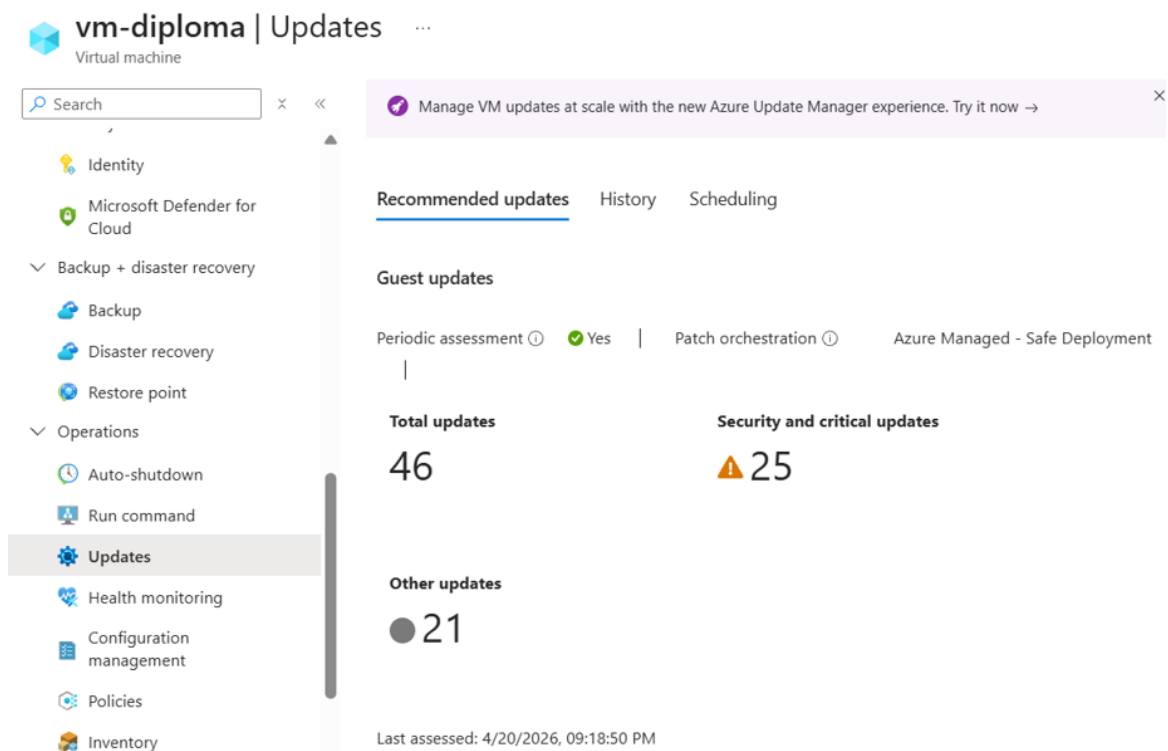


Рисунок 3.35 – Інтерфейс керування оновленнями віртуальної машини vm-diploma

Linux (1)				
Machine Name	Resource type	Periodic assessment ①	Hotpatch ①	Patch orchestration ①
-	-	Select and apply to all ▾	Select and apply to all ▾	Select and apply to all
vm-diploma	Azure virtual machine	Enable ▾	Not available ▾ ①	Azure Managed - Safe Deployment

Рисунок 3.36 – Конфігурування параметрів оновлення для Linux-сервера з увімкненим періодичним оцінюванням

З метою гарантування відмовостійкості системи та можливості швидкого відновлення після збоїв було налаштовано сервіс Azure Backup. Створення сховища Recovery Services vault (rsv-banking-app) дозволяє централізовано керувати резервними копіями віртуальної машини. Для автоматизації процесу було розроблено політику щоденного резервного копіювання, яка визначає час створення знімка системи та термін його зберігання у хмарному сховищі (рис. 3.37).

The screenshot displays the 'Create policy' interface for an Azure Virtual Machine. The policy is named 'Daily-Policy'. The backup schedule is set to 'Daily' at '2:00 AM' in '(UTC) Coordinated Universal Time'. The 'Instant restore' section shows 'Retain instant recovery snapshot(s) for 1 Day(s)'. The 'Retention range' section is checked for 'Retention of daily backup point', with settings 'At 2:00 AM' and 'For 7 Day(s)'.

Рисунок 3.37 – Конфігурування параметрів резервного копіювання для віртуальної машини vm-diploma та створення Recovery Services vault

Для забезпечення цілісності системи та оперативного виявлення несанкціонованих змін у конфігураційних файлах або встановленому програмному забезпеченні було впроваджено функціонал Change Tracking and Inventory. Налаштування здійснюється шляхом інтеграції віртуальної машини з АМА та робочою областю Log Analytics. Це дозволяє автоматично відстежувати стан сервісів, зміни в реєстрі для Windows або конфігураційних файлах для Linux та формувати звіти про встановлене ПЗ (рис. 3.38).

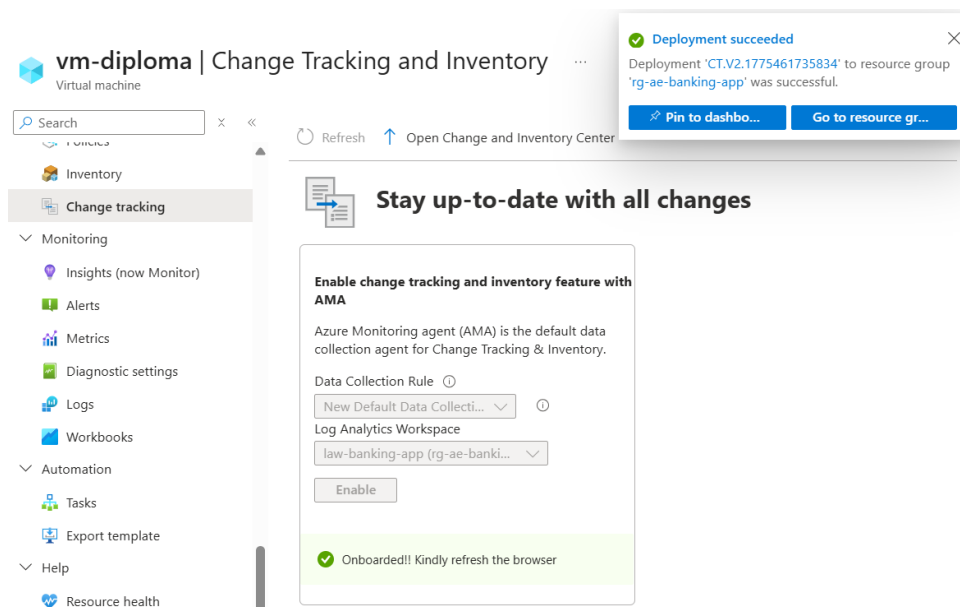


Рисунок 3.38 – Процес активації функції Change Tracking та інвентаризації для віртуальної машини vm-diploma за допомогою AMA-агента

3.4. Верифікація безпеки та моніторинг стану системи

3.4.1. Аналіз безпекових сповіщень у Defender for Cloud та Azure Sentinel

Централізоване управління інцидентами реалізовано через хмарну SIEM-систему Microsoft Sentinel, яка інтегрується з іншими захисними рішеннями. Налаштування зв'язку між робочою областю Log Analytics та Microsoft Sentinel дозволяє агрегувати події з усіх рівнів інфраструктури. Це забезпечує видимість загроз у режимі «єдиного вікна». Рисунки демонструють успішне підключення та готовність системи до аналізу аномалій, спроб брутфорсу або підозрілої активності Managed Identity (рис. 3.39).

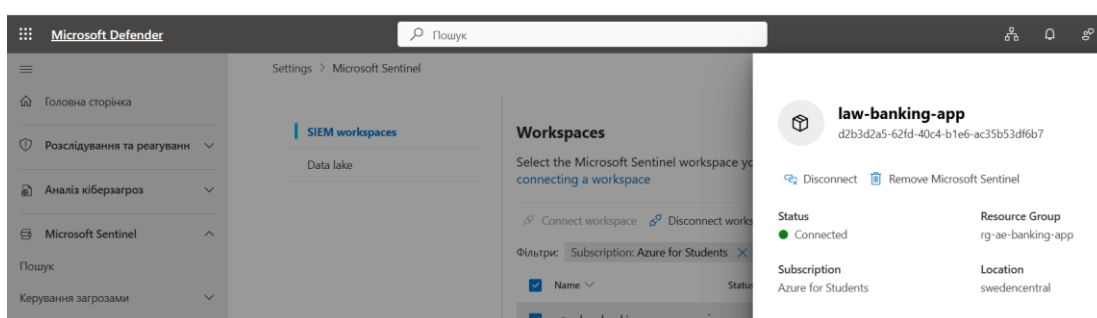


Рисунок 3.39 – Статус інтеграції Log Analytics Workspace law-banking-app із Microsoft Sentinel

У межах налаштування системи моніторингу було активовано плани CSPM та CWP. Це дозволяє здійснювати безперервний аудит конфігурацій та захищати безпосередньо сервери та бази даних від цілеспрямованих атак. На рисунку продемонстровано повне покриття моніторингом ключових ресурсів проєкту (рис. 3.40).

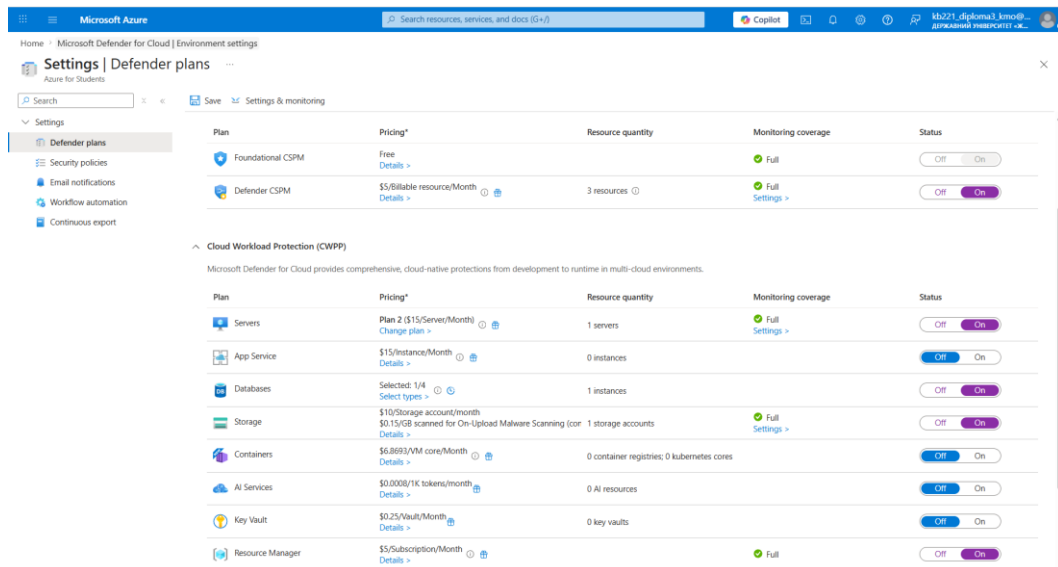


Рисунок 3.40 – Активація планів захисту в Microsoft Defender for Cloud для моніторингу серверів та баз даних

З метою оперативного виявлення цілеспрямованих атак на вебдодаток у консолі журналювання було розроблено аналітичні KQL-запити до таблиці AGWFWirewallLogs. Запит фокусується на ідентифікації подій, які були успішно перехоплені та заблоковані Application Gateway, із виведенням ключових атрибутів: IP-адреси зломисника, target URL, типу сигнатури атаки за класифікацією OWASP та детальних даних запиту (див. рис. 3.41).

Custom logs via AMA (Preview) ...

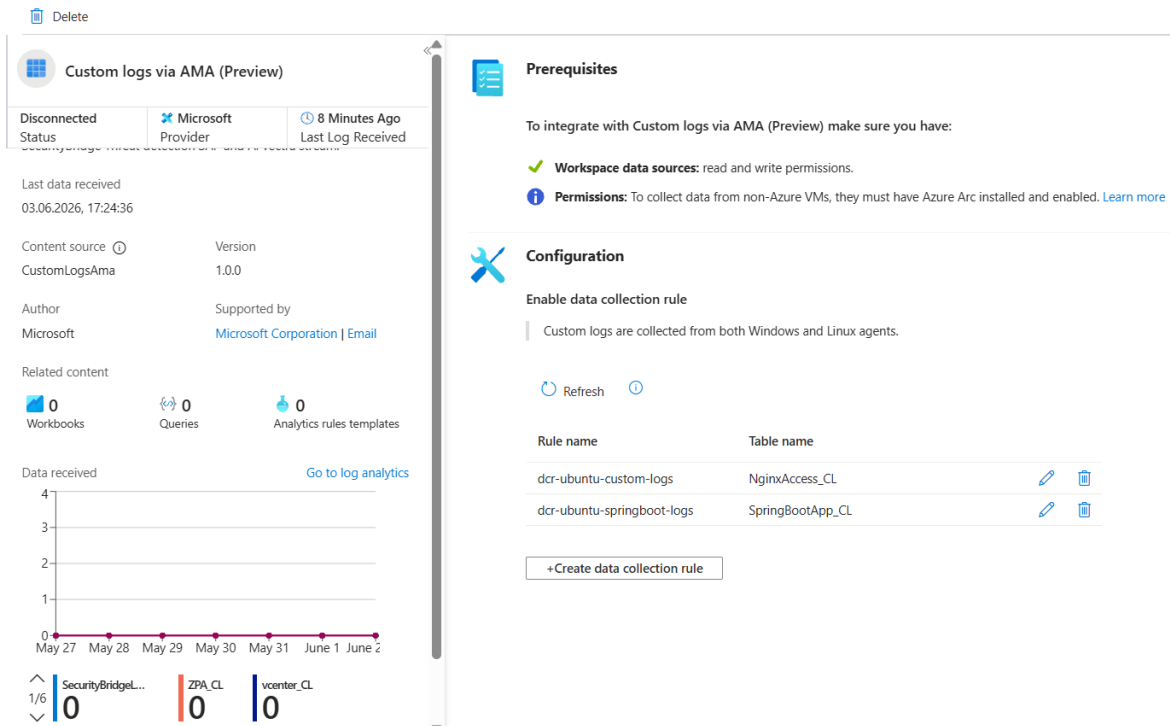


Рисунок 3.41 – Моніторинг та аналітична фільтрація заблокованих подій WAF за допомогою KQL-запиту в Azure Sentinel

Для переведення процесу моніторингу в проактивний режим у розділі Analytics системи Microsoft Sentinel було активовано та налаштовано аналітичне правило WAF - Attack Detected. Дане правило виконує циклічний аналіз потоку логів WAF та автоматично ініціює створення інцидентів безпеки у разі реєстрації критичних спроб компрометації (наприклад, SQL-ін'єкцій або спроб доступу до системних файлів) (рис. 3.42).

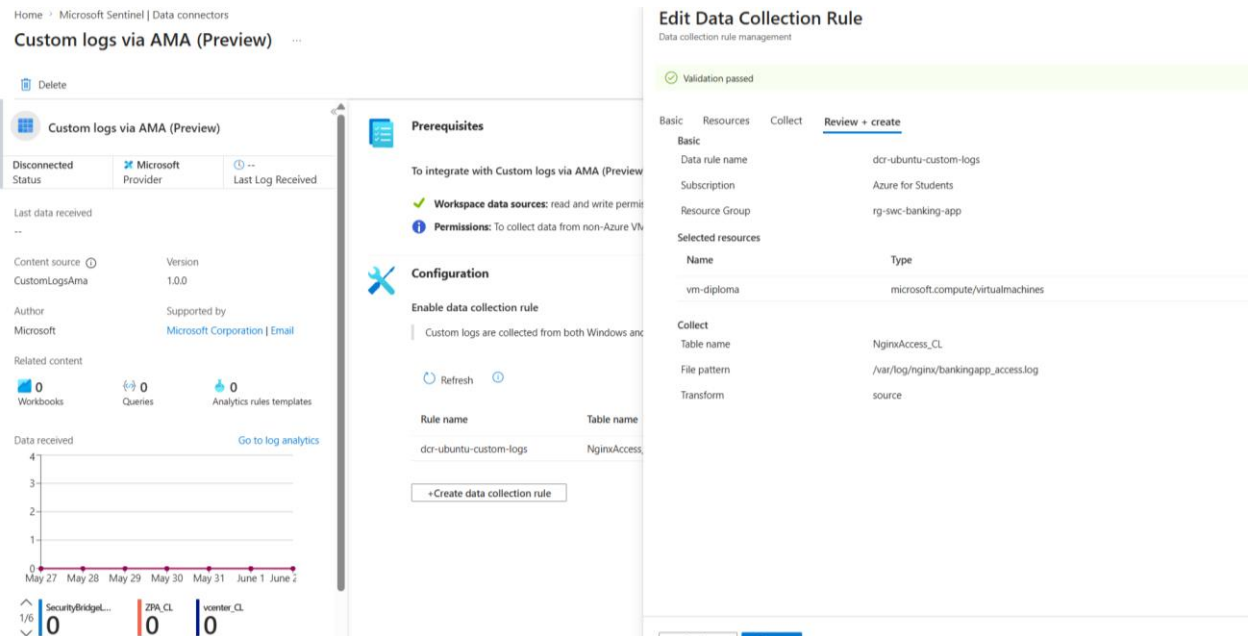


Рисунок 3.42 – Налаштування та статус Analytics rule у Microsoft Sentinel для генерації інцидентів на основі логів WAF

Завдяки налаштованим Automation rules, платформа Sentinel перевірила статус виконання блокування на рівні WAF і зафіксувала, що загрозу вже нейтралізовано. Система автоматично додала службовий коментар до інциденту із зазначенням деталей заблокованої сесії, що демонструє ефективну роботу SOAR-функціоналу в умовах автоматичного реагування на кіберзагрози (див рис. 3.43).

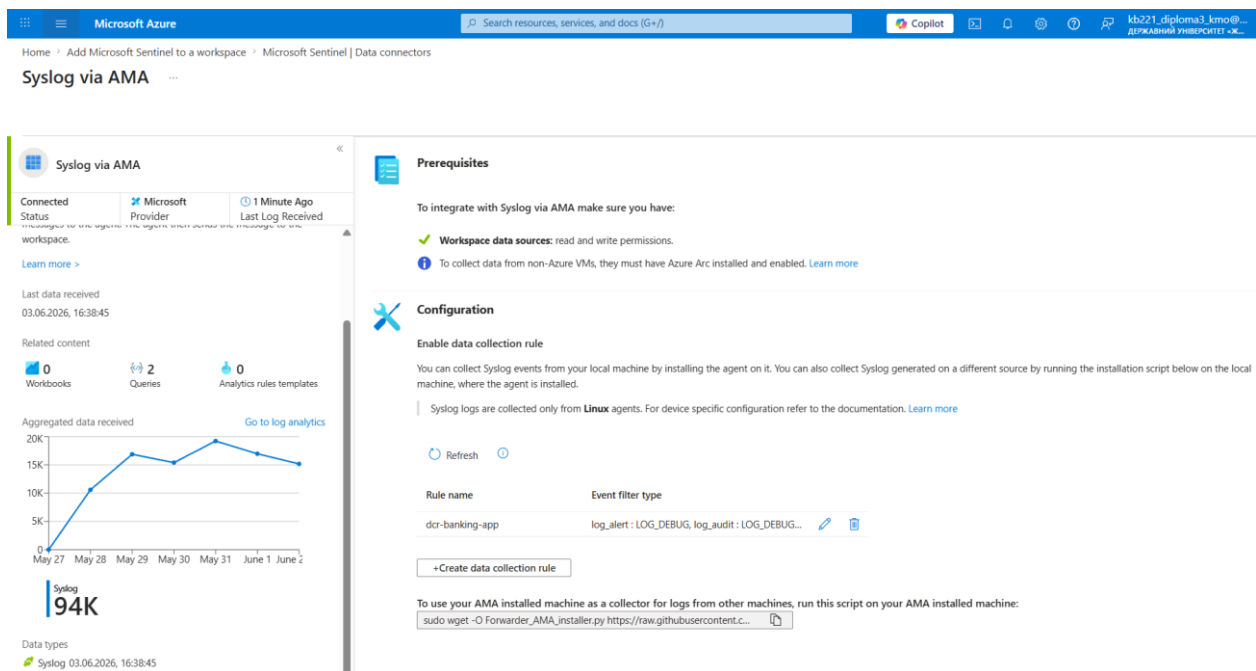


Рисунок 3.43 – Журнал реєстрації інцидентів та автоматичне додавання коментарів безпеки у Microsoft Sentinel

Після розгортання інфраструктури критично важливим є етап оцінки загального стану безпеки CSPM. Використання панелі керування Microsoft Defender for Cloud дозволяє централізовано переглядати Secure Score, виявляти потенційні шляхи атак та отримувати рекомендації щодо усунення вразливостей на рівні підписок Azure. Огляд консолі підтверджує успішне підключення ресурсів до системи моніторингу та готовність до збору аналітики безпеки (рис. 3.44).

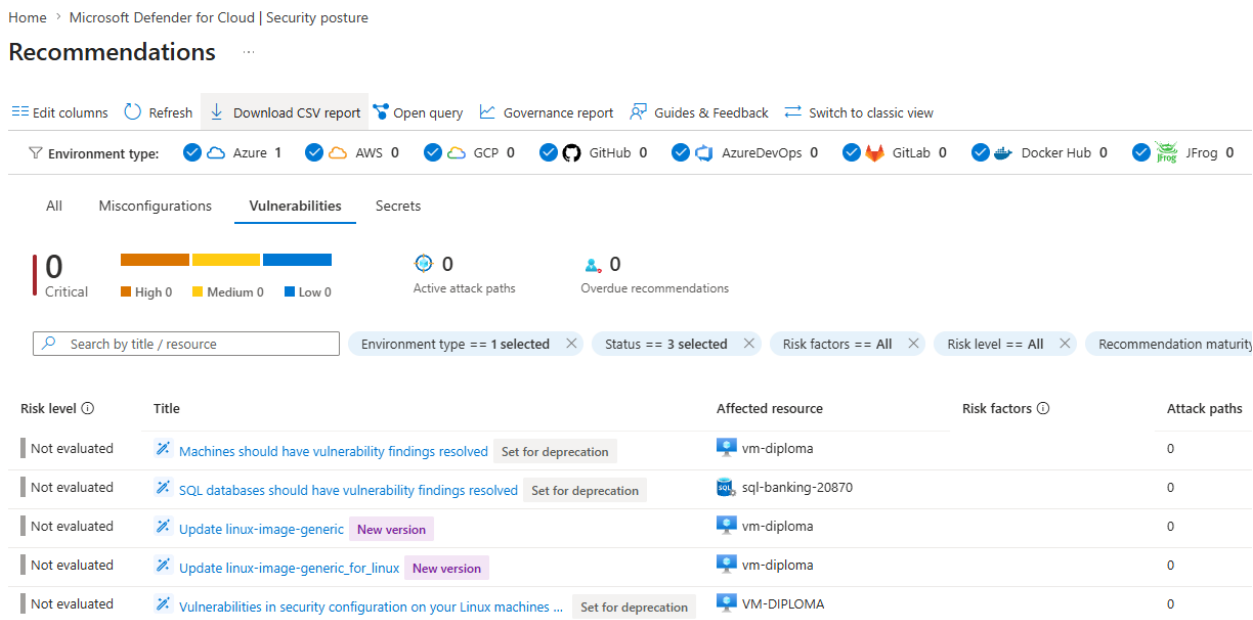


Рисунок 3.44 – Моніторинг стану безпеки підписки та активних рекомендацій

3.4.2. Тестування стійкості застосунку до OWASP атак через інтерфейс Application Gateway

Для захисту вебзастосунку на рівні прикладного протоколу використано Azure Application Gateway з активованим компонентом WAF. Конфігурація базується на наборі правил CRS, що автоматично блокують поширені типи атак з переліку OWASP Top 10, такі як SQL-ін'єкції та XSS (рис. 3.45).

Під час тестування було перевірено роботу WAF у режимі «Prevention», що дозволило заблокувати спроби несанкціонованого доступу.

☐	Rule Id	Description	Action	Status	Exclusions	Rule group	Rule set
☒	OWASP_3.2 (187)						
☐	200002	Failed to Parse Request Body.	Anomaly score	Enabled		General	OWASP_3.2
☐	200003	Multipart Request Body Strict Validation.	Anomaly score	Enabled		General	OWASP_3.2
☐	200004	Possible Multipart Unmatched Boundary.	Anomaly score	Enabled		General	OWASP_3.2
☐	911100	Method is not allowed by policy	Anomaly score	Enabled		REQUEST-911-METHOD-ENFOR...	OWASP_3.2
☐	913100	Found User-Agent associated with securit...	Anomaly score	Enabled		REQUEST-913-SCANNER-DETEC...	OWASP_3.2

Рисунок 3.45 – Налаштування правил WAF для захисту від вразливостей OWASP

Для верифікації ефективності захисту було проведено тестування застосунку на стійкість до атак із переліку OWASP Top 10. Зокрема, змодельовано спробу SQL-ін'єкції шляхом введення конструкції '1'=1' у форму автентифікації користувача. Політика Web Application Firewall (waf-policy-diploma), яка працює в режимі перехоплення (Prevention), ідентифікувала шкідливу сигнатуру у вхідному HTTP-запиті. В результаті Azure Application Gateway миттєво заблокував запит на периметрі мережі, не допустивши його до бекенду Spring Boot, та повернув зловмиснику стандартну помилку 403 Forbidden. Процес ініціації та успішного блокування веб-атаки продемонстровано на рисунку 3.46.

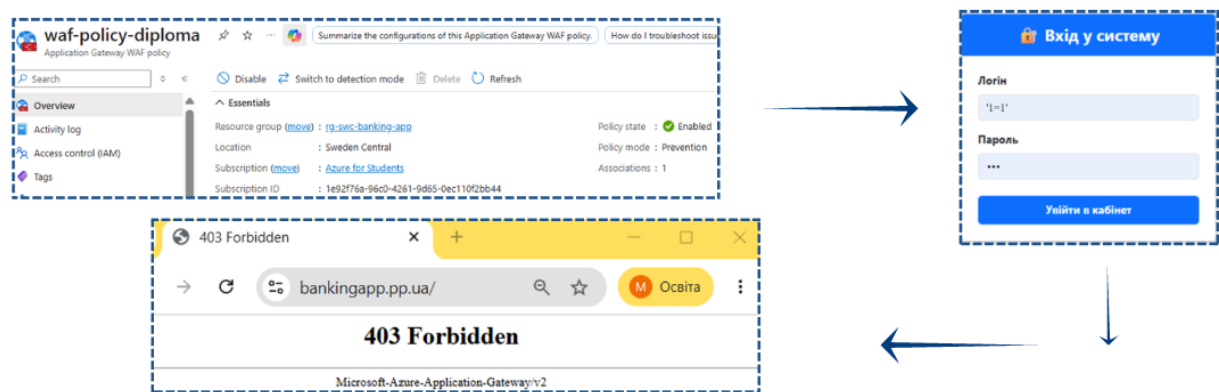


Рисунок 3.46 – Імітація SQL-ін'єкції та її успішне блокування політикою Azure WAF із поверненням коду 403 Forbidden

3.4.3. Візуалізація та аналіз логів у Log Analytics Workspace

У підрозділі моніторингу було налаштовано Diagnostic Settings для бази даних SQL. Це дозволяє передавати логи аудиту, метрики продуктивності та результати сканування на вразливості безпосередньо в Log Analytics. Така конфігурація є критичною для проведення ретроспективного аналізу в разі виникнення інциденту безпеки (рис. 3.47).

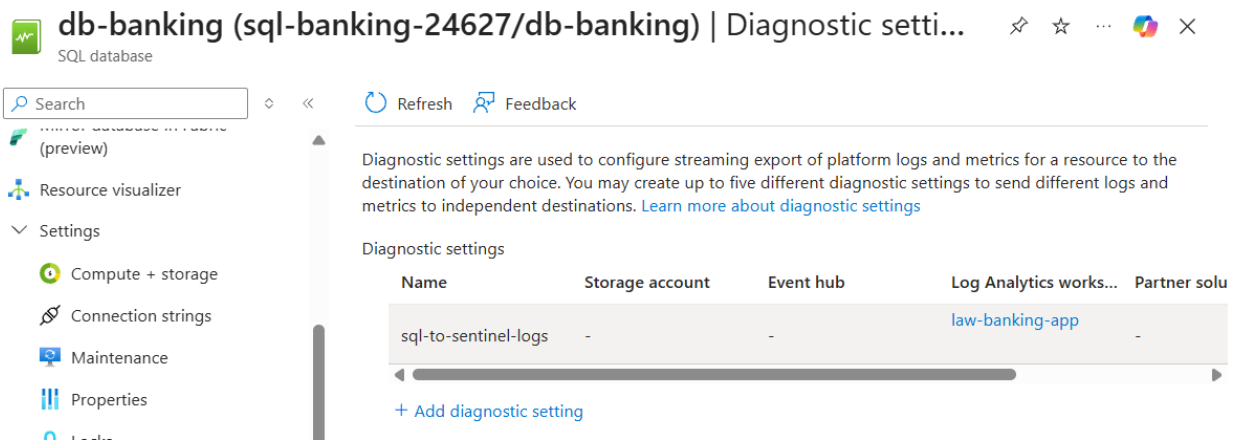


Рисунок 3.47 – Конфігурація потокової передачі логів діагностики SQL Database у централізоване сховище

Для централізованого збору подій безпеки та системних логів було впроваджено DCR. Це дозволяє агрегувати дані з віртуальної машини vm-diploma та передавати їх до сховища Log Analytics Workspace. Створення правила dcr-banking-app забезпечує чітке визначення джерел та місця призначення, що є фундаментом для подальшого аналізу в Azure Sentinel (рис. 3.48).

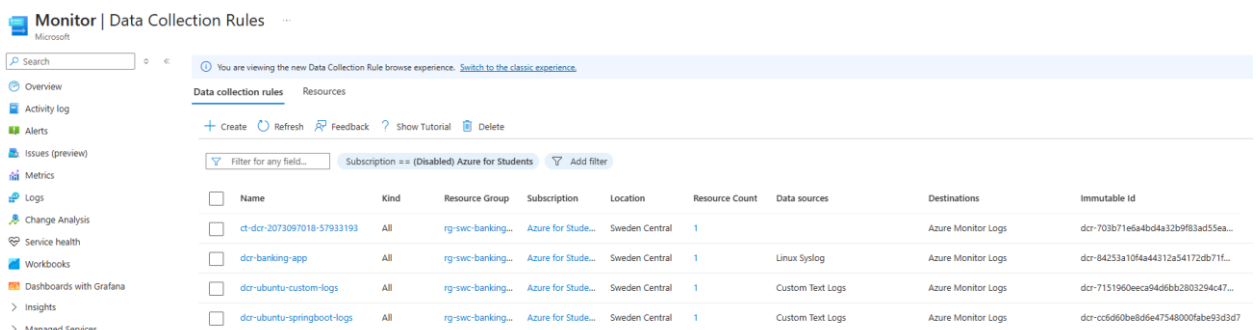


Рисунок 3.48 – Перелік активних Data collection rules у групі ресурсів проєкту

Ключовим компонентом системи моніторингу є робоча область Log Analytics (law-banking-app). Вона слугує центральним репозиторієм для збору метрик та журналів подій. Стан «Active» та статус оперативних питань «ОК» підтверджують працездатність системи збору логів. Також враховано вимоги безпеки передачі даних, що підтверджується повідомленням про використання протоколу TLS 1.2 для шифрування трафіку моніторингу (рис. 3.49).

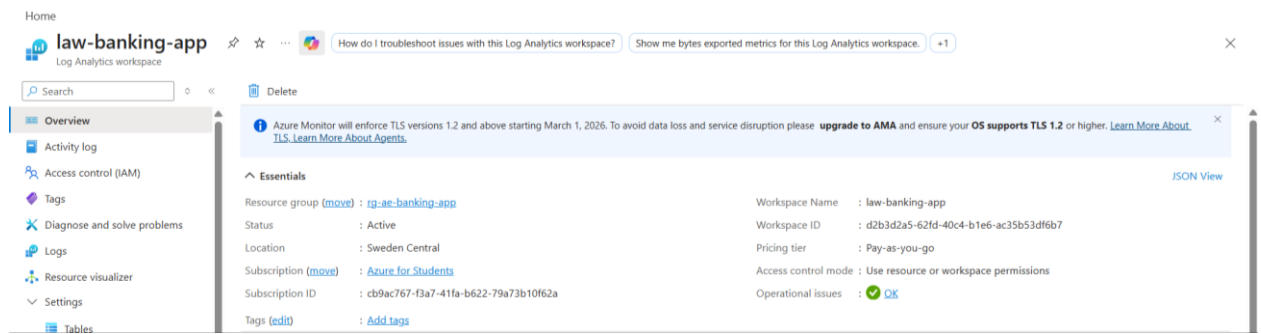


Рисунок 3.49 – Огляд робочої області Log Analytics law-banking-app, що використовується для агрегації та аналізу логів системи.

Висновок до третього розділу

У третьому розділі виконано практичне розгортання та верифікацію захищеної інфраструктури вебдодатка в Microsoft Azure. У межах віртуальної мережі vnet-banking налаштовано безпечне адміністрування Ubuntu Server 24.04 через Azure Bastion, що дозволило повністю закрити SSH-порти від зовнішнього доступу. Захист прикладного рівня успішно забезпечено розгортанням Azure Application Gateway з модулем WAF, який підтвердив свою ефективність під час тестування стійкості системи до атак OWASP Top 10. Операційну безпеку автоматизовано сервісами Backup й Update Management, а безперервний проактивний моніторинг та аналіз логів централізовано в контурі Log Analytics Workspace, Defender for Cloud та Azure Sentinel.

ВИСНОВКИ

У кваліфікаційній роботі успішно вирішено актуальне науково-практичне завдання, яке полягало в розробці, проектуванні та практичній реалізації комплексної системи безпеки хмарної інфраструктури Microsoft Azure для надійного розміщення і функціонування вебдодатку на базі фреймворку Spring Boot.

Проведений аналіз принципів побудови безпечних хмарних архітектур дозволив визначити, що лідерство платформи Microsoft Azure зумовлене її високою гнучкістю та наявністю потужного вбудованого інструментарію комплаєнсу, такого як Azure Advisor, Azure Policy та спеціалізована документація з відповідності міжнародним стандартам. На основі цього дослідження було доведено, що для побудови ефективного ешелонованого захисту вебдодатків критично важливим є комплексне поєднання засобів фільтрації трафіку на рівні L7 через Azure Application Gateway з Web Application Firewall, ізоляції адміністративного доступу за допомогою Azure Bastion, а також організації приватного підключення до сховищ даних через Private Endpoints під контролем централізованого SIEM/XDR-моніторингу на базі Defender for Cloud та Azure Sentinel.

Під час проектування інфраструктури та внутрішньої архітектури застосунку було розроблено архітектурні рішення, що забезпечили сувору сегментацію мережі, впровадження політик найменших привілеїв та чітке розмежування ролей на основі моделі RBAC. Важливим досягненням на рівні програмної архітектури Java-додатку стало інтегрування технології Managed Identity, що дозволило повністю відмовитися від збереження критичних компрометуючих даних, таких як паролі чи токени доступу, у конфігураційних файлах та вихідному коді застосунку.

Практична реалізація та розгортання безпечного вебдодатку підтвердили надійність спроектованих рішень. У ході розгортання було створено ізольовану віртуальну мережу з виділеними підмережами, де рух трафіку жорстко регламентується правилами мережних груп безпеки. Було успішно налаштовано захищений SSH-доступ до цільового сервера Ubuntu 24.04 через шлюз Azure Bastion, а також розгорнуто бази даних Azure SQL.

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				91
Змн.	Арк.	№ докум.	Підпис	Дата		

Безпеку на рівні самого застосунку та операційної системи посилено шляхом програмного впровадження механізмів автентифікації й авторизації через Spring Security, конфігурування Nginx як реверс-проксі для SSL-термінації, а також активації системних сервісів автоматичного резервного копіювання, управління оновленнями та відстеження змін. Фінальна верифікація та оцінка ефективності впровадженої системи безпеки продемонстрували високий рівень захищеності розгорнутого середовища. Комплексне тестування вебдодатку на стійкість до вразливостей із переліку OWASP Top 10 підтвердило, що інтегрований Web Application Firewall успішно блокує спроби ін'єкцій та зловмисних запитів на межі хмарного периметра. Налагоджений збір логів у Log Analytics Workspace у зв'язці з Azure Sentinel забезпечив повну видимість подій безпеки, швидке виявлення аномалій та автоматизацію реагування на потенційні інциденти, що доводить відповідність створеної архітектури сучасним стандартам кібербезпеки та її готовність до експлуатації у реальних бізнес-умовах.

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				92
Змн.	Арк.	№ докум.	Підпис	Дата		

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Azure Architecture Center documentation [Електронний ресурс]. - Режим доступу: <https://learn.microsoft.com/azure/architecture/> (дата звернення: 15.02.2026).
2. Microsoft Azure Well-Architected Framework [Електронний ресурс]. - Режим доступу: <https://learn.microsoft.com/azure/architecture/framework/> (дата звернення: 18.02.2026).
3. Azure security fundamentals: Shared responsibility in the cloud [Електронний ресурс].- Режим доступу: <https://learn.microsoft.com/azure/security/fundamentals/shared-responsibility> (дата звернення: 22.02.2026).
4. Cloud Adoption Framework: Cloud service models [Електронний ресурс]. - Режим доступу: <https://learn.microsoft.com/azure/cloud-adoption-framework/strategy/cloud-service-models> (дата звернення: 25.02.2026).
5. Secure VM Access with Azure Bastion | Step-by-Step Deployment & Demo [Електронний ресурс]. - Режим доступу: <https://www.youtube.com/watch?v=Y8B1TOiz7oM> (дата звернення: 28.02.2026).
6. Defense in Depth with Azure Firewall and Azure WAF: Zero Trust in Practice [Електронний ресурс]. - Режим доступу: <https://www.youtube.com/watch?v=xOC0L1Kj3v0> (дата звернення: 03.03.2026).
7. Microsoft Defender for Cloud introduction [Електронний ресурс]. - Режим доступу: <https://learn.microsoft.com/azure/defender-for-cloud/defender-for-cloud-introduction> (дата звернення: 05.03.2026).
8. How to deploy a Spring Boot application to Microsoft Azure [Електронний ресурс]. - Режим доступу: https://www.youtube.com/watch?v=53T_R3xAPTg (дата звернення: 10.03.2026).
9. How to deploy Spring Boot microservices to Azure | Azure Tips and Tricks [Електронний ресурс]. - Режим доступу: https://www.youtube.com/watch?v=L2OngE_V3JA (дата звернення: 12.03.2026).
10. Документація з Бази даних Azure SQL [Електронний ресурс]. - Режим доступу: <https://learn.microsoft.com/uk-ua/azure/azure-sql/database/?view=azuresql> (дата звернення: 15.03.2026).

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				93
Змн.	Арк.	№ докум.	Підпис	Дата		

11. Azure Monitor overview [Електронний ресурс]. - Режим доступу: <https://learn.microsoft.com/azure/azure-monitor/overview> (дата звернення: 19.03.2026).
12. Azure Bastion Explained: Connect to Your VMs Securely [Електронний ресурс]. - Режим доступу: <https://www.youtube.com/watch?v=HEvqDuuEMTk> (дата звернення: 22.03.2026).
13. Azure Cloud Architecture Guide [Електронний ресурс]. - Режим доступу: <https://learn.microsoft.com/en-us/azure/architecture/guide/> (дата звернення: 26.03.2026).
14. Azure cloud security baseline for Network Security [Електронний ресурс]. - Режим доступу: <https://learn.microsoft.com/en-us/azure/security/fundamentals/network-security> (дата звернення: 30.03.2026).
15. Microsoft Defender for Cloud — Cloud Security Posture Management [Електронний ресурс]. - Режим доступу: <https://learn.microsoft.com/en-us/azure/defender-for-cloud/defender-for-cloud-introduction> (дата звернення: 02.04.2026).
16. Microsoft Sentinel overview [Електронний ресурс]. - Режим доступу: <https://learn.microsoft.com/en-us/azure/sentinel/overview> (дата звернення: 06.04.2026).
17. What is Azure Web Application Firewall on Azure Application Gateway? [Електронний ресурс]. - Режим доступу: <https://learn.microsoft.com/en-us/azure/application-gateway/waf-overview> (дата звернення: 09.04.2026).
18. What is Azure Virtual Network? [Електронний ресурс]. - Режим доступу: <https://learn.microsoft.com/en-us/azure/virtual-network/virtual-networks-overview> (дата звернення: 13.04.2026).
19. What is Azure Bastion? [Електронний ресурс]. - Режим доступу: <https://learn.microsoft.com/en-us/azure/bastion/bastion-overview> (дата звернення: 16.04.2026).
20. Azure SQL Database security capabilities [Електронний ресурс]. - Режим доступу: <https://learn.microsoft.com/en-us/azure/azure-sql/database/security-overview> (дата звернення: 20.04.2026).

- 21.What are managed identities for Azure resources? [Електронний ресурс]. - Режим доступу: <https://learn.microsoft.com/en-us/azure/active-directory/managed-identities-azure-resources/overview> (дата звернення: 22.04.2026).
- 22.Log Analytics tutorial — Azure Monitor [Електронний ресурс]. - Режим доступу: <https://learn.microsoft.com/en-us/azure/azure-monitor/logs/log-analytics-tutorial> (дата звернення: 25.04.2026).
- 23.Private Endpoints for Azure Storage overview [Електронний ресурс]. - Режим доступу: <https://learn.microsoft.com/en-us/azure/storage/common/storage-private-endpoints> (дата звернення: 28.04.2026).
- 24.Network security groups — Azure Virtual Network [Електронний ресурс]. - Режим доступу: <https://learn.microsoft.com/en-us/azure/virtual-network/network-security-groups-overview> (дата звернення: 30.04.2026).
- 25.Spring Security Project Reference [Електронний ресурс]. - Режим доступу: <https://spring.io/projects/spring-security> (дата звернення: 02.05.2026).
- 26.Spring Boot Reference Documentation [Електронний ресурс]. - Режим доступу: <https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/> (дата звернення: 05.05.2026).
- 27.Securing a Web Application — Spring Guide [Електронний ресурс]. - Режим доступу: <https://spring.io/guides/gs/securing-web/> (дата звернення: 08.05.2026).
- 28.OWASP Top 10 Privacy Risks Project [Електронний ресурс]. - Режим доступу: <https://owasp.org/www-project-top-ten/> (дата звернення: 11.05.2026).
- 29.OWASP Web Security Testing Guide [Електронний ресурс]. - Режим доступу: <https://owasp.org/www-project-web-security-testing-guide/> (дата звернення: 14.05.2026).
- 30.NGINX documentation: Configuring HTTPS servers [Електронний ресурс]. - Режим доступу: https://nginx.org/en/docs/http/configuring_https_servers.html (дата звернення: 16.05.2026).
- 31.ISO/IEC 27001:2022 — Information security, cybersecurity and privacy protection [Електронний ресурс]. - Режим доступу: <https://www.iso.org/standard/27001> (дата звернення: 19.05.2026).

ДОДАТКИ

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				96
Змн.	Арк.	№ докум.	Підпис	Дата		

Порівняння переваг Microsoft Azure, AWS та Google Cloud для розміщення вебдодатків

Параметр/ Хмарна плат- форма	AWS	GCP	Microsoft Azure
Інтеграція з корпоративними продуктами	Часткова інтеграція, основна увага на Linux	Обмежена інтеграція з Windows	Глибока інтеграція з Windows, Active Directory, SQL Server, Microsoft 365
Обчислювальні сервіси	EC2, Lambda, контейнерні сервіси	Compute Engine, Cloud Functions, контейнерні сервіси	Virtual Machines, Azure Functions, AKS, App Service
Мережні та безпекові сервіси	VPC, Security Groups, WAF, Shield	VPC, Firewall, Cloud Armor	VNet, NSG, Azure Firewall, Bastion, Application Gateway + WAF
Масштабованість і доступність	Висока, багато регіонів і зон доступності	Середня, обмежена кількість регіонів	Висока, понад 60 регіонів, геореплікація, балансування навантаження
Сховище даних	S3, RDS, DynamoDB, Aurora	Cloud Storage, BigQuery, Cloud SQL	Blob Storage, Azure SQL, Cosmos DB, Private Endpoints
Підтримка стандартів безпеки та комплаєнсу	ISO, SOC, PCI DSS	ISO, SOC, PCI DSS	ISO, NIST, CIS, GDPR; інтеграція з Azure Policy і Defender for Cloud
Моніторинг та аналітика	CloudWatch, CloudTrail	Stackdriver, Cloud Logging	Azure Monitor, Log Analytics, Azure Sentinel
Гнучкість ціноутворення	Плата за використання, резервування	Плата за використання, комбінації	Плата за використання, резервування, інтеграція з корпоративними ліцензіями Microsoft

Скрипт автоматизованого розгортання безпечної хмарної інфраструктури

```
#!/bin/bash

echo "=== 0. Реєстрація провайдера SQL ==="

az provider register --namespace Microsoft.Sql

sleep 90

RG=rg-swc-banking-app

LOCATION=swedencentral

VNET=vnet-banking

SUBNET_APPGW=10.0.1.0/28

SUBNET_VM=10.0.2.0/28

SUBNET_BASTION=10.0.4.0/27

SUBNET_PE=10.0.6.0/24

VM_NAME=vm-diploma

SQL_SERVER=sql-banking-$RANDOM

SQL_DB=db-banking

DNS_ZONE=privatelink.database.windows.net

AZURE_AD_USER_EMAIL=kb221_diploma3_kmo@student.ztu.edu.ua

AZURE_AD_OBJECT_ID=1e92f76a-96c0-4261-9d65-0ec110f2bb44

echo "=== 1. Створення групи ресурсів ==="

az group create --name $RG --location $LOCATION

echo "=== 2. Створення віртуальної мережі ==="

az network vnet create \

    --resource-group $RG \

    --name $VNET \

    --address-prefix 10.0.0.0/16
```

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				98
Змн.	Арк.	№ докум.	Підпис	Дата		

echo "=== 3. Створення підмереж ==="

```
az network vnet subnet create --resource-group $RG --vnet-name $VNET --name  
subnet-app --address-prefix $SUBNET_APPGW
```

```
az network vnet subnet create --resource-group $RG --vnet-name $VNET --name  
subnet-vm --address-prefix $SUBNET_VM
```

```
az network vnet subnet create --resource-group $RG --vnet-name $VNET --name  
AzureBastionSubnet --address-prefix $SUBNET_BASTION
```

```
az network vnet subnet create --resource-group $RG --vnet-name $VNET --name  
subnet-private-endpoint --address-prefix $SUBNET_PE
```

echo "=== 4. Створення Віртуальної Машини з System-Assigned Managed Identity
==="

```
az vm create \
```

```
--resource-group $RG \
```

```
--name $VM_NAME \
```

```
--location $LOCATION \
```

```
--image "Ubuntu2404" \
```

```
--size Standard_B2als_v2 \
```

```
--admin-username azureuser \
```

```
--generate-ssh-keys \
```

```
--vnet-name $VNET \
```

```
--subnet subnet-vm \
```

```
--assign-identity '[system]'
```

echo "=== 5. Створення Публічного IP для Bastion ==="

```
az network public-ip create \
```

```
--resource-group $RG \
```

```
--name pip-bastion \
```

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				99
Змн.	Арк.	№ докум.	Підпис	Дата		

```

--location $LOCATION \

--sku Standard

echo "=== 6. Створення Azure Bastion ==="

az network bastion create \

--name bastion \

--resource-group $RG \

--vnet-name $VNET \

--public-ip-address pip-bastion \

--location $LOCATION \

--sku Basic

sleep 60

echo "=== 7. Створення Azure SQL Server ==="

SQL_ADMIN=sqladminuser

SQL_PASS=$(openssl rand -base64 16)

az sql server create \

--name $SQL_SERVER \

--resource-group $RG \

--location $LOCATION \

--admin-user $SQL_ADMIN \

--admin-password $SQL_PASS

sleep 60

echo "=== 8. Назначення Адміна Microsoft Entra ID ==="

az sql server ad-admin create \

--resource-group $RG \

--server $SQL_SERVER \

--display-name "$AZURE_AD_USER_EMAIL" \

```

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				100
Змн.	Арк.	№ докум.	Підпис	Дата		

```

--object-id $AZURE_AD_OBJECT_ID

sleep 60

echo "=== 9. Створення бази даних ==="

az sql db create \

--resource-group $RG \

--server $SQL_SERVER \

--name $SQL_DB \

--service-objective Basic

sleep 60

echo "=== 10. Створення Private Endpoint для SQL ==="

SQL_ID=$(az sql server show \

--name $SQL_SERVER \

--resource-group $RG \

--query id -o tsv)

az network private-endpoint create \

--name pe-sql \

--resource-group $RG \

--vnet-name $VNET \

--subnet subnet-private-endpoint \

--private-connection-resource-id $SQL_ID \

--group-id sqlServer \

--connection-name sql-connection

sleep 15

echo "=== 11. Налаштування Private DNS Зони ==="

az network private-dns zone create \

```

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				101
Змн.	Арк.	№ докум.	Підпис	Дата		

```

--resource-group $RG \

--name $DNS_ZONE

az network private-dns link vnet create \


--resource-group $RG \

--zone-name $DNS_ZONE \

--name dns-link \

--virtual-network $VNET \

--registration-enabled false

az network private-endpoint dns-zone-group create \

--resource-group $RG \

--endpoint-name pe-sql \

--name dns-zone-group \

--private-dns-zone $DNS_ZONE \

--zone-name $DNS_ZONE

az sql server update \

--resource-group rg-swc-banking-app \

--name sql-banking-20870 \

--set publicNetworkAccess="Disabled"

az vm identity assign \

--resource-group rg-swc-banking-app \

--name vm-diploma \

--identities mi-vm-diploma

echo "==== BCE ГОТОВО ===="

echo "SQL SERVER NAME: $SQL_SERVER"

echo "SQL LOCAL ADMIN: $SQL_ADMIN"

```

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				102
Змн.	Арк.	№ докум.	Підпис	Дата		

echo "SQL ADMIN PASSWORD: \$SQL_PASS"

echo "DATABASE NAME: \$SQL_DB"

		Коротун М.О.			ККІКБ.КР.Б – 125 – 06 – ПЗ	Арк.
		Миколайчук В.В.				103
Змн.	Арк.	№ докум.	Підпис	Дата		