Vol. 7, No. 2, 2025

DOI: https://doi.org/10.23939/csn2025.02.114

УДК 004.02; 004.05; 004.09

ОГЛЯД МІКРОСЕРВІСНОЇ АРХІТЕКТУРИ ТА АНАЛІЗ ТИПОВИХ ВРАЗЛИВОСТЕЙ

Д. О. Кудрявцев, Л. З. Мичуда

Національний університет "Львівська політехніка",
кафедра безпеки інформаційних технологій
E-mail: dmytro.o.kudriavtsev@lpnu.ua, lesia.z.mychuda@lpnu.ua

Розглянуто безпеку мікросервісних архітектур з огляду на типові вразливості, що виникають у розподілених системах. Проаналізовано сутність мікросервісного підходу, який, попри свої переваги в масштабованості та гнучкості, породжує нові виклики у сфері кібербезпеки. Основна увага приділяється проблемам управління доступом, налаштуванню мережевих з'єднань та захисту даних під час їх передачі між сервісами, що можуть створювати додаткові вектори атак. Емпіричні дослідження останніх років засвідчують, що приблизно п'ята частина Kubernetes-маніфестів містить щонайменше одну критичну помилку конфігурації, а понад 90 % контейнерних образів мають відомі вразливості, що суттєво розширює площину атаки.

Проведено систематичний огляд наукових досліджень, де детально розглянуто переваги та недоліки мікросервісних систем, а також визначено ключові рекомендації щодо забезпечення безпеки. Окремо висвітлено сучасні технології виявлення вразливостей, зокрема методи статичного та динамічного аналізу, а також підходи до моніторингу контейнеризованих середовищ. Проаналізовано методології аналізу загроз, зокрема традиційні підходи та сучасні моделі моделювання потенційних атак, що дають змогу виявляти слабкі місця системи та оцінювати ризики. Результати дослідження свідчать про необхідність комплексного підходу до безпеки, який передбачає інтеграцію ефективних механізмів аутентифікації, ретельну перевірку налаштувань мережі та постійний моніторинг з використанням сучасних технологій аналізу загроз.

Стаття окреслює сучасний стан проблем безпеки мікросервісних архітектур, пропонує узагальнення отриманих даних та формулює рекомендації для подальших досліджень і вдосконалення заходів захисту в умовах зростаючих кіберзагроз.

Ключові слова: аналіз загроз, аутентифікація/авторизація, вразливості, кібербезпека, контейнеризація, мікросервісна архітектура, моніторинг.

Вступ

Мікросервісна архітектура стала популярним підходом у розробці програмних систем завдяки своїй здатності розподіляти функціональні завдання між незалежними сервісами. Такий підхід дає змогу підвищити масштабованість та гнучкість систем, проте водночас відкриває нові виклики у сфері забезпечення безпеки.

Дослідження зосереджені на виявленні типових вразливостей, пов'язаних із взаємодією окремих сервісів, та розробці методів для їх мінімізації [1]. Дослідження присвячене аналізу основних аспектів безпеки в мікросервісних архітектурах з метою покращення підходів до захисту сучасних інформаційних систем.

Постановка задачі

У сучасних умовах стрімкого розвитку інформаційних технологій мікросервісна архітектура стає дедалі популярнішою завдяки своїй здатності забезпечувати гнучкість, масштабованість та оптимізацію процесів розробки. Водночас перехід від класичних монолітних рішень до розподілених мікросервісів супроводжується появою нових викликів у сфері безпеки.

Розподілений характер мікросервісних систем спричиняє появу унікальних загроз: численні точки взаємодії між сервісами створюють додаткові можливості для атак, ускладнюється контроль за аутентифікацією та авторизацією [2], а також збільшується ризик некоректної конфігурації мережевих протоколів і систем моніторингу. Традиційні підходи до забезпечення безпеки можуть виявитися недостатньо ефективними, що підкреслює необхідність переосмислення існуючих рішень.

Зважаючи на вищевикладене, виникає нагальна потреба у комплексному аналізі мікросервісної архітектури з погляду кібербезпеки. Такий аналіз дасть змогу виявити основні вразливості сучасних систем, розробити ефективні методики їхньої ідентифікації та запобігання [3], що сприятиме створенню надійних та безпечних інформаційних систем у майбутньому.

Метою статті є проведення огляду та порівняльного аналізу мікросервісної архітектури з погляду кібербезпеки для визначення пріоритетних завдань дослідження.

Огляд літературних джерел

Огляд літературних джерел ґрунтується на аналізі низки наукових досліджень, які досить глибоко висвітлюють проблематику безпеки мікросервісної архітектури та специфічні виклики, що виникають під час впровадження розподілених систем.

У статті [4] докладно розглянуто перехід від монолітної до мікросервісної архітектури, зокрема акцентовано на перевагах мікросервісів: простота масштабування, ізоляція збоїв та швидке оновлення. Водночас автори звертають увагу на нові безпекові ризики: широку поверхню атаки, складність моніторингу та важливість людського фактора. Серед ключових рекомендацій виокремлено правильну аутентифікацію й авторизаці., шифрування даних як “у процесі”, так і “в стані спокою”, сегментацію мережі, підхід багаторівневого захисту та ретельне логування подій. Хоча мікросервісна архітектура відкриває широкі можливості для масштабованості й використання різних технологій, її розподілена структура потребує більш ретельного планування і впровадження інтегрованих систем безпеки.

У статті [5] здійснено систематизований огляд 54 наукових публікацій, які стосуються безпеки у мікросервісних архітектурах, зокрема загроз і стратегій захисту для взаємодії між окремими сервісами. Автори наголошують, що мікросервіси через свою розподілену природу мають ширше поле атаки порівняно з монолітними системами. Крім того, різноманітність інструментів (контейнери, оркестрація, різні мови програмування) ще більше ускладнює проблему безпеки. Під час аналізу основними ризиками виявилися: розширений периметр атаки та недостатня ізоляція між сервісами, вразливості контейнерів та платформ оркестрації, проблеми з аутентифікацією й авторизацією, а також брак ефективних механізмів моніторингу і виявлення загроз. Натомість найпоширенішими підходами до зменшення цих ризиків є використання DevSecOps-практик, принципу “zero trust”, багаторівневого захисту, шифрування, контроль доступу на рівні кожного сервісу, а також чітке розмежування мереж і відстеження усіх транзакцій.

У статті [6] окреслено механізм динамічної оцінки довіри для мікросервісних архітектур, що використовують контейнеризацію та сервісну сітку. Основна мета – виявляти вразливості в реальному часі та коригувати “рівень довіри” мікросервісів, особливо у складних сценаріях 5G/6G. Автори інтегрують сканування вразливостей (на прикладі Trivy) із сервісною сіткою (Istio) в Kubernetes-кластері, щоб у режимі реального часу зчитувати відомі вразливості, зменшувати “рівень довіри” пошкоджених сервісів і, якщо буде потреба, підвищувати його після виправлень. Тестування проведено на відкритому бенчмарку Train Ticket, де автоматично розраховані показники демонст-

рують, як зміна бібліотек/модулів впливає на безпеку та “довіру” до мікросервісів. Отож праця пропонує метод виявлення й усунення вразливостей на етапі експлуатації, що сприяє підвищенню стійкості мікросервісної архітектури, зокрема в середовищах 5G/6G.

Актуальність безпекових завдань мікросервісної архітектури

Зі зростанням популярності мікросервісної архітектури традиційні підходи до забезпечення безпеки вже не відповідають сучасним вимогам. Розподілений характер мікросервісних систем створює додаткові точки потенційного нападу, що ускладнює контроль за доступом і потребує адаптивних рішень для виявлення та запобігання загрозам. Емпіричне дослідження 2039 Kubernetes-маніфестів зафіксувало 1051 помилкову конфігурацію в 11 категоріях, зокрема відсутність securityContext і лімітів ресурсів, що демонструє масштабність проблеми на рівні інфраструктурного коду [7].

Сучасні інформаційні системи працюють у режимі постійного оновлення, що змушує розробників враховувати не лише функціональність, а й безперервний моніторинг безпеки. Інтенсивне використання API, численні взаємодії між сервісами та часті зміни в конфігурації підсилюють ризик виникнення вразливостей, які можуть використати зловмисники для несанкціонованого доступу до критичних даних. Наприклад, у попередньому вимірюванні 100 публічних REST-API 25 виявилися схильними до mass assignment, а в шести з них підтверджено дев’ять реальних вразливих операцій; окремі методики тестування цієї вразливості презентовано й на ICSE’23 [8].

У цьому контексті питання безпеки набуває особливої актуальності, адже ефективна система захисту є невід’ємною складовою успішного функціонування будь-якого сучасного підприємства. Показово, що навіть за наявності рекомендованих практик (наприклад, CIS для Kubernetes) 17,9–18,0 % реальних конфігураційних файлів у відкритих та корпоративних репозиторіях містять принаймні одне порушення; водночас найкращі з досліджених SAST-інструментів демонструють обмежену точність (precision до 0,41–0,43) і recall до 0,53–0,65, що підкреслює потребу в комбінації процедурного контролю, оглядів коду й runtime-моніторингу [9].

Отже, дослідження вразливостей та розробка нових методик захисту мають важливе значення для підтримки високого рівня кібербезпеки в умовах стрімких технологічних змін, що робить цю тему актуальною для сучасної науки та практики розробки інформаційних систем.

Огляд сучасних інструментів та технологій для виявлення вразливостей

У сучасних умовах розробки програмних систем забезпечення безпеки набуває першочергового значення, особливо у мікросервісних архітектурах, де численні незалежні компоненти взаємодіють між собою. Для своєчасного виявлення потенційних загроз розробники використовують різноманітні рішення, які працюють як на етапі написання коду, так і під час експлуатації системи.

На початкових стадіях створення програмного продукту широко застосовуються засоби статичного аналізу, такі як SonarQube, Checkmarx чи Fortify. Вони дають змогу автоматично перевіряти вихідний код, виявляючи потенційні недоліки та вразливості ще до переходу продукту у фазу тестування чи експлуатації [10].

Під час функціонування системи важливим є використання інструментів динамічного аналізу. Програми типу OWASP ZAP та Burp Suite надають можливість моделювати атаки та аналізувати поведінку сервісів у режимі реального часу, що допомагає виявити слабкі місця у взаємодії між компонентами [11].

Не менш актуальним є аналіз безпеки контейнеризованих середовищ. Зважаючи на те, що сучасні мікросервісні рішення часто розгортаються за допомогою Docker чи Kubernetes, спеціалізовані інструменти, такі як Aqua Security та Clair, дозволяють перевіряти образи контейнерів щодо відомих вразливостей і невідповідностей [12].

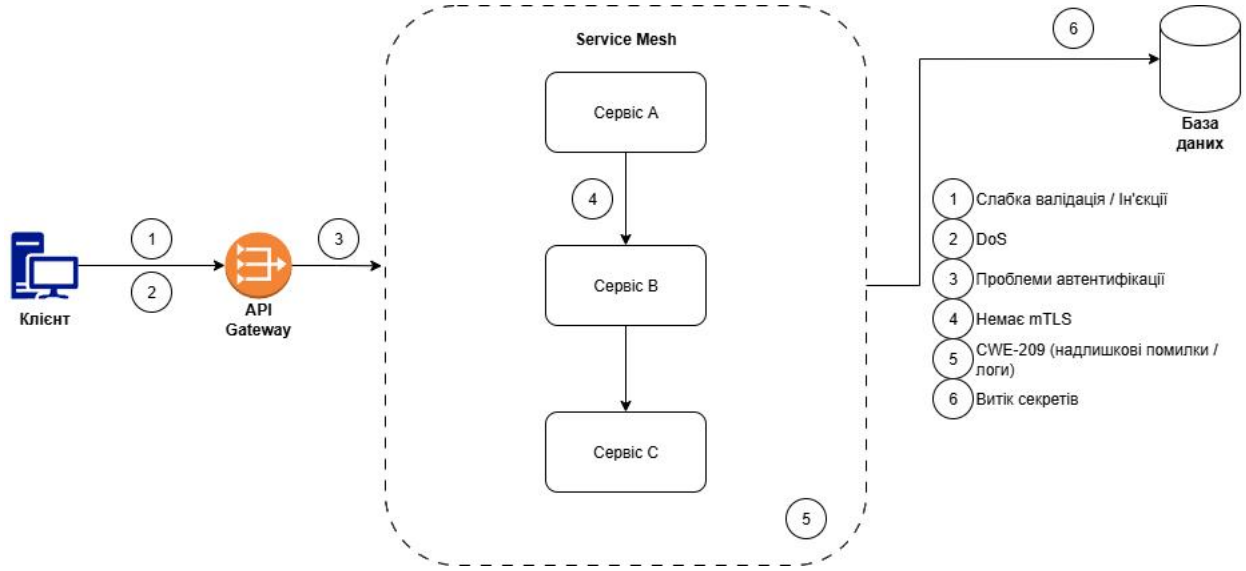
Крім того, системи моніторингу, наприклад Prometheus у поєднанні з Grafana, сприяють виявленню аномалій у роботі сервісів [13]. Використання методів машинного навчання для аналізу

поведінкових патернів допомагає не тільки вчасно ідентифікувати потенційні загрози, а й прогнозувати розвиток інцидентів.

Отож сучасні технології для виявлення вразливостей забезпечують комплексний підхід до безпеки мікросервісних систем, поєднуючи автоматизований аналіз коду, динамічне тестування, перевірку контейнерних середовищ та постійний моніторинг. Це дає змогу своєчасно виявляти та усувати потенційні загрози, підтримуючи високий рівень надійності та стійкості систем у мінливих умовах сучасного кіберпростору.

Аналіз типових вразливостей мікросервісної архітектури

Мікросервісна архітектура здобула широке застосування завдяки своїй здатності розподіляти функціональність на окремі незалежні компоненти, що дозволяє системам бути більш гнучкими та масштабованими. Проте така модульність відкриває нові можливості для зловмисників, оскільки кожен сервіс може стати окремою точкою для потенційної атаки (рисунок).



Потенційні точки атаки

Однією з ключових проблем є управління доступом між мікросервісами. Різні сервіси можуть мати власні механізми аутентифікації та авторизації, що створює ризик виникнення розбіжностей у політиках безпеки [14]. У випадках, коли налаштування доступу здійснюється некоректно, це може дозволити несанкціонований доступ до критично важливих ресурсів або даних. Подібна ситуація поглиблюється, коли комунікація між сервісами не інтегрована у єдину систему контролю, що робить її вразливою до зловмисних маніпуляцій.

Ще однією суттєвою вразливістю є помилки в конфігурації мережевих з'єднань [15]. Розподілена природа системи означає, що дані передаються через численні канали зв'язку, що підвищує ймовірність виникнення недоліків у налаштуваннях мережевих протоколів. Використання застарілих або неадаптованих протоколів, а також неправильна конфігурація мережевих шлюзів і брандмауерів можуть створити додаткові точки входу для атак, зокрема для атак типу "відмова в обслуговуванні" (DoS) або для несанкціонованого доступу до даних.

Передача даних між сервісами також потребує особливої уваги. Неправильне впровадження криптографічних алгоритмів або відсутність належного шифрування під час обміну інформацією може призвести до перехоплення та компрометації даних [16]. Особливо небезпечно, коли сервіси обмінюються чутливою інформацією через мережі з низьким рівнем захисту, що дає змогу зловмисникам отримати доступ до конфіденційних даних.

Не менш важливим є аспект обробки помилок та ведення логів. Надмірно докладні повідомлення про помилки, які містять технічні деталі, можуть стати джерелом інформації для потенційних атак [17]. Такі логи можуть розкривати структуру системи, використовувані технології та внутрішні процеси, що суттєво полегшує завдання зловмисникам, які намагаються знайти слабкі місця в архітектурі.

Також варто зазначити, що тісна взаємодія між мікросервісами створює додаткові ризики. Компрометація одного із сервісів може стати ланцюговою реакцією, яка вплине на весь комплекс системи. Через численні точки взаємодії вразливість одного компонента може стати відправною точкою для подальшого розповсюдження атаки, що ускладнює виявлення та локалізацію проблеми.

Отже, детальний аналіз типових вразливостей мікросервісної архітектури свідчить про необхідність комплексного підходу до безпеки. Сучасні системи мають поєднувати ретельний аудит налаштувань, інтегровані механізми управління доступом, ефективні засоби шифрування та безпечне логування [18]. Лише завдяки такому всебічному підходу можливо забезпечити високий рівень захищеності розподілених систем у сучасних умовах постійно зростаючих кіберзагроз.

Огляд методологій аналізу загроз

Методології аналізу загроз відіграють важливу роль у забезпеченні безпеки мікросервісних архітектур, дозволяючи ідентифікувати потенційні вразливості та оцінити ризики для системи. Серед найпоширеніших підходів варто виділити як класичні, так і сучасні моделі, що дають змогу комплексно підійти до проблематики кіберзахисту.

Однією з базових методологій є аналіз ризиків, який передбачає ідентифікацію активів, потенційних загроз і оцінку можливих наслідків їх реалізації. Такий підхід дозволяє ранньо визначити критичні компоненти системи та зосередити зусилля на їхньому захисті [19]. Методика може доповнюватися використанням стандартних підходів, таких як OCTAVE або NIST SP 800-30, що надають структуровані рекомендації для проведення аналізу ризиків.

Інший важливий напрям – це моделювання загроз. Методики, як-от STRIDE або PASTA, дають змогу системно розглядати можливі сценарії атак [20], розбиваючи їх на окремі елементи, такі як підробка, порушення цілісності, відмова в обслуговуванні тощо. Цей підхід допомагає не лише виявити слабкі місця, а й розробити конкретні заходи для їхнього усунення. Особливу цінність має адаптація цих методів до специфіки мікросервісних систем, де взаємодія між численними компонентами ускладнює процес моніторингу та контролю за безпекою.

Крім того, сучасні підходи передбачають застосування атакових дерев, які являють собою графічну модель можливих шляхів досягнення зловмисними акторами своїх цілей [21]. Аналіз таких дерев дає змогу детально розглянути кожен можливий вектор атаки, оцінити складність його реалізації та потенційний вплив на систему.

Важливим аспектом є інтеграція автоматизованих інструментів, що підтримують процес аналізу загроз [22]. Системи, що базуються на машинному навчанні, здатні виявляти аномальні патерни поведінки в реальному часі, що значно підвищує оперативність реагування на нові види загроз. Це дає змогу не лише прогнозувати можливі інциденти, а й адаптувати захисні заходи відповідно до змін у середовищі.

Отже, огляд методологій аналізу загроз свідчить про різноманіття підходів до вирішення проблем безпеки в мікросервісних архітектурах. Поєднання традиційних методів оцінки ризиків із сучасними моделями моделювання загроз та автоматизованими засобами моніторингу дозволяє отримати більш повну картину потенційних вразливостей [23] та розробити ефективну стратегію їхнього усунення.

Результати дослідження

Проведений аналіз літературних джерел, огляд сучасних технологій виявлення вразливостей та розгляд методологій аналізу загроз дали змогу виявити кілька ключових аспектів, які потребують додаткової уваги під час забезпечення безпеки мікросервісних архітектур.

По-перше, дослідження показало, що проблема розподіленості систем значно ускладнює процес управління доступом. Розбіжності в реалізації механізмів аутентифікації та авторизації створюють ризик виникнення неузгодженостей, що може призвести до несанкціонованого доступу до окремих сервісів. Систематичний аналіз виявив, що багато з існуючих рішень недостатньо інтегровані, що спричиняє потенційні вектори атак.

По-друге, особлива увага була приділена проблемам налаштування мережевих з'єднань. Неправильна конфігурація протоколів і шлюзів може стати точкою входу для зловмисників. Було встановлено, що застосування сучасних стандартів безпеки та регулярний аудит мережевих налаштувань є критично важливими для зниження ризиків.

Також дослідження свідчить про те, що недостатній рівень шифрування та неправильна обробка помилок можуть розкрити суттєву інформацію про внутрішню структуру системи. З цього випливає необхідність впровадження більш жорстких заходів безпеки під час передачі даних між сервісами, а також удосконалення механізмів логування з урахуванням вимог до мінімізації витоків інформації.

Методології аналізу загроз, зокрема підходи типу STRIDE, PASTA та використання атакових дерев, довели свою ефективність у виявленні потенційних векторів атак. Результати дослідження підкреслюють, що інтегроване застосування традиційних методів оцінки ризиків із сучасними технологіями моніторингу дає змогу отримати більш повну картину можливих загроз (таблиця).

Порівняльний аналіз підходів до виявлення/оцінки загроз

Підхід	Функція	Переваги
Оцінка ризиків (OCTAVE / NIST SP 800-30)	Пріоритизація активів, загроз і впливів	Узгодження з бізнес-цілями, зрозумілі критерії
STRIDE	Каталог типових загроз за категоріями	Простий старт, добре для рев'ю дизайну
PASTA	Сценарії атак з прив'язкою до бізнес-цілей і ризиків	Глибина й пріоритизація, зв'язок "загроза-контроль"
Атакові дерева	Візуалізація шляхів та ланцюгів атак	Добре показує "вузькі місця" і латеральний рух
Автоматизовані інструменти (SAST/DAST/SCA, APM/ML)	Безперервне виявлення вразливостей/аномалій	Масштабованість, інтеграція в CI/CD, швидкий фідбек

Отже, результати дослідження вказують на необхідність комплексного підходу до забезпечення безпеки мікросервісних архітектур, який охоплює регулярний аудит налаштувань, інтеграцію єдиних механізмів аутентифікації та авторизації, а також використання сучасних інструментів аналізу загроз і автоматизованого моніторингу. Ці висновки створюють надійну базу для подальшого вдосконалення підходів до управління безпекою в умовах постійно зростаючих кіберзагроз.

Висновки

Проведене дослідження дало змогу сформувати цілісне уявлення про безпекові виклики, що виникають у мікросервісних архітектурах. Аналіз літературних джерел, огляд сучасних технологій виявлення вразливостей та розгляд методологій аналізу загроз свідчать про те, що розподіленість систем значно ускладнює забезпечення належного рівня захисту.

Ключовим аспектом є недосконалість механізмів управління доступом, що призводить до можливості несанкціонованого проникнення через численні точки взаємодії. Невідповідність налаштувань мережевих з'єднань та проблеми з шифруванням даних створюють додаткові вектори атак, що підкреслює необхідність впровадження єдиних стандартів безпеки для всіх компонентів системи.

Методології аналізу загроз, такі як STRIDE, PASTA та використання атакових дерев, демонструють свою ефективність у виявленні слабких місць системи. Інтеграція традиційних підходів із

сучасними автоматизованими засобами моніторингу дає змогу створити більш адаптивну та гнучку стратегію управління ризиками.

Отже, комплексний підхід до безпеки мікросервісних архітектур, який передбачає регулярний аудит налаштувань, впровадження єдиних механізмів аутентифікації та авторизації, а також застосування сучасних технологій аналізу загроз, є ключовим фактором для підвищення стійкості систем в умовах сучасних кіберзагроз. Подальші дослідження у цьому напрямі можуть сприяти вдосконаленню методів захисту, що дозволить забезпечити надійний функціонал розподілених систем навіть за постійно зростаючих викликів кібербезпеки.

Список літератури

1. Jayalath, R. K., Ahmad, H., Goel, D., Syed, M. S., & Ullah, F. (2024). Microservice Vulnerability Analysis: A Literature Review with Empirical Insights. DOI: https://doi.org/10.48550/arXiv.2408.03960.

2. Hannousse, A., & Yahiouche, S. (2021). Securing microservices and microservice architectures: A systematic mapping study. Computer Science Review, 41, 100415. DOI: https://doi.org/10.48550/arXiv.2003.07262.

3. Nasab, A. R., Shahin, M., Raviz, S. A. H., Liang, P., Mashmool, A., & Lenarduzzi, V. (2022). An Empirical Study of Security Practices for Microservices Systems. DOI: https://doi.org/10.48550/arXiv.2112.14927.

4. Mateus-Coelho, Nuno & Cruz-Cunha, Maria & Ferreira, Luis. (2021). Security in Microservices Architectures. Procedia Computer Science. 181. 1225-1236. 10.1016/j.procs.2021.01.320. DOI: https://doi.org/10.1016/j.procs.2021.01.320.

5. Haindl, Philipp & Kochberger, Patrick & Sveggen, Markus. (2024). A Systematic Literature Review of Inter-Service Security Threats and Mitigation Strategies in Microservice Architectures. IEEE Access. PP. 1–1. 10.1109/ACCESS.2024.3406500. DOI: http://dx.doi.org/10.1109/ACCESS.2024.3406500.

6. Alboqmi, Rami & Gamble, Rose. (2025). Enhancing Microservice Security Through Vulnerability-Driven Trust in the Service Mesh Architecture. Sensors. 25. 914. 10.3390/s25030914. DOI: http://dx.doi.org/10.3390/s25030914.

7. Rahman, A., Shamim, S. I., Bose, D. B., & Pandita, R. (2023). Security misconfigurations in open source Kubernetes manifests: An empirical study. ACM Transactions on Software Engineering and Methodology, 32(4), Article 99. DOI: https://doi.org/10.1145/3579639.

8. Mazidi, A., Corradini, D., & Ghafari, M. (2024, June). Mining REST APIs for potential mass assignment vulnerabilities. In Proceedings of the 28th International Conference on Evaluation and Assessment in Software Engineering (EASE '24) (pp. 369–374). ACM. DOI: https://doi.org/10.1145/3661167.3661204.

9. Shamim, S. I., Hu, H., & Rahman, A. (2025). On prescription or off prescription? An empirical study of community-prescribed security configurations for Kubernetes. In Proceedings of the IEEE/ACM 47th International Conference on Software Engineering (ICSE '25) (pp. 2432–2444). IEEE Press. DOI: https://doi.org/10.1109/ICSE55347.2025.00170.

10. Piskachev, G., Becker, M. & Bodden, E. (2023). Can the configuration of static analyses make resolving security vulnerabilities more effective? – A user study. Empir Software Eng 28, 118. DOI: https://doi.org/10.1007/s10664-023-10354-3.

11. Ihor Sasovets. (2025). Dynamic Application Security Testing: The Ultimate Guide. TechMagic article. URL: https://www.techmagic.co/blog/dast.

12. Javed, Omar & Toor, Salman. (2021). Understanding the Quality of Container Security Vulnerability Detection Tools. 10.48550/arXiv.2101.03844. DOI: http://dx.doi.org/10.48550/arXiv.2101.03844.

13. Jani, Yash. (2024). Unified Monitoring for Microservices: Implementing Prometheus and Grafana for Scalable Solutions. Journal of Artificial Intelligence, Machine Learning and Data Science. 2. 848-852. 10.51219/JAIMLD/yash-jani/206. DOI: http://dx.doi.org/10.51219/JAIMLD/yash-jani/206.

14. Almeida, Murilo & Canedo, E.D.. (2022). Authentication and Authorization in Microservices Architecture: A Systematic Literature Review. Applied Sciences. 12. 3023. 10.3390/app12063023. DOI: https://doi.org/10.3390/app12063023.

15. Shahzad Bhatti. (2024). Security Challenges in Microservice Architecture. Medium article. URL: https://shahbhat.medium.com/security-challenges-in-microservice-architecture-a6065dbedce9.

16. Pereira-Vale A., Fernandez E. B., Monge R., Astudillo H., Márquez G. (2021). Security in microservice-based systems: A Multivocal literature review. Computers & Security, 103, 102200. DOI: https://doi.org/10.1016/j.cose.2021.102200.

17. CWE-209 Description. Generation of Error Message Containing Sensitive Information. URL: https://cwe.mitre.org/data/definitions/209.html.

18. OWASP. Microservices Security - OWASP Cheat Sheet Series. URL: https://cheatsheetseries.owasp.org/cheatsheets/Microservices_Security_Cheat_Sheet.html.

19. Oluwatobiloba, Anifowose & Adelusi, Joshua. (2025). Security Challenges in Cloud-Native Microservices: A Risk Assessment and Mitigation Framework. URL: https://www.researchgate.net/publication/388956545_Security_Challenges_in_Cloud-Native_Microservices_A_Risk_Assessment_and_Mitigation_Framework.

20. Bipin Gajbhiye, Akshun Chhapola & Shalu Jain. (2023). Advanced Threat Modeling Techniques for Micro services Architectures. URL: https://ijnrd.org/papers/IJNRD2304737.pdf.

21. National Cyber Security Centre of United Kingdom. (2023). Using Attack Trees to Understand Cyber Security Risk. URL: https://www.ncsc.gov.uk/collection/risk-management/using-attack-trees-to-understand-cyber-security-risk.

22. Abdulsatar M., Ahmad H., Goel D., Ullah F. (2024). Towards Deep Learning Enabled Cybersecurity Risk Assessment for Microservice Architectures. DOI: https://doi.org/10.48550/arXiv.2403.15169.

23. Leines-Vite L., Pérez-Arriaga J. C., Limón X. (2021). Information and Communication Security Mechanisms For Microservices-based Systems. International Journal of Network Security & Its Applications, 13(6), 85–103. DOI: https://doi.org/10.48550/arXiv.2111.01218.

OVERVIEW OF MICROSERVICE ARCHITECTURE AND ANALYSIS OF TYPICAL VULNERABILITIES

D. O. Kudriavtsev, L. Z. Mychuda

Lviv Politechnic National University,
Department of Information Technology Security

The article examines the security of microservice architectures in the context of common vulnerabilities that arise in distributed systems. The authors analyze the essence of the microservice approach, which, despite its advantages in scalability and flexibility, introduces new challenges in the field of cybersecurity. The main focus is on issues of access management, network configuration, and data protection during transmission between services, which can create additional attack vectors. Empirical studies in recent years indicate that about one-fifth of Kubernetes manifests contain at least one critical configuration error, while over 90 % of container images have known vulnerabilities, significantly expanding the attack surface.

The article provides a systematic review of scientific research, detailing the advantages and disadvantages of microservice systems, and identifies key recommendations for ensuring security. Special attention is given to modern vulnerability detection technologies, including static and dynamic analysis methods, as well as approaches to monitoring containerized environments.

The authors also examine threat analysis methodologies, including traditional approaches and modern models for simulating potential attacks, which help identify system weaknesses and assess risks. The results of the study highlight the need for a comprehensive security approach that integrates effective authentication mechanisms, careful network configuration review, and continuous monitoring using advanced threat analysis technologies.

Thus, the article outlines the current state of security issues in microservice architectures, summarizes the obtained findings, and formulates recommendations for further research and the enhancement of protection measures in the face of growing cyber threats.

Keywords: Microservice architecture, cybersecurity, vulnerabilities, threat analysis, monitoring, containerization, authentication/authorization.