УДК 004.4 DOI: https://doi.org/10.30837/IYF.IIS.2024.326

АНАЛІЗ СУЧАСНИХ ПІДХОДІВ ДО ЗАБЕЗПЕЧЕННЯ БЕЗПЕКИ В МІКРОСЕРВІСНИХ АРХІТЕКТУРАХ

Коломойцев П. А.
Науковий керівник – д.т.н проф. Єрохін А. Л.
Харківський національний університет радіоелектроніки, каф. ПІ
м. Харків, Україна
e-mail: pavlo.kolomoitsev@nure.ua

Researching various approaches and tools aimed at ensuring security in microservices architectures is an important topic in modern information security. This issue becomes particularly relevant due to the widespread adoption of containerization and orchestration in software development.

Мікросервіс – це техніка розробки програмного забезпечення, яка є варіацією стилю сервісно-орієнтованої архітектури або SOA (Service oriented architecture). Нажаль, немає єдиної визначеної концепції для мікросервісів. Мартін Фаулер сказав наступне: "Коротко кажучи, мікросервісний архітектурний стиль – це підхід до розробки одного додатка як набору невеликих сервісів, кожен з яких працює у власному процесі та взаємодіє за допомогою легких механізмів, часто через API ресурсів HTTP" [1].

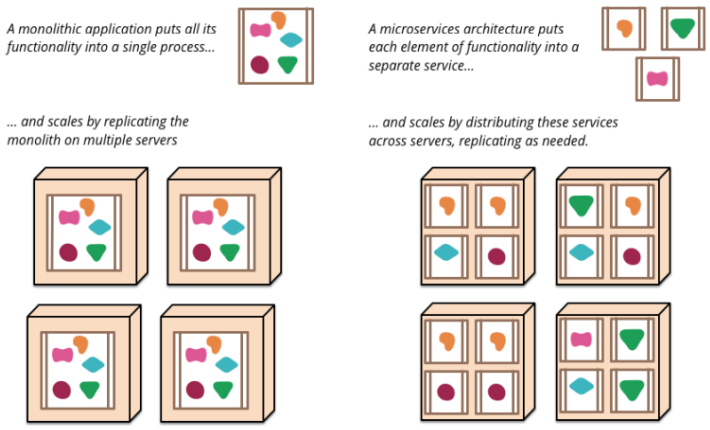

Рисунок 1 – Мікросервісна архітектура против монолітної [2]

Зазвичай, коли ми говоримо про мікросервіси, ми маємо на увазі самостійні частини функціональності бізнесу з чіткими інтерфейсами. Загальна ознака мікросервісів полягає в тому, що вони зазвичай будуються як додатки, орієнтовані на хмарні технології. Додаток на основі мікросервісів має структуру у вигляді набору слабо зв'язаних сервісів, які побудовані навколо бізнес-можливостей. Кожен сервіс розгортається незалежно, і ми прагнемо зменшити централізоване керування сервісами. Ще один цікавий аспект полягає в тому, що кожен сервіс може бути

написаний на різних мовах програмування та використовувати різні типи сховищ даних.

Під час дослідження методів та підходів до забезпечення безпеки в мікросервісних архітектурах, де використовується контейнеризація та оркестрація, важливо враховувати широкий спектр факторів. До таких факторів можуть відноситися технічні характеристики контейнерів, механізми автоматизації та керування ресурсами, а також алгоритми обнаруження та реагування на загрози безпеки.

Розділення служб за ознаками безпеки є важливим для забезпечення надійного безпекового стану у мікросервісних архітектурах. Шляхом класифікації служб за типом доступу (таким як публічний, внутрішній і т. д.) та необхідними привілеями організації можуть ефективно впроваджувати Принцип Мінімальних Привілеїв (PoLP).

PoLP визначає, що кожен процес, програма або користувач повинен мати лише мінімальні привілеї, необхідні для виконання своєї визначеної функції. Наприклад, при створенні облікового запису користувача для взаємодії з базою даних слід уникати надання адміністративних привілеїв. Також розробники, які працюють над проектами старого зразка, не повинні мати доступ до особистих записів, якщо це необхідно для їх завдань. PoLP є синонімом таких принципів, як Принцип Мінімальних Привілеїв (PoMP) або Принцип Мінімальної Авторитетності (PoLA) і широко визнаний як найкраща практика в інформаційній безпеці.

Переваги дотримання PoLP різноманітні:

– Підвищена безпека: Незаконний доступ до мільйонів файлів NSA Едвардом Сноуденом підкреслює ризики, пов'язані з наданням занадто великих привілеїв. Обмеження адміністративних привілеїв може значно зменшити такі безпекові порушення [3].

– Зменшення поверхні атак: Компрометація 70 мільйонів облікових записів клієнтів магазину Target була сприяна HVAC-підрядником, якому дозволено завантажувати виконувані файли. Дотримання PoLP допомагає мінімізувати потенційну поверхню атак і підвищує загальну безпеку.

– Обмеження поширення вредоносного програмного забезпечення: Вредоносне програмне забезпечення, яке проникає в систему, створена відповідно до принципу мінімальних привілеїв, часто міститься лише в певних модулях, що зменшує ризик поширення.

– Покращена стабільність системи: PoLP сприяє стабільності системи, обмежуючи вплив змін у окремих модулях та мінімізуючи перерви в роботі всієї архітектури.

– Покращена готовність до аудиту: Впровадження принципу PoLP спрощує аудитові процеси, зменшуючи обсяг і складність оцінок та забезпечуючи відповідність вимогам регуляторних органів [4].

Окрім розділення на основі характеристик безпеки, перевірка вхідних даних є важливою для захисту мікросервісних архітектур від різних

вразливостей і атак. Всі вхідні запити, відповіді, повідомлення та події повинні пройти ретельну перевірку для виявлення та усунення потенційних загроз безпеці.

Основний висновок з дослідження полягає в тому, що використання принципу мінімальних привілеїв (PoLP) є важливим для забезпечення безпеки в мікросервісних архітектурах. При дотриманні цього принципу ризики, пов'язані з наданням занадто великих привілеїв, значно зменшуються. Також виявлено, що використання PoLP допомагає уникнути ситуацій, коли атакувальники можуть отримати доступ до всього додатку через вразливість у одному з мікросервісів.

Крім того, перевірка вхідних даних перед їх обробкою виявилася ефективним методом для захисту мікросервісних архітектур від різних вразливостей і атак. Це дозволяє запобігти атакам, які базуються на введенні некоректних даних, таких як SQL-ін'єкції або кросс-сайтові скрипти.

Загалом, дослідження показало, що впровадження принципу мінімальних привілеїв, перевірка вхідних даних та контроль версій залежностей є ключовими аспектами забезпечення безпеки в мікросервісних архітектурах.

Список використаних джерел**:**

1. U. Dinesh Kumar, David Nowicki, Dinesh Verma, J E Ramírez-Márquez. Reliability and maintainability allocation to minimize total cost of ownership in a series-parallel system. Proceedings of the Institution of Mechanical Engineers Part O Journal of Risk and Reliability. 2007. P. 113–140.

2. Micro-Service Architecture. Medium: веб-сайт. URL: https://champikamendis-cm.medium.com/micro-service-architecture-821e3c6c7826 (дата звернення: 21.02.2024).

3. Real-time Performance Profiling & Analytics for Microservices using Apache Spark. Medium : веб-сайт. URL: https://medium.com/@a0x8o/real-time-performance-profiling-analytics-for-microservices-using-apache-spark-96d026083021 (дата звернення: 21.02.2024).

4. Filatov V.O., Yerokhin A.L., Zolotukhin O.V., Kudryavtseva M.S. Methods of intellectual analysis of processes in medical information systems. Information Extraction and Processing. 2020. 48(124), P. 92–98. DOI:https://doi.org/10.15407/vidbir2020.48.092.