УДК 004.738.1.056(043.2)

А.С. Бойченко
(Національний Авіаційний Університет, Україна)

Можливість використання мікросервісної архітектури в WEB-додатках для захисту інформації

Детально розглянуто підхід до проектування WEB-додатків, як частини комплексної системи захисту інформації. Наведено схему мікросервісної архітектури. Розглянуто можливість використання в захищених системах.

Мікросервісна архітектура – архітектурний шаблон програмного забезпечення, який складається з малих частин (модулів), які виконують кожен свою конкретну функцію. Кожна частина програмного забезпечення може бути реалізована різними мовами програмування та технологіями зберігання даних. Модулі взаємодіють між собою за допомогою протоколів передачі даних, таких як HTTP, HTTPS, TLS/SSL, тощо. Схема мікросервісної архітектури WEB-додатку наведена на рис. 1.

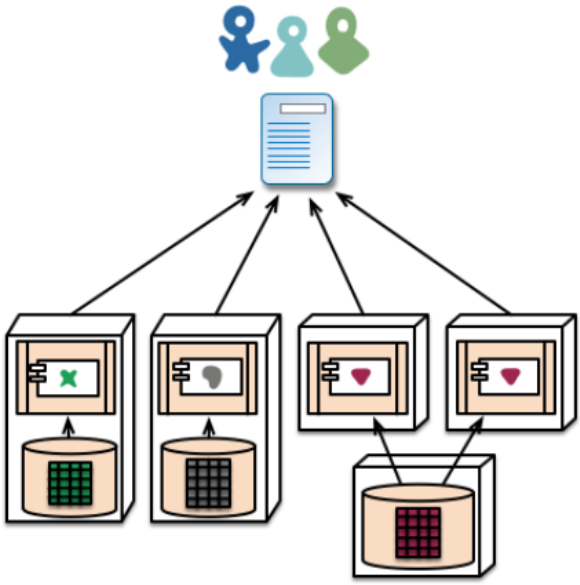

Рис. 1. Схема мікросервісної архітектури

Головними перевагами мікросервісної архітектури є:

- гнучкість, оскільки кожен сервіс є самостійним та зміни у ньому не впливають на інші сервіси;

- незалежне масштабування;
- ефективність використання ресурсів комп'ютерної системи, на якій розгорнуто WEB-додаток;
- відсутність прив'язки до однієї мови програмування та технології розробки;

Мікросервісна архітектура має також свої недоліки:

- складність швидкого розгортання за рахунок своєї будови;
- складність забезпечення безпеки WEB-додатку;
- підвищення мережевої затримки під час взаємодії з WEB-додатком;
- незалежність сервісів призводить до дублювання програмного коду.

У неструктурованому розподіленому середовищі дуже легко втратити контроль над множиною мікросервісів. З цієї причини виконується контроль мікросервісів, структурування їх за рівнями (Рис. 2).

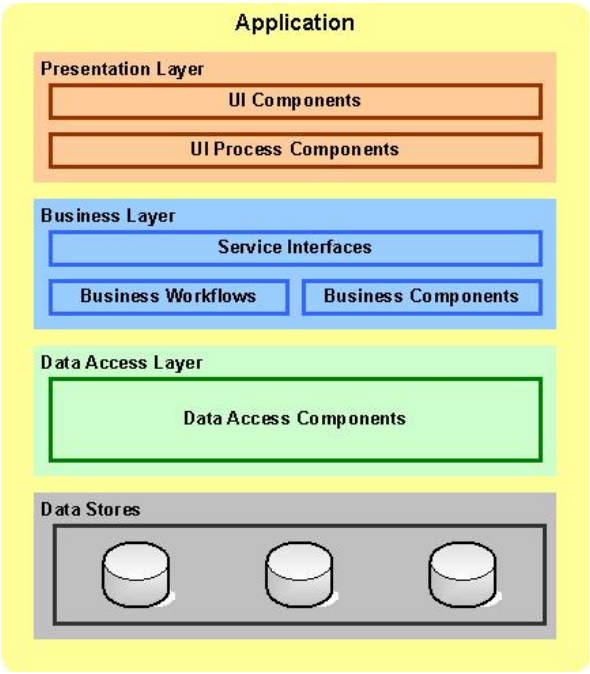

Рис. 2. Схема рівнів WEB-додатку

Під час проектування WEB-додатку, виділяють наступні рівні:

- рівень відображення (Presentation Layer) – відповідає за формування інтерфейсу та відображення інформації користувачеві;
- рівень бізнес-логіки (Business Layer) – відповідає за обробку та виконання функцій, які надходять від користувача. На цьому рівні розміщують більшість мікросервісів;
- рівень доступу до даних (Data Access Layer) – виконує розмежування доступу інших мікросервісів до бази даних, де зберігається інформація;
- рівень даних (Data Stores) – виконує функції зберігання та читання інформації (база даних).

З точки зору захисту інформації, мікросервісна архітектура передбачає розділення функціональності WEB-додатку на дрібніші частини, що зменшує ймовірність повної втрати функціональності програмного забезпечення в разі збою в одному з сервісів.

Однак, мікросервісна архітектура може створити додаткові ризики. Оскільки кожен сервіс може мати власну базу даних та інші залежності, це може призвести до збільшення кількості векторів атаки, які можуть бути використані для злому системи.

Крім того, збільшення кількості сервісів також означає збільшення кількості точок доступу до системи. Тому, якщо кожен сервіс не буде належним чином захищений, це може стати потенційним джерелом вразливостей та стати причиною несанкціонованого доступу до комп'ютерної системи.

Зважаючи на всі переваги та недоліки, мікросервісна архітектура може бути використана в комплексних системах захисту інформації. Однак, необхідно покращити заходи безпеки під час її реалізації, а саме:

- захист мережі: необхідно належним чином налаштувати мережу, в якій розгорнуто мікросервіси, та обмежити доступ до сервісів ззовні, а також захистити мережеві протоколи від атак;
- захист даних: кожен сервіс має мати відповідні механізми захисту даних, такі як шифрування, хешування, захист від ін'єкцій та інших атак на дані;
- моніторинг та логування: важливо мати систему моніторингу та логування для кожного сервісу, щоб вчасно виявляти та реагувати на потенційні загрози та вразливості;
- резервне копіювання: важливо мати резервну копію сервісів на випадок збоїв, які унеможливлюють їх роботу або на випадок відновлення після атак на комплексну систему в цілому;
- своєчасне оновлення: важливо регулярно оновлювати сервіси, щоб покращити захист від відомих вразливостей та атак.

Список літератури

1. L. Lamport, "The Implementation of Reliable Distributed Multiprocess Systems", 1978 [Режим доступу] http:// research.microsoft.com/en-us/um/people/lamport/pubs/implementation.pdf
2. L. Lamport, R. Shostak, M. Pease, "The Byzantine Generals Problem", 1982 [Режим доступу] http://www.cs.cornell.edu/courses/cs614/2004sp/papers/lsp82.pdf
3. Орхан Гасімов. Мікросервісна архітектура. [Режим доступу] https://www.globallogic.com/ua/insights/blogs/microservices-architecture-for-beginners-part-one/
4. Experiences from Failing with Microservices. [Режим доступу] https://web.archive.org/web/20160303221756/http://www.infoq.com/news/2014/08/failing-microservices
5. Eric Evans. Domain-Driven Design: Tackling Complexity in the Heart of Software., 2003, 560 с.
6. Комплексні системи захисту інформації. Навчальний посібник [Електронний ресурс] / Ю. Є. Яремчук, П. В. Павловський, В. С. Катаєв, В. В. Сінюгін – Режим доступу: https://web.posibnyky.vntu.edu.ua/fmib/41yaremchuk_kompleksni_systemy_zahystu_informaciyi/rozdil7.html
7. НД ТЗІ 1.1-002-99. Загальні положення щодо захисту інформації в комп'ютерних системах від несанкціонованого доступу / Департамент спеціальних телекомунікаційних систем та захисту інформації Служби безпеки України – 1999