Microservice Architecture

Security Issues and Mitigation Approaches

Peter Harvey and Narissa Tsuboi

CPSC5200 Software Architecture

March 6th, 2023

Microservice Architecture

microservice

independent service

represents a domain

communicates via requests to API endpoints

microservice architecture (MA)

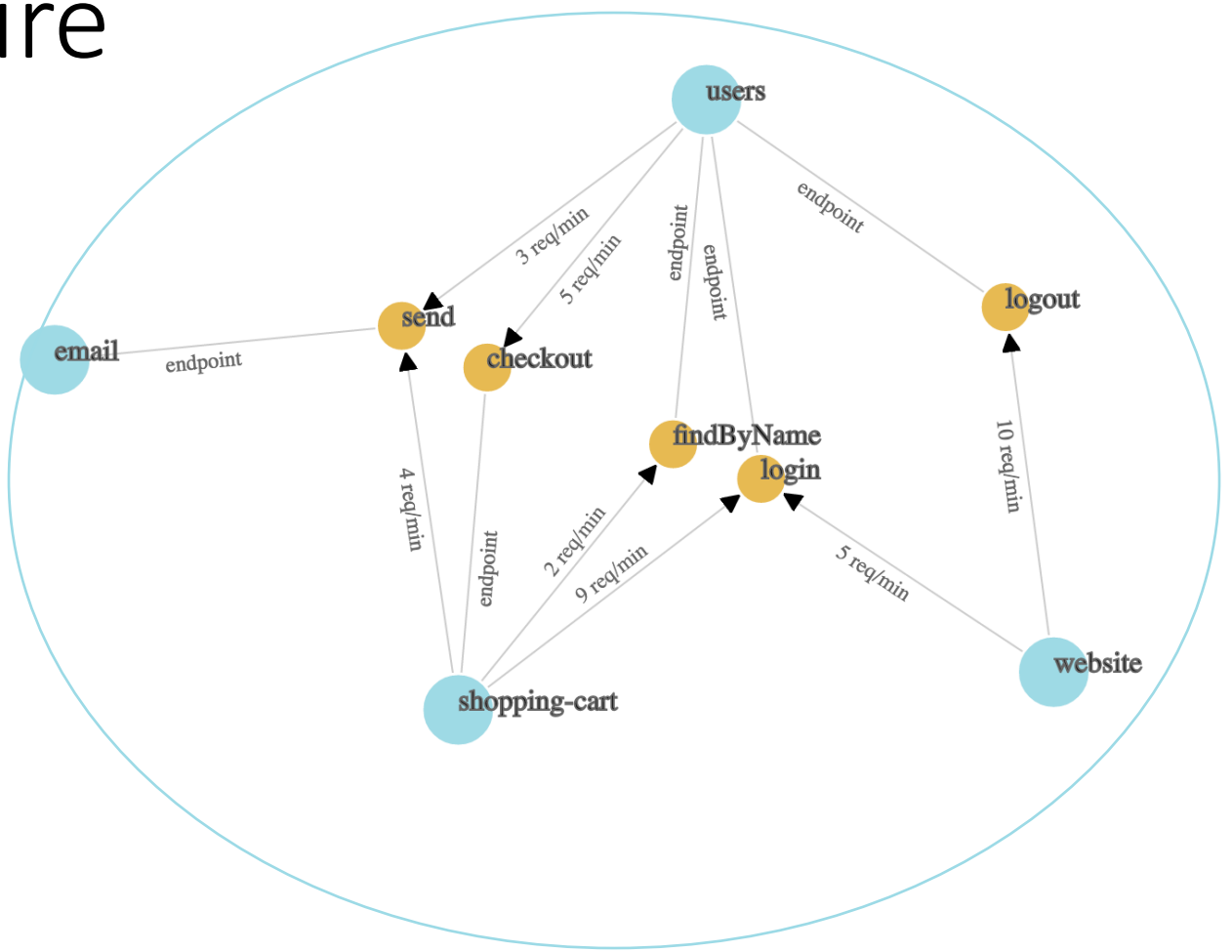
loosely coupled network of microservices

deployed in the cloud

technologically agnostic

changes and updates implemented rapidly

robust, fast, and scalable



Source: rlazoti microservice-dependency-graph repo
https://github.com/rlazoti/microservice-dependency-graph

Microservice Security

RQ1. ***What microservice security issues exist and what are their causes?***

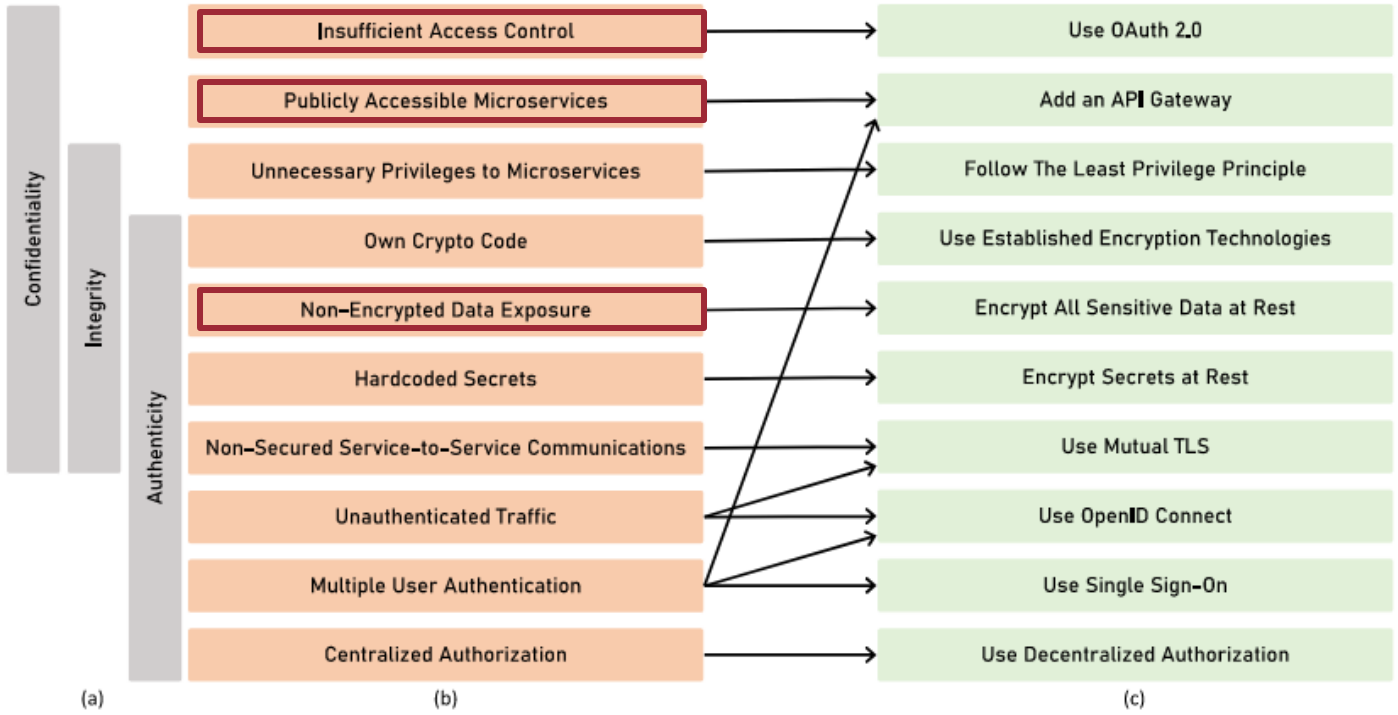
RQ2. ***What strategies exist to mitigate security issues?***

https://www.moesif.com/blog/technical/api-security/API-Security-Threats-Every-API-Team-Should-Know/

Smells and refactorings for Microservice Architecture security: A multivocal literature review [2]

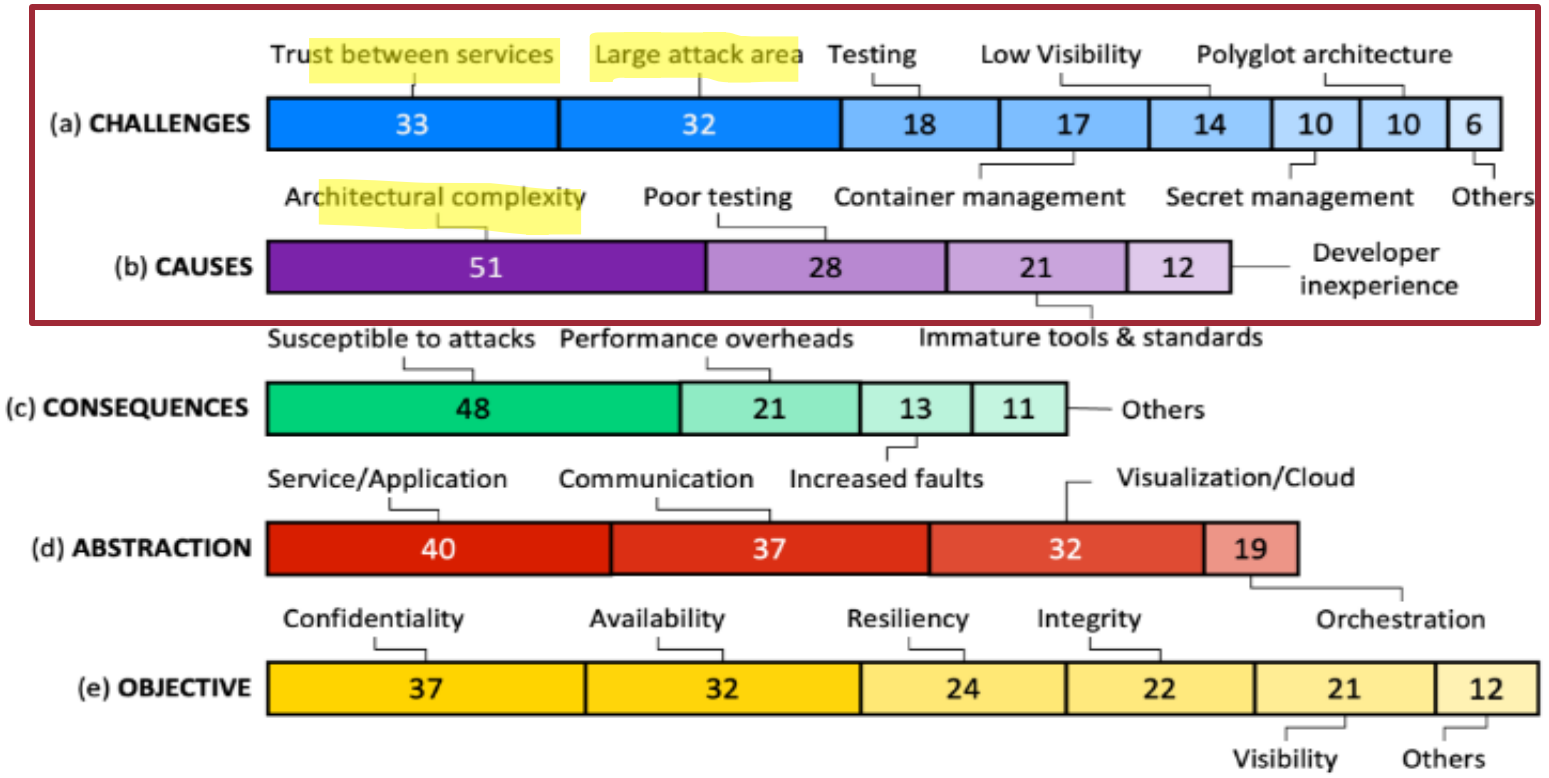
- "Distill well-known smells for securing microservices"
- White and grey SLR.
  - Kitchenham and Charters' guidelines.
  - 58 sources, 40 grey, 2015 to 2022.
  - No code repos.
- 10 security smells identified into RQ1 taxonomy associated with ISO security properties and common refactorings.
- Most frequent
  - Insufficient access control.
  - Unnecessary privileges.
  - Non-secured service to service comms.

Taxonomy of microservice security properties, smells, and refactorings (from left to right) [2].



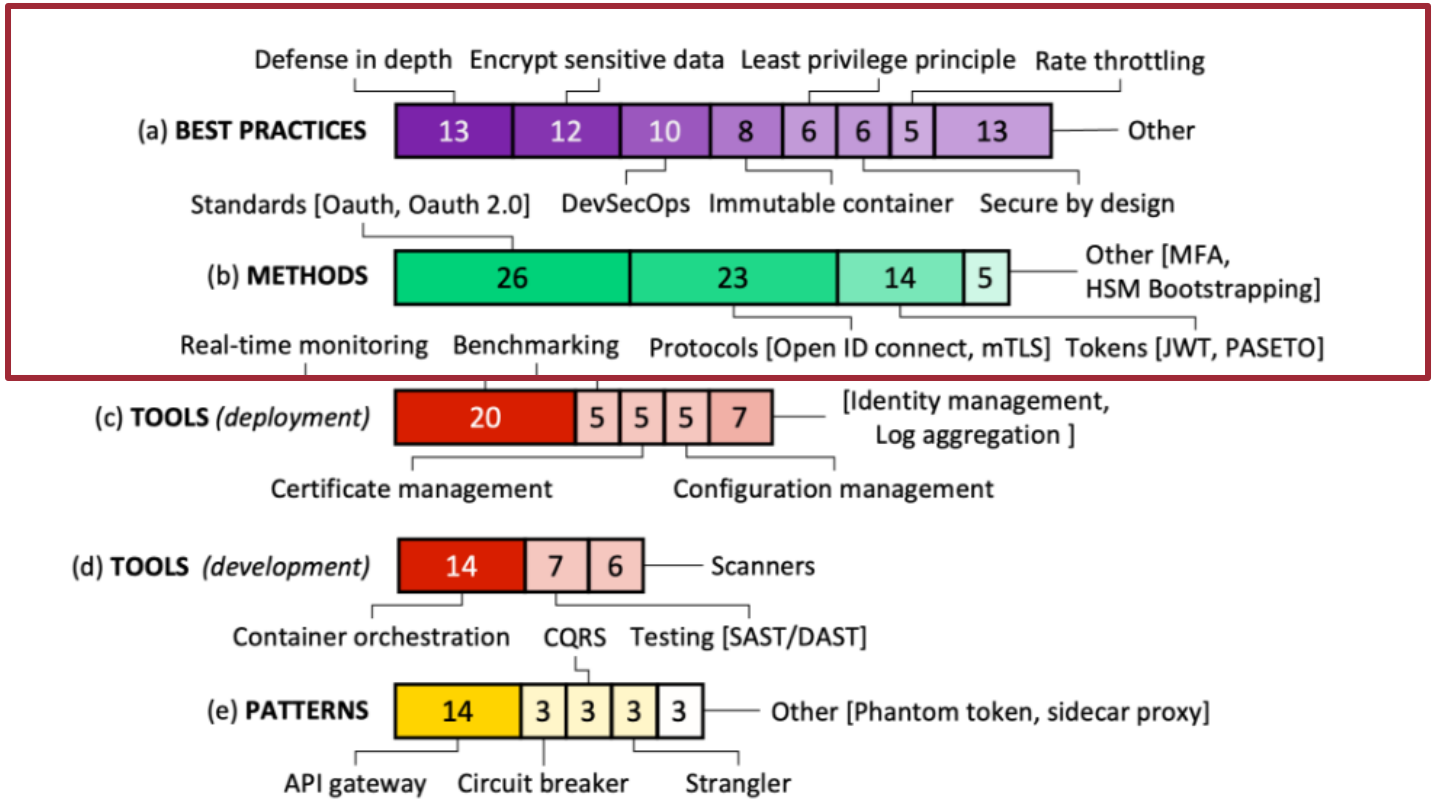
SoK: Security of Microservice Applications: A Practitioners' Perspective on Challenges and Best Practices [1]

- Identify challenges that arise when securing microservices.
- Aggregate solutions recommended by practitioners.
- Grey SLR.
  - Petersen et al guidelines.
  - 57 sources, 2015 to 2022.
  - Included code repos.



Microservice challenges and their associated causes, consequences, abstractions, and objectives identified in the SoK SLR [1].

SoK: Security of Microservice Applications: A Practitioners' Perspective on Challenges and Best Practices [1]



Microservice best practices and their associated methods, tools, and patterns identified in the SoK SLR [1].

An Empirical Study of Security Practices for Microservices Systems [3]

- Identified microservice security practices and had microservice practitioners evaluate their usefulness.
- Empirical software engineering
  - Analyzed GitHub issues and Stack Overflow posts, that include “design decisions, challenges, or solutions” in microservice systems for security practices.
  - Surveyed 74 microservice practitioners on the usefulness of these security practices.
- Identified 28 security practices. All were deemed useful by practitioners.
- Cataloged into 6 categories
  - Authorization and Authentication
  - Tokens and Credentials
  - Internal and External Microservices
  - Microservices Communications
  - Private Microservices
  - Databases and Environments

PA4	A large microservices system is recommended to use an API Gateway approach for securing/authorizing/routing microservices.	(GitHub member, 2021i; Stack Overflow member, 2021d,h; Github member, 2019a; Stack Overflow member, 2021q)
PI3	Whether microservices are only internally used within an organization or are externally accessible to third parties, authentication is required either way.	(Stack Overflow member, 2021f; GitHub member, 2020e,b)
PI4	In an internal microservice use case, “client credential” should not get exposed to the third party.	(Stack Overflow member, 2021n)
PI5	Microservices systems made of components should be isolated and internal calls should not be leaked outside their boundaries.	(GitHub member, 2021l)
PI6	Encrypt tokens if they are going to be exposed to the outside of the system boundary.	(Stack Overflow member, 2021c)

Top five most useful microservice security practices [3].

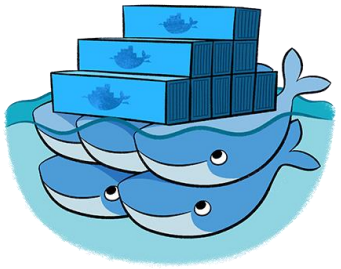
Overcoming Security Challenges in Microservice Architectures [4]

- Goals:
  - Provide a taxonomy of microservices security and assess the security implications of the microservice architecture
  - Survey related contemporary industry solutions and trends
  - Describe the design and implementation of a simple security framework for microservices that can be leveraged by practitioners

Overcoming Security Challenges in Microservice Architectures [4]

- Properties distilled from the literature review:
  - Do one thing and do it well
  - Automated, immutable deployment
  - Isolation through loose coupling
  - Diversity through system heterogeneity
  - Fail fast

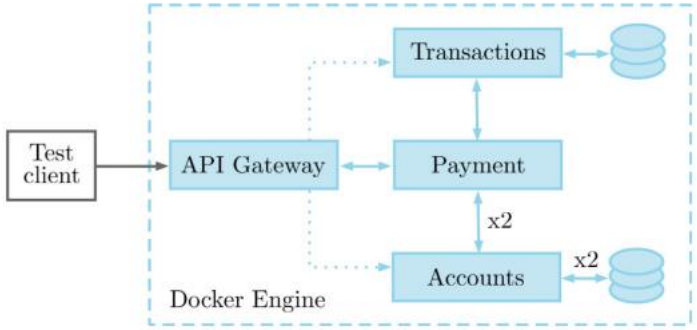
NETFLIX



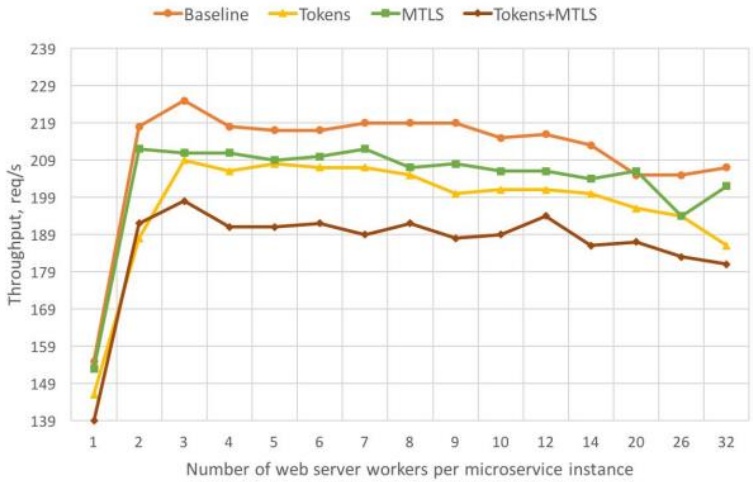
- Emerging Security Practices in Industry:
  - Mutual authentication using MTLS with a self-hosted Public Key Infrastructure (PKI)
    - Used by Docker Swarm and Netflix
  - Principal Propagation via Security Tokens
  - Fine-Grained Authorization

Overcoming Security Challenges in Microservice Architectures [4]

- Open-Source Microservice Security Framework (MiSSFire)
  - Establishes trust between individual microservices by using MTLS and principal propagation via JWT.
  - Experiment accomplished to test the performance of the framework. Results: loss in performance from using the proposed framework is minimal based on the fact that microservice solutions are slow in general.



Experiment setup: payment operation [4]



Performance of the bank model under load of 50 test clients making payments [4]

RQ1. ***What microservice security issues exist and what are their causes?***

Issues

Of the 46 security issues between the four papers, many overlap into broad categories.

- **Difficult to establish trust** (auth, identity) and enforce **data integrity** between services.
- **Large attack area** (an API for every service) for SQL injection attacks, DDOS attacks.
- **DevOps** monitoring a large, dynamic group of services and managing their containers is difficult.
- **Testing** typically not prioritized during development in rapid Agile environments, hard to detect where failures are occurring.

Causes

- Inherent architectural complexity.
- Weak standards employed during development, or none at all.
- Poor testing.
- Developer inexperience with microservice architecture.

RQ2. ***What strategies exist to mitigate security issues?***

Trust

- Use Oauth 2.0, 2 Factor Authentication, Single Sign-On.
- Secure communications between microservices with Mutual TLS security protocol.
- API Gateway.
- Follow the Least Privilege Principle.

Data Integrity

- Make containers immutable after deployment – keep data storage outside of containers.

DevOps to DevSecOps

- Build security into DevOps from the start.

Secure-by-design

- Security should not be an afterthought, but a consideration throughout the microservice development lifecycle!

Future Directions

Explore the benefits and drawbacks of a polyglot architecture – our core papers disagreed on this.

Perform surveys with microservice practitioners to gain more insight on the following:

- If any practices in industry are not captured in white/grey papers
- To find concrete examples of security breaches in microservice systems
- To understand how high of a priority security is when it comes to building microservice systems and what can be done to prioritize it

Search for other microservice security frameworks like MiSSFire and perform tests to compare their performances or effectiveness

References

[1] P. Billawa, A. Bambhore Tukaram, N. E. Díaz Ferreyra, J.-P. Steghöfer, R. Scandariato and G. Simhandl, "SoK: Security of microservice applications: A practitioners' perspective on challenges and best practices," in *Proceedings of the 17th International Conference on Availability, Reliability and Security*, Vienna, 2022.

[2] F. Ponce, J. Soldani, H. Astudillo and A. Brogi, "Smells and refactorings for microservices security: A multivocal literature review," *Journal of Systems and Software*, vol. 192, p. 111393, October 2022.

[3] A. Rezaei Nasab, M. Shahin, S. A. Hoseyni Raviz, P. Liang, A. Mashmool and V. Lenarduzzi, "An empirical study of security practices for microservices systems," *Journal of Systems and Software*, vol. 198, p. 111563, 2023.

[4] T. Yarygina and A. H. Bagge, "Overcoming security challenges in microservice architectures," in *2018 IEEE Symposium on Service-Oriented System Engineering (SOSE)*, Bamberg, 2018.

Q&A