Aalto University
School of Science
Bachelor's Programme in Science and Technology

Security in Microservice Architecture

- Impact of a Switch from Monolith to Microservices

Bachelor's Thesis

xx. xxxxxkuuta 2020

Tommi Jäske

Aalto-yliopisto
Perustieteiden korkeakoulu
Teknistieteellinen kandidaattiohjelma

KANDIDAATINTYÖN
TIIVISTELMÄ

Tekijä:	Tommi Jäske
Työn nimi:	Turvallisuus mikropalveluarkkitehtuurissa - Monoliitisesta arkkitehtuurista siirtyminen mikropalveluarkkitehtuuriin ja sen vaikutukset.
Päiväys:	xx. xxxxxkuuta 2020
Sivumäärä:	?
Pääaine:	Computer Science
Koodi:	SCI3027
Vastuuopettaja:	Professori Eero Hyvönen
Työn ohjaaja(t):	Professori Tuomas Aura (Tietotekniikan laitos)
Kirjoitetaan myöhemmin.	
Avainsanat:	avain, sanoja, niitäkin, tähän, vielä, useampi, vaikkei, niitä, niin, montaa, oikeasti, tarvitse
Kieli:	Suomi

Aalto University
School of Science
Bachelor's Programme in Science and Technology

ABSTRACT OF
BACHELOR'S THESIS

Author:	Tommi Jäske
Title of thesis:	Security in Microservice Architecture - Impact of a Switch from Monolith to Microservices
Date:	MonthName 31, 2020
Pages:	?
Major:	Computer Science
Code:	SCI3027
Supervisor:	Professor Eero Hyvönen
Instructor:	Professor Tuomas Aura (Department of Computer Science)
Will be written.	
Keywords:	key, words, the same as in FIN/SWE
Language:	English

Contents

1 Introduction

In recent years the mobile app has revolutionized our daily lives. These services have infiltrated our social life, shopping and almost every aspect of our existence. The services and their apps compete of our time and markets are reinvented constantly.

The rapid expansion and at times even faster decline of these web services need a matching architecture to meet these very specific needs.

In many cases the monolith services have started to use certain aspects from the microservice world, such as access tokens and REST API:s. Monoliths have served us well but the time has come to evolve with the customer needs.

There are many web services already in use which have been designed and implemented before the onslaugh of microservices. Some of these services need to evolve to be of use in the future. The pressure from new competitors starting immediatelly with new technologies and the fact that the industry and its developer base are extremely young dictates that the old and established services have to address the situation someway or another.

Stackoverflow annual survey (Stack Overflow) conducted on finds that half of the respondents identified as full-stack or backend developers. The professional developers had very little experience and about 40% of them had less than five years of professional experience.

The new developers entering the work force have very different mindset than the older more seasoned professionals. Thus, it is very clear that the ways of working and paradigms to be used are in constant change. The old and established have embrace the change and refactor their architecture before it is too late. Microservices are not the proper choice for all needs (Newman, 2019) but in many cases there simply is no other valid choice. This change needs to happen in an orderly and safe way and the security aspects need to be addressed.

Microservice Architecture (MSA) differs in many ways from the more tradition Monolith Architecture (MA). This shift entails very specific security issues.

In this thesis the MSA and security literature is evaluated and the main differences between MA and MSA security aspects are found.

Chapter 2. presents the definitions used in this thesis. Chapter 3. discusses the Confidentiality aspect of a switch from MA to MSA. In chapter 4. Integrity of the information is discussed in the context of MA and MSA. Chapter 5. presents Availability when changing from MA to MSA. In Chapter 6. other relevat security aspects are presented. Chapter 7. contains the conclusions and presents further research topics.

2 Definitions

This thesis uses the following definitions.

2.1 Microservice

Microservice architecture is in essence an extension of the service oriented architecture (SOA). It's guiding principles are stated in the SOA manifesto (Arsanjani et al., 2009) and one is to prioritize:

- *Business value over technical strategy*
- *Strategic goals over project-specific benefits*
- *Intrinsic interoperability over custom integration*
- *Shared services over specific-purpose implementations*
- *Flexibility over optimization*
- *Evolutionary refinement over pursuit of initial perfection*

A microservice is a service that: is independently deployable, is modeled around business domain, that owns the data that they need to operate, that communicates via network, is technology agnostic, that encapsulates data storage and retrieval and that has stable interface (Newman, 2019).

2.2 Security

Security can be defined in multiple ways but in this thesis security and more specifically information security is defined as consisting of Confidentiality, Integrity, and Availability (CIA) as is stated in the pocket book on ISO/IEC 27001 -standard for information security (Calder, 2008).

The ISO/IEC 27001 standard defines confidentiality as such that information or property is available to the authorized user only. Integrity means that the data or property is safeguarded for accuracy and completeness. Availability in this web service context is defined as such that the property or information is only available or diclosed to authorized users. The authorized users can consist of persons, processes or entities to whom the information or property can be disclosed.

3 Confidentiality

Confidentiality in a web service is usually critical security feature. Users should not be able to use or view content not authorized to him/her. In an MA access control can be implemented using sessions. A user authenticates using appropriate channels and a session with a session key is created. The session can have an expiration time and the messages originating from the user interface (UI) carry this key. Sessions and session keys can be used in a distributed system which MSA is but the implementation is more difficult (He and Yang, 2017).

3.1 Introduction

3.2 Effects on Confidentiality

3.3 Example case

4 Integrity

Information integrity in an MA web service is usually left to a single database and sound architectural choices (REALLY? SOURCE). Transactions can be used when updating database constents to make sure that atomicity, consistency, isolation, and durability (ACID) (Haerder and Reuter, 1983) is followed. When using MSA according to the definition each of the micro services should contain or have access to it's own data i.e. database. This leads to extreme difficulties in information integrity. TODO

4.1 Introduction

4.2 Effects on Integrity

4.3 Example case

5 Availability

5.1 Introduction

5.2 Effects on Availability

5.3 Example case

6 Other security matters

6.1 Platforms

Docker Swarm Kubernetes (K8s) Azure

sandbox virtualization

6.2 Software Development

6.3 Deployment

Developing software using the MA the structure the whole application or service is usually deployed as a whole and the program code can be compiled, tested and used as a single unit or multiple modules. In contrast to this a service implemented by using a MSA can be deployed in single microservice units and thus a single service can be worked upon individually.

7 Conclusion

References

A. Arsanjani, Grady Booch, Toufic Boubez, Paul C. Brown, David Chappell, John deVadoss, Thomas Erl, Nicolai Josuttis, Dirk Krafzig, Mark Little, Brian Loesgen, Anne Thomas Manes, Joe McKendrick, Steve Ross-Talbot, Stefan Tilkov, Clemens Utschig-Utschig and Herbjörn Wilhelmsen. SOA Manifesto, 2009. Available `http://www.soa-manifesto.org`. Viewed 2.2.2020.

A. Calder. *ISO27001 / ISO27002 A Pocket Guide*. IT Governance Publishing, Cambridgeshire, 2008. ISBN 978-1-84928-166-9.

T. Haerder and A. Reuter. Principles of transaction-oriented database recovery. *CM Computing Surveys (CSUR)*, 15(5):287–317, December 1983.

X. He and X. Yang. Authentication and authorization of end user in microservice architecture. *The 2017 International Conference on Cloud Technology and Communication Engineering (CTCE2017)*, 2017.

S. Newman. *Monolith to Microservices. Evolutionary patterns to transform your monolith.* O'Reilly Media, Inc., 2019. ISBN 9781492047841. 1st edition.

Stack Overflow. Stack Overflow Developer Survey Results 2019. Available `https://insights.stackoverflow.com/survey/2019`. Viewed 1.2.2020.