Microservices: An Architectural Approach to Software Development
(Technical Paper)

Cybersecurity Measures within Microservice Applications
(STS Paper)

A Thesis Prospectus
In STS 4500
Presented to
The Faculty of the
School of Engineering and Applied Science
University of Virginia
In Partial Fulfillment of the Requirements for the Degree
Bachelor of Science in Computer Science

By
Wen Ip

October 27, 2022

On my honor as a University student, I have neither given nor received unauthorized aid on this assignment as defined by the Honor Guidelines for Thesis-Related Assignments.

ADVISORS

Rosanne Vrugtman, Department of Computer Science

Bryn E. Seabrook, Department of Engineering and Society

Prospectus

Introduction

Architecture is the backbone of every software application and determines how the application is developed and maintained by engineers. Through architectural reviews, companies found value in incorporating architecture in the company's software development methods (Joseph Maranzano et al., 2005). One of the problems that these companies experience are that projects fail or end up needed significant rework past their original schedules, which is why it is important to research these systems and architectures.

With increases in features of an application, the appropriate software metrics and architecture must be determined (Milić & Makajić-Nikolić, 2022). Microservices is a new approach to how an application is structured and built. Microservice architecture is an emerging approach that revolves around the idea that self-contained components of functionality make up a larger system (Rory O' Connor et al, 2017). It allows for an application to be built upon a collection of individual services that can be arranged, organized, and maintained into multiple applications. The potential of implementing this architecture transforms the way companies utilize cloud technology, as well (Huseyin Ünlu et al., 2022). With the emerging popularity of cloud technology, microservice architecture has been paired with cloud computing technology as cloud services are transitioning into microservices for development and deployment (Yingying Wen et al., 2022). An example of microservices being implemented with cloud technology is that the cloud services used to help manage input/output data are produced by microservices within different data containers (Alfredo Barron et al., 2022). With the use of automation on cloud platforms, microservices can be deployed and scaled quickly. Each microservice serves as a business need or functionality, meaning an application can easily be altered by choosing to

include a microservice or not. Microservice architectures allows for technology to achieve scalability and high level of availability (Sayed Heidari et al., 2022). Applications with microservice architecture can handle a higher workload and deploy faster with this architecture. Implementing microservices also avoid duplications of objects, and reduced time to search and create them from their registers (Muhammad Jarwar et al., 2018).

While microservices are a new architectural method for applications, there are drawbacks and unknown vulnerabilities to microservices. Implementing microservices also increased management complexity and posed a challenge for the standardization of microservices across teams. Microservices governance has concerns for application stability and arose problems, such as load balancing, fault detection, and autoscaling (Lu Wang et al., 2022). Another concern for microservices are its unknown vulnerabilities that serve as a potential security threat to sensor networks (Fei Ying et al., 2022). Since microservices are being paired with cloud technology, security becomes a big concern and users' data are at risk with implementing microservices. Cybersecurity is not only a concern for the company developing the application, but also users that use the application want their data protected as well.

A more efficient strategy for managing and standardizing microservices should be explored to maximize the potential of microservices in applications. From this study, the drawbacks and advantages of microservices will be discussed and will be used to recommend potential implementation methods for microservices that will improve applications. Specifically, web applications, such as complex mobile and IoT applications, have been switching over from monolithic to microservices (Feiyan Guo et al., 2022). For my technical topic, my internship experience project is a web application that uses microservice architecture and will be discuss and explore microservice implementation within a company. With the emergence of

microservices, my STS topic will explore effective implementation strategies for microservices by researching the drawbacks and advantages to optimize applications and address security concerns for businesses and users.

Technical Topic

There are many software architectures for different applications and purposes, but microservices applications are most often compared with monolithic architectures. Monolithic architectures are software applications that are constrained to one single operating system, meaning that they cannot be ran independently (Nicola Dragoni et al., 2017). This type of architectural method has its constraints, such as being memory-intensive, computational-intensive and require certain niche requirements (Nicola Dragoni et al., 2017). A large problem with this type of architecture is that it leads to a lot of time being wasted on fixing defects, which can be very costly for the company. Another drawback of monoliths is that they are limited, and developers are locked into using the language and framework set out by the original application (Nicola Dragoni et al., 2017). Focusing on certain attributes and values of an application, a study found that an increase in features lead to a higher value in these attributes in microservices (Milić & Makajić-Nikolić, 2022). Microservices have been able to address these problems that monoliths face.

The proposed alternative to monolithic architecture is a microservices architecture paired with cloud technology. Microservices have great potential and have been implemented into applications for many big companies, such as Amazon, Uber, and Netflix (Preimesberger, 2016). This method means that the model is segmented and worked on by smaller teams and independently deployable (Preimesberger, 2016). Compared with monoliths, microservices solve

the problem of having slower deployments in an application. An example of microservices architecture being implemented is for smart living devices that make up a sensor network of microservices (Eric Hitimana et al., 2022). From this implementation of microservices, the sensor nodes were found to be easily updated and connected with huge amounts of user requests going through, which helped contribute to fire prevention situations (Eric Hitimana et al., 2022). Seen in these examples, applications that implement microservice architecture are easily scalable and perform with faster network systems.

Choosing an architecture is very important because it determines the structure of the application and how it will be developed, which affects the users' experience. Microservices are also able to be written in different languages and frameworks for what sees fits as they connect to other microservices (Nicola Dragoni et al., 2017). The flexibility of microservices allows engineers to scale up and down easily and satisfy business needs by customizing the microservice with the necessary language, frameworks, APIs and more.

Since microservices are a newly implemented approach in many companies, the vulnerabilities of microservices are not as well known. One major concern of an application is its security, and it can protect users' data from hackers. Privacy of data and security is an important concern for users as they trust companies with their sensitive information.

To explore the implementation of microservice architecture at a company, I will be discussing my internship work. Over the course of 10 weeks, I worked on an application for a large sized banking company. It was developed on a team of 5 software engineering interns with a scrum master and product owner. The team worked in 2-week sprint using scrum methodologies. The purpose of the application is to allow financial advisors to give feedback on

how single-page applications are meeting their business needs. This application uses microservice architecture and can be imported into different financial advisors' platforms as a button on the taskbar of the page.

The microservice was used with Spring Boot framework and layers. The database was built on MongoDB. The frontend side of the project used Angular, Typescript, HTML, and CSS. The backend side of the project used Java. Other IDEs' used to create the project were Visual Studio Code, Postman, Gradle, and IntelliJ.

STS Topic

In this time of technology, users are expecting more from applications and faster deployments. Users have an expectation from businesses to deliver software products that perform faster and robustly with more every update (Preimesberger, 2016). Microservices affect the way the company structure and the ability to develop these applications for users. Understanding infrastructure design can help with software development implementations, such as scaling up an application or managing large quantities of data (Star, 1999). It is important to look at microservices architecture in practice because it does affect users' experience and other concerns, such as security of users' data. These software products have political qualities and embedded systems attached to them that can affect people (Winner, 1980). Products that deal with user information must hold trust and uphold responsibility to protect users' data from being compromised.

Especially with growing tensions of cyber threats, the implementation of cybersecurity for microservice applications is an increasing need for users. With microservice computerization and easy flexibility, comes drawbacks in security. One study highlighted a concern for container

image security issues in microservices that has the potential for hackers to access all the microservice architecture if they gain access to a vulnerable container image (Fei Ying et al., 2022). Another security concern is with microservice architecture APIs that are used to exchange personal health data (Chatterjee & Prinz, 2022). The health industry has more interconnection-enabled medical grade devices that are integrated in a larger health security system that runs on microservice communication APIs (Chatterjee & Prinz, 2022). Implementing microservices allows for these devices to connect easily and be scalable as a system, but there is a worry of how scalable the security measures are in place to support the microservice architecture. By reviewing the security concerns in real-life microservice applications, the security of microservices must be deeply considered at as it affects users' sensitive data and their experience.

However, companies often overlook cybersecurity practices because it is too costly for them. If cybersecurity measures are not considered then, it can damage the trust with users and hurt the economy and society (Domingo-Ferrer & Blanco-Justicia, 2020). From this study, they found that the main values around cyber security, such as security, privacy, fairness and autonomy, conflict with one another. One example of these tensions is that security can harm privacy, in the sense that monitoring the state of an application can increase security can result in sacrificing privacy. These are tradeoffs in users' and engineers' values that should be considered when implementing a microservice architecture, especially in cloud environments.

Security and architecture are concepts that affect one another and important for an engineer to determine because they affect the relationship and trust between engineers and users. In an 18-month participant observation study at a medium sized company, it highlighted that "the technology is the machine's relations with its users" (Woolgar, 1990). The framework that

Woolgar uses in this study will be the main framework used to guide understanding the relationships between engineers and users regarding the implementation and practice of microservices. With implementation of a new technology with a product, it is important to understand the relationship between the machine and the users, specifically how it is used by the users and what values do they hold. As mentioned before with the tradeoff between security and privacy, engineers must understand their relationship with users to make decisions on what to value more in an application. We live in a world where social software is in all aspects of people's lives, which means our relationship with technology is affected by emotions, sociability, and new values we hold like security and performance (Roberto Pereira et al., 2013).

Research Question and Methods

The research question I will be looking at is, "How does the implementation of microservice architecture for an application affect its security and data of its users?"

I will analyze the outcome of the 10-week project from my internship and see how implementing a microservice application worked within an organization. I also want to look at other case studies of other companies implementing microservices and compare the effectiveness of different implementation strategies (Milić & Makajić-Nikolić, 2022). The challenges from developing as well as maintaining a microservice application will be discussed through a literature review. I also want to look at case studies of microservices and how security measures and data protection are impacted. This will help determine the how manageable microservices are when facing potential security threats and how they perform to satisfy stakeholder's needs, such as meeting business standards for the company to properly execute business tasks or protecting the user's personal data for customers that use the application. This will give context

to how engineers should develop and react to security threats for microservice applications and highlight the security concerns around implementing this architecture.

Conclusion

The issues and concerns with monolithic architectures can be addressed with microservice architecture as an alternative. However, this architectural style brings new concerns, such as security. Since microservices are new and not research about around security issues, users may be concerned about the implementation of microservices for applications that handle their data. With more representation of microservice architecture methods in research, companies can make better decisions on how to develop their application and consider the users' values and business needs. Companies will be able to see what implementation methods are effective for using microservices along with keeping security measures and concerns at the forefront.

References

Barron, A., Sanchez-Gallegos, D. D., Carrizales-Espinoza, D., Gonzalez-Compean, J. L., & Morales-Sandoval, M. (2022). On the Efficient Delivery and Storage of IoT Data in Edge–Fog–Cloud Environments. Sensors (14248220), 22(18), 7016–N.PAG. https://doi-org.proxy1.library.virginia.edu/10.3390/s22187016

Chatterjee, A., & Prinz, A. (2022). Applying Spring Security Framework with KeyCloak-Based OAuth2 to Protect Microservice Architecture APIs: A Case Study. *Sensors* (14248220), 22(5), N.PAG. https://doi-org.proxy1.library.virginia.edu/10.3390/s22051703

Domingo-Ferrer, J., & Blanco-Justicia, A. (2020). Ethical Value-Centric Cybersecurity: A Methodology Based on a Value Graph. *Science & Engineering Ethics*, 26(3), 1267–1285. https://doi.org/10.1007/s11948-019-00138-8

Dragoni N., Giallorenzo S., Lafuente A.L., Mazzara M., Montesi F., Mustafin R., Safina L. (2017). Microservices: Yesterday, today, and tomorrow. Present and Ulterior Software Engineering Mazzara M., Meyer B. ; Springer: Cham, Switzerland, 195-216

Guo, F., Tang, B., & Tang, M. (2022). Joint optimization of delay and cost for microservice composition in mobile edge computing. World Wide Web, 25(5), 2019–2047. https://doi-org.proxy1.library.virginia.edu/10.1007/s11280-022-01017-2

Heidari, S. M., & Paznikov, A. A. (2022). Multipurpose Cloud-Based Compiler Based on Microservice Architecture and Container Orchestration. Symmetry (20738994), 14(9), 1818–N.PAG. https://doi.org/10.3390/sym14091818

Hitimana, E., Bajpai, G., Musabe, R., Sibomana, L., & Jayavel, K. (2022). Containerized Architecture Performance Analysis for IoT Framework Based on Enhanced Fire Prevention Case Study: Rwanda. Sensors (14248220), 22(17), 6462. https://doi-org.proxy1.library.virginia.edu/10.3390/s22176462

Jarwar, M. A., Kibria, M. G., Ali, S., & Chong, I. (2018). Microservices in Web Objects Enabled IoT Environment for Enhancing Reusability. Sensors (14248220), 18(2), 352. https://doi.org/10.3390/s18020352

Maranzano, J. F., Rozsypal, S. A., Zimmerman, G. H., Warnken, G. W., Wirth, P. E., & Weiss, D. M. (2005). Architecture Reviews: Practice and Experience. IEEE Software, 22(2), 34–43. https://doi.org/10.1109/MS.2005.28

Milić, M., & Makajić-Nikolić, D. (2022). Development of a Quality-Based Model for Software Architecture Optimization: A Case Study of Monolith and Microservice Architectures. Symmetry (20738994), 14(9), 1824–N.PAG. https://doi.org/10.3390/sym14091824

O'Connor, R. V., Elger, P., & Clarke, P. M. (2017). Continuous software engineering-A microservices architecture perspective. Journal of Software: Evolution & Process, 29(11), n/a-N.PAG. https://doi.org/10.1002/smr.1866

Pereira, R., Baranauskas, M. C. C., & da Silva, S. R. P. (2013). Social Software and Educational Technology: Informal, Formal and Technical Values. Journal of Educational Technology & Society, 16(1), 4–14.

Preimesberger, C. (2016). How Enterprises Can Use Microservices to Their Advantage. EWeek, 1.

Star, S. L. (1999). The Ethnography of Infrastructure. American Behavioral Scientist, 43(3), 377. https://doi.org/10.1177/00027649921955326

Ünlü, H., Bilgin, B., & Demirörs, O. (2022). A survey on organizational choices for microservice-based software architectures. Turkish Journal of Electrical Engineering & Computer Sciences, 30(4), 1187–1203. https://doi.org/10.55730/1300-0632.3843

Wang, L., Jiang, Y. X., Wang, Z., Huo, Q. E., Dai, J., Xie, S. L., Li, R., Feng, M. T., Xu, Y. S., & Jiang, Z. P. (2022). The operation and maintenance governance of microservices architecture systems: A systematic literature review. Journal of Software: Evolution & Process, 1. https://doi-org.proxy1.library.virginia.edu/10.1002/smr.2433

Wen, Y., Cheng, G., Deng, S., & Yin, J. (2022). Characterizing and synthesizing the workflow structure of microservices in ByteDance Cloud. Journal of Software: Evolution & Process, 34(8), 1–18. https://doi-org.proxy1.library.virginia.edu/10.1002/smr.2467

Winner, L. (1980). Do artifacts have politics? Daedalus, 109(1), 121 – 36.

Woolgar, S. (1990). Configuring the user: the case of usability trials. Sociological Review, 38(1), 58–99. https://doi.org/10.1111/j.1467-954X.1990.tb03349.x

Ying, F., Zhao, S., & Deng, H. (2022). Microservice Security Framework for IoT by Mimic Defense Mechanism. Sensors (14248220), 22(6), 2418. https://doi.org/10.3390/s22062418