Security Mechanisms Used in Microservices-Based Systems: A Systematic Mapping

Anelis Pereira Vale*, Gastón Márquez*, Hernán Astudillo*, Eduardo B. Fernandez†
*Universidad Técnica Federico Santa María, Valparaíso, Chile
Email: anelis.pereira@sansano.usm.cl, gaston.marquez@sansano.usm.cl, hernan@inf.utfsm.cl
†Florida Atlantic University, Boca Raton, USA
Email: ed@cse.fau.edu

***Abstract*—Microservices is an architectural style that conceives systems as a modular, costumer, independent and scalable suite of services; it offers several advantages but its growing popularity has given rise to security challenges. Building secure systems is greatly helped by deploying existing security mechanisms, but current literature does not guide developers about which mechanisms are actually used by developers of microservices-based systems. This article describes the design and results of a systematic mapping study to identify the security mechanisms used in microservices-based systems described in the literature. The study yielded 321 articles, of which 26 are primary studies. Key findings are that (i) the studies mention 18 security mechanisms; (ii) the most mentioned security mechanisms are authentication, authorization and credentials; and (iii) almost 2/3 of security mechanisms focus on stopping or mitigating attacks, but none on recovering from them. Additionally, it emerges that experiments and case studies are the most used empirical strategies in microservices security research. The clear identification of most-used security solutions will facilitate the reuse of existing architectural knowledge to address security problems in microservices-based systems.**

***Index Terms*—secure software development, security mechanisms, microservices-based systems, systematic mapping**

I. Introduction

Microservices-based systems are an emerging service-oriented architecture, composed of by several small independent services [1],[2]. Each of these services, are executed in their own process and communicate with each other through lightweight mechanisms [3]. The concept of microservices arises from the industrial practice of dividing large monolithic applications into smaller services that work synergistically. These services are implemented independently through fully automated implementation tools [4].

Although this emerging architectural style brings several advantages to develop complex systems, also new challenges emerge, such as security. Recent industrial reports[1] reveal that companies, such as Netflix, received massive attacks on their microservices-based systems in the last years. These attacks cover from specific services to the whole architecture. For this reason, companies are constantly creating variants of standard security mechanisms with the aim of protecting themselves from such attacks, since the development of architectures of this type implies great challenges.

[1]https://www.whitehatsec.com/

To develop secure software, it is important to take into account the vulnerabilities that one faces in order to find the right solutions to mitigate them. Although the use of security mechanisms has been informed in the development of microservices-based systems, their visibility is not yet explicit enough. This lack of visibility makes developers have a vague view of what aspects are useful to develop secure microservices-based systems .

In this article, we perform a **systematic mapping study** to detect the presence of security mechanisms used in the development of microservices-based systems. We selected 26 primary studies out of a total of 321 articles, and applied a rigorous protocol to extract, classify, and organize them.

The main **contributions** of our study are (i) a state of the art of the presence of security mechanisms that are referenced in microservices research, and (ii) the identification of the most-used security solutions in the development of microservices-based systems.

The reminder of this article is structured as follows: Section II overviews related work; Section III describes the study design; Section IV presents the results and Section V discusses the main findings; Section VI addresses threats to validity; and Section VII summarizes and concludes.

II. Related work

Several studies have already gathered and organized previous proposals for secure development of microservice-based systems.

Yu et al. [5] surveyed work related to security risks for microservices-based fog applications, and argued that security issues arise in four system aspects: containers (used as deployment and operational environment for microservices), data (communicated among services), permissions (as guarantee of services security), and network security (the foundation for secure communication). Finally, they proposed security solutions regarding services communication.

Ahmadvand et al. [6] also proposed a taxonomy of services, this time along the system, defense, and attack perspectives.

While these studies give a wide view of security in microservice-based systems, from a service level or from a fog and network perspective, they do not allow developers to determine which specific security mechanisms have been

proposed in the literature to develop secure microservices-based systems.

III. Systematic mapping protocol

To conduct the study, we followed the guidelines proposed by Petersen et al. [7] for systematic mapping studies, complemented with the strategies presented by Kitchenham et al. [8] for systematic literature reviews.

A. Goal and research questions

The goal of our mapping review is to *identify and classify the evidence about the presence of security mechanisms in microservices-based systems*. To achieve this goal, we pose the following research questions (RQ's):

- **RQ1**: *Which **security mechanisms** have been reported in microservices-based systems research?* We consider as "security mechanism" the recognizable, recurrent standard software structures that many times (but not necessarily) result from the use of systematic design, e.g. security patterns [9].
  - **RQ1.1**: *In which **security categories** these mechanisms fall?* We use the "security categories" proposed by Fernandez et al. [10] to classify security tactics: attack detection, attack stop or mitigation, attack reaction, and attack recovery.
- **RQ2**: *Which **empirical strategies** have been used to validate research on security mechanisms?* We use the empirical strategies proposed by Wholin et al. [11]: surveys, case studies, experiments, and replications.
- **RQ3**: *Which **research strategies** are used in the research of security mechanisms for microservices-based systems?* We use the research strategies proposed by Wieringa et al. [12]: evaluation research, solution proposal, validation research, philosophical research, opinion paper, and personal experience paper.

B. Mapping review process

Figure 1 illustrates the main steps of the mapping review process executed in this study. In the following sections, we shall describe each step.

*1) **Study search**:* We reviewed 7 major electronic databases to extract primary studies. These databases are: IEEE Xplorer, ACM Digital Library, Wiley, SpringerLink, Web of Science, ScienceDirect, and Google Scholar. In order to explore these databases, we used the following search string to obtain studies: ("secure" OR "security") AND ("mechanism"*) AND ("microservi"* OR "micro servi"* OR "micro-servi"* OR "microservices architect"* OR "microservices design" OR "microservices structur"*). We conducted the mapping from November 2018 to March 2019.

*2) **Study selection**:* We defined the following inclusion and exclusion criteria aiming to collect primary studies.

- *Inclusion criteria*: (i) The study is related to security in microservices contexts or the use of security mechanisms in microservices-based systems; (ii) the study should focus on security concerns; (iii) the study should be written in English.
- *Exclusion criteria*: (i) Short studies (less than 3 pages); (ii) secondary or tertiary studies (such as literature reviews, surveys, and others); (iii) studies without full text available; (iv) studies structured as tutorial, editorials, and others.

*3) **Snowballing process**:* We applied the "snowballing" process [13] aiming not to miss relevant studies. This technique is an iterative process that reviews the references of each selected study. Finally, we combined the studies obtained in this process along with the studies chosen by the previous step.

*4) **Quality assessment**:* We established quality criteria in order to assess the primary studies' quality selected by the review. We evaluated each quality criterion as *strongly agree*, *agree*, *neither agree nor disagree*, *disagree*, and *strongly disagree*. After brainstorming sessions among the members of the research team, we defined the following quality criteria:

- **QC1**: Does the study describe the background in which the security mechanism is being used?
- **QC2**: Does the study describe the problem which the security mechanism is trying to solve?
- **QC3**: Does the study describe explicitly the solution which the security mechanism provide?

*5) **Data extraction and synthesis**:* In order to answer the RQ's, we extracted the studies' data using the following fields: (i) id; (ii) name; (iii) venue; (iv) year; (v) security mechanism reported; (vi) empirical strategy; (vii) research strategy; (viii) security strategy.

To synthesize the data, in RQ1, RQ1.1, RQ2, and RQ3 we used descriptive statistics, frequency analysis, and we tabulated the main results.

IV. Results

We obtained a total of 26 primary studies (see Table I) published between 2015 and 2018, labeled as M1 through M26. We observed an increment in the number of publications of around 50% per year (see Figure 2). Most publications (54%) appeared on conferences, followed by journals (31%) and symposia (15%); see Figure 3.

Table II describes the quality assessment results:

- QC1: few studies described the context where the security mechanism is used (e.g., only M3 described specifically the context).
- QC2: in most studies, the addressed security problems are described explicitly.
- QC3: most studies (65%) described explicitly the solution provided by the security mechanisms.

In those studies where two or more security mechanisms are reported (i.e. most of them, see Figure 7), we used brainstorming to converge on a decision about the quality assessment assignment for each quality criterion.

TABLE I
PRIMARY STUDIES SELECTED.

Papers	Title	Authors	Published in	Year	Cite
M1	Securing IoT Microservices with Certificates	M.-O. Pahl, L. Donini	IEEE/IFIP Network Operations and Management Symposium	2018	[14]
M2	Security-as-a-Service for Microservices-Based Cloud Applications	Y. Sun, T. Jaeger, S. Nanda	International Conference on Cloud Computing Technology and Science	2015	[15]
M3	Docker Container Security via Heuristics-Based Multilateral Security-Conceptual and Pragmatic Study	Manu A R,J. Kumar Patel, S. Akhtar,V K Agrawal, K N Bala	International Conference on Circuit, Power and Computing Technologies	2016	[16]
M4	Requirements Reconciliation for Scalable and Secure Microservice (De)composition	M. Ahmadvand, A. Ibrahim	IEEE International Requirements Engineering Conference Workshops	2016	[17]
M5	Security and Privacy for Cloud-Based Data Management in the Health Network Service Chain: A Microservice Approach	C. Esposito, A. Castiglione, C.-A. Tudorica, F. Pop	IEEE Communications Magazine	2017	[18]
M6	Overcoming Security Challenges in Microservice Architectures in Microservice Architectures	T. Yarygina, A. Helene	IEEE Symposium on Service-Oriented System Engineering	2018	[19]
M7	Claimsware: A Claims-based Middleware for Securing IoT Services	V. Merin, Q. H. Mahmoud	Annual Computer Software and Applications Conference	2017	[20]
M8	Access Control with Delegated Authorization Policy Evaluation for Data-Driven Microservice Workflows	D. Preuveneers, W. Joosen	Future Internet Journal	2017	[21]
M9	Defense-in-depth and Role Authentication for Microservice Systems	Kai Jander, Lars Braubach, Alexander Pokahr	Procedia Computer Science	2018	[22]
M10	Low-Level Exploitation Mitigation by Diverse Microservices	C. Otterstad, T. Yarygina	European Conference on Service-Oriented and Cloud Computing	2017	[23]
M11	Integrating Personalized and Accessible Itineraries in MaaS Ecosystems Through Microservices	A. Melis, S. Mirri, C. Prandi, M. Prandini, P. Salomoni, F. Callegati	Mobile Networks and Applications	2018	[24]
M12	Modelling, validating, and ranking of secure service compositions	A. D. Brucker, B. Zhou, F. Malmignati, Q. Shi, M. Merabti	Software: Practice and Experience	2017	[25]
M13	Secure Cloud Computing: Reference Architecture for Measuring Instrument under Legal Control	A. Oppermann, F. Grasso, F. Thiel, J.-P. Seifert	Security Privacy	2018	[26]
M14	Docker ecosystem – Vulnerability Analysis	A. Martin, S. Raponib, T. Combea, R. Di Pietrob	Computer communications	2018	[27]
M15	Secure Cloud Micro Services Using Intel SGX	S. Brenner, T. Hundt, G. Mazzeo, R. Kapitza	International Conference on Distributed Applications and Interoperable Systems	2017	[28]
M16	A Game of Microservices: Automated Intrusion Response	T. Yarygina, C. Otterstad	International Conference on Distributed Applications and Interoperable Systems	2018	[29]
M17	Securing Docker Containers from Denial of Service (DoS) Attacks	J. Chelladhurai, P. R. Chelliah, S. Alampalayam	IEEE International Conference on Services Computing	2016	[30]
M18	Leveraging Cloud Native Design Patterns for Security-as-a-Service Applications	K. Torkura, M.I.H Sukmana, F. Cheng, C. Meinel	IEEE International Conference on Smart Cloud	2017	[31]
M19	Security Services Using Blockchains: A State of the Art Survey	T. Salman, M. Zolanvari, A. Erbad, R. Jain, M. Samaka	IEEE Communications Surveys and Tutorials	2018	[32]
M20	A Microservices Architecture for Reactive and Proactive Fault Tolerance in IoT Systems	A. Power, G. Kotonya	IEEE International Symposium on A World of Wireless, Mobile and Multimedia Networks	2018	[33]
M21	Integrity Protection Against Insiders in Microservice-Based Infrastructures: From Threats to a Security Framework	M. Ahmadvand, A. Pretschner, K. Ball, D. Eyring	Software Technologies: Applications and Foundations	2018	[34]
M22	Integrating Continuous Security Assessments in Microservices and Cloud Native Applications	K. A. Torkura, M. I.H. Sukmana, C. Meinel	International Conference on Utility and Cloud Computing	2017	[35]
M23	A Cyber Risk Based Moving Target Defense Mechanism for Microservice Architectures	K. A. Torkura, M. I.H. Sukmana, A. V.D.M. Kayemz	IEEE International Conference on Parallel and Distributed Processing with Applications,	2018	[36]
M24	Supporting Micro-services Deployment in a Safer Way: a Static Analysis and Automated Rewriting Approach	B. Benni, S. Mosser, P. Collet, M. Riveill	Symposium on applied Computing	2018	[37]
M25	Authentication and Authorization orchestrator for microservice-based software architectures	A. Bánáti, E. Kail, K. Karóczkai, M. Kozlovszky	International Convention on Information and Communication Technology, Electronics and Microelectronics	2018	[38]
M26	Integrity Protection Against Insiders in Microservice-Based Infrastructures: From Threats to a Security Framework	M. Ahmadvand, A. Pretschner, K. Ball, D. Eyring	Software Technologies: Applications and Foundations	2018	[39]

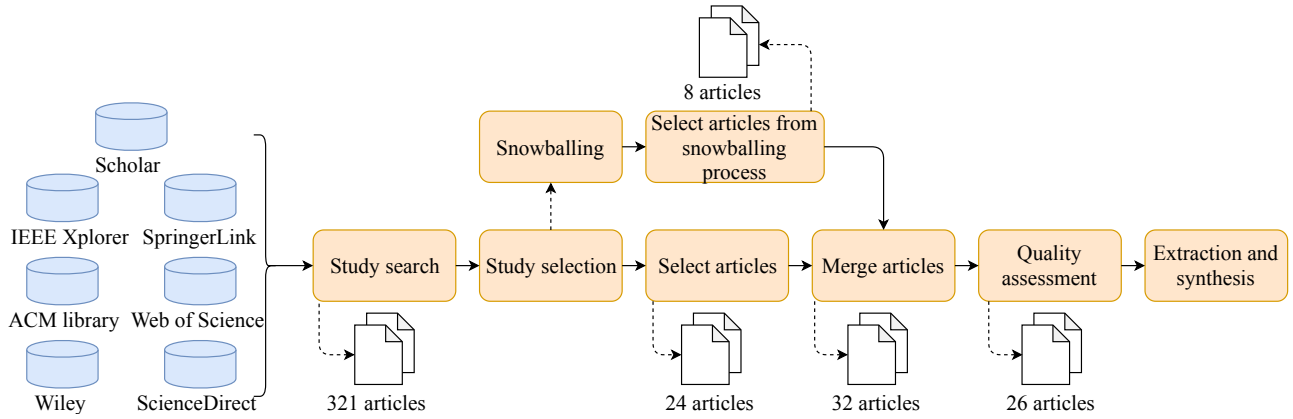

Fig. 1. Mapping review process.

TABLE II
QUALITY ASSESSMENTS RESULTS. LETTER "A" INDICATES *strongly disagree*; "B" INDICATES *disagree*; "C" INDICATES *neither agree nor disagree*; "D" INDICATES *agree*; "E" INDICATES *strongly agree*.

	QC1					QC2					QC3				
	A	*B*	*C*	*D*	*E*	*A*	*B*	*C*	*D*	*E*	*A*	*B*	*C*	*D*	*E*
M1				●					●						●
M2				●					●					●	
M3					●					●				●	
M4			●							●					●
M5			●						●					●	
M6			●					●							●
M7				●					●					●	
M8				●						●					●
M9			●							●					●
M10			●						●						●
M11			●						●						●
M12				●						●					●
M13			●							●			●		
M14			●							●			●		
M15				●					●						●
M16			●						●						●
M17				●						●				●	
M18				●						●					●
M19			●						●						●
M20				●					●					●	
M21				●					●					●	
M22			●							●					●
M23				●						●					●
M24			●							●					●
M25			●						●						●
M26				●						●					●

Summary

QC1	*A*	0%
	B	0%
	C	50%
	D	46%
	E	4%
QC2	*A*	0%
	B	0%
	C	46%
	D	4%
	E	50%
QC3	*A*	0%
	B	0%
	C	8%
	D	27%
	E	65%

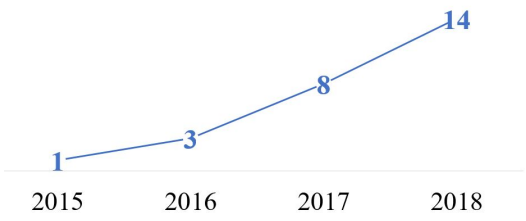

Fig. 2. Number of relevant publications per year, from 2015 to 2018.

A. RQ1: Security mechanisms in microservices-based systems studies

Figure 4 presents the security mechanisms reported in microservices-based systems studies. The most reported are **Authorization** (46% of studies), **Authentication** (42%) and **Credentials** (31%). These three mechanisms are closely related, and describe (i) who is authorized to access specific resources in a system and in what way, (ii) how to verify that the subject that intends to access the system is who it claims, and (iii) how to perform secure authentication and authorization registration, respectively. Besides, Credentials are an authorization and authentication mechanism.

The remaining reported security mechanisms appear in between 19% and 4% of studies.

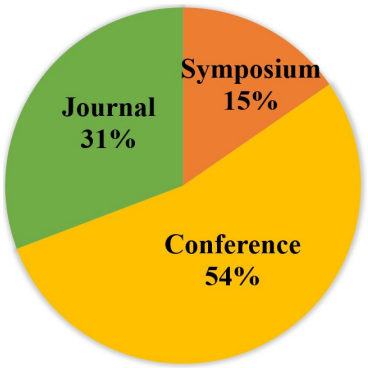

Fig. 3. Types of publication venues for the primary studies.

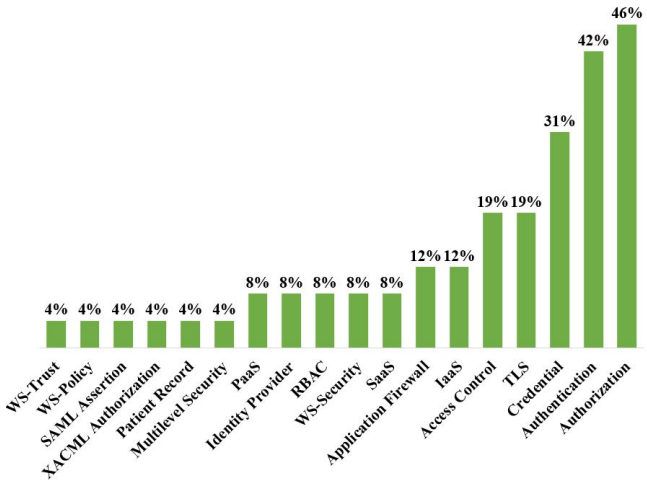

Fig. 4. Security mechanisms reported in the reviewed articles.

B. RQ1.1: Security strategies in microservices-based systems studies

Figure 5 illustrates the incidence of security strategies identified in the surveyed studies of microservices-based systems. Most mentioned security mechanisms aim to *stop or mitigate attacks* (58%); indeed, the Authentication, Authorization and Credentials security mechanisms have the largest presence among studies. Almost all other identified security mechanisms (39%) aim to *detect attacks*, and only the remaining 4% aim to *react to attacks*. We find out that none of the identified proposed security mechanisms aim to *recover from attacks*.

C. RQ2: Empirical strategies in research on microservices-based systems

The selected articles were classified along the empirical strategies of Wholin et al. [11], according to the type of validation. We found no reports about the use of *replications* for proposal validation. *Case studies* and *experiments* are used in equal measure, with approximately 42% each. *Surveys* are are also used but to a lesser extent (12%). Table III presents the validation approach for each article.

We were also surprised by the lack of replications to validate security mechanisms for microservices-based systems.

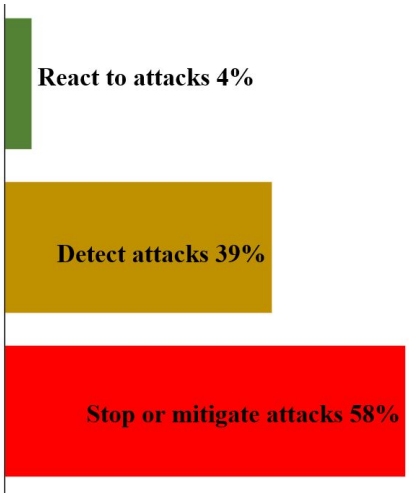

Fig. 5. Classification of articles according to Architectural security strategies in microservices-based systems.

One possible cause is that studies on secure development of microservices-based systems are quite recent (see Figure 2), and probably that there not yet techniques that can be reproduced.

TABLE III
CLASSIFICATION OF ARTICLES ACCORDING TO THE VALIDATION PROPOSED IN [11].

Papers	Experiment	Survey	Replication	Case Study
M1				
M2				✓
M3		✓		✓
M4				✓
M5	✓			
M6	✓			✓
M7	✓			✓
M8				
M9				
M10				
M11				
M12				✓
M13				
M14	✓			✓
M15	✓			
M16				
M17	✓			
M18	✓			
M19		✓		
M20	✓			
M21		✓		✓
M22	✓			✓
M23	✓			
M24	✓			✓
M25				
M26				✓
TOTAL	**42%**	**12%**	**0**	**42%**

D. RQ3: Research strategies for studies in microservices-based systems

For each study, we obtained its goals and problems and discussed them in brainstorming sessions. We found that the strategies used in primary studies are *evaluation research* (54%), *validation research* (38%), and *opinion document* (8%) (see Figure 6). Do recall that evaluation research studies establish clearly (1) the problem and its causal properties, and (2) the research method used and the validation of the proposed solution.

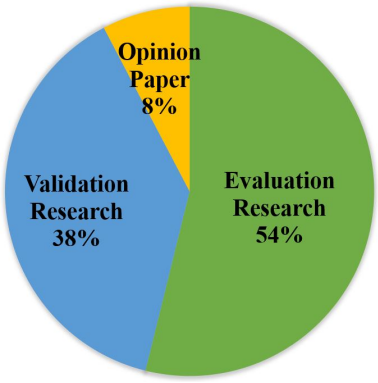

Fig. 6. Research Strategies used in the primary studies.

V. Discussion

This section discusses the main findings of this study.

A. Authorization, authentication and credentials in microservices research

Figure 7 presents the 18 security mechanisms most reported by the primary studies on microservices-based systems selected in our mapping review. The X axis reflects year of publication and the Y axis shows the reported security mechanisms (arranged alphabetically).

The most reported security mechanisms (see Figure 7) are Authorization, Authentication and Credentials. Industrial sources[2] [3] [4] indicate that the behavior of these mechanisms are related to how many microservices it is possible to secure, and mention several approaches to implement authorization and authentication in microservices-based systems; the main two are *Global Authentication and Authorization* (where authentication and authorization is carried out by front-end services), *Global Authentication and Service Authorization* (where authentication is performed in the global front-end services layer but authorization occurs in the back-end of each service). *Global Authentication and Service Authorization* are actually the most recommended approaches to secure microservices-based systems since they provides security mechanisms at the network level, specifically for microservice calls.

[2]InfoQ: https://www.infoq.com
[3]CodeBurst: https://codeburst.io
[4]DZone: https://dzone.com

He et al. [40] suggest that the most appropriate strategies for authentication and authorization in microservices architectures are (i) distributed session, (ii) SSO solutions, and (iii) JSON web token on the client side and JWT including API Gateway. To use authentication and authorization in microservices-based systems, they suggest to imitate in each microservice the same techniques used for monolithic architectures by the *Authenticator* and *Authorizer* security patterns [9]; thus, each service uses its own database or a shared database that stores credential data, but implements its own user verification.

Banati et al. [41] propose to separate the implementations of authentication and authorization in different modules. This strategy allows these security mechanisms to work even if the microservices-based systems changes.

To explore if the authentication, authorization, and credential mechanisms are actually used by developers of microservices-based systems, we built upon our previous study that maps which architectural patterns are actually used in which open source projects (described in [42] and summarized Table IV).

TABLE IV
Open source microservices-based systems described in [42].

Name	URL (https://github.com/)
Gizmo	nytimes/gizmo
Genie	Netflix/genie
Graph Processing	kbastani/spring-boot-graph-processing-example
Acme air	acmeair/acmeair-nodejs
Movie recommendation	mdeket/spring-cloud-movie-recommendation
Sock Shop	microservices-demo/microservices-demo
Microservices book	ewolff/microservice
Piggy Metrics	sqshq/PiggyMetrics
Netflix microservice	yidongnan/spring-cloud-netflix-example
microService	bishion/microService
Share bike	JoeCao/qbike
Lelylan	lelylan/lelylan
E-Commerce App	venkataravuri/e-commerce-microservices-sample
Task track support	yun19830206/CloudShop-MicroService-Architecture
CAS Microservice	ArcanjoQueiroz/cas-microservice-architecture
Warehouse microservice	HieJulia/warehouse-microservice
Microservices Reference	mspnp/microservices-reference-implementation
Vehicle tracking	mohamed-abdo/vehicle-tracking-microservices
EnterprisePlanner	gfawcett22/EnterprisePlanner
Micro company	idugalic/micro-company
Freddy's bbq joint	william-tran/freddys-bbq
Photo uploader	nginxinc/mra-ingenious
Delivery system	matt-slater/delivery-system
Service Commerce	antonio94js/servicecommerce
Blog post	fernandoabcampos/spring-netflix-oss-microservices
Tap-And-Eat	jferrater/Tap-And-Eat-MicroServices
WeText	daxnet/we-text
Pitstop	EdwinVW/pitstop
SiteWhere	sitewhere/sitewhere

This study focuses on identifying which *frameworks* implement which security mechanisms. In software development, the concept of frameworks is widely used to describe a pre-defined design to be used in the development and/or implementation of applications: "a collection of reusable software elements that provide generic functionality addressing recurring domain and quality attribute concerns across a broad range of applications." [43].

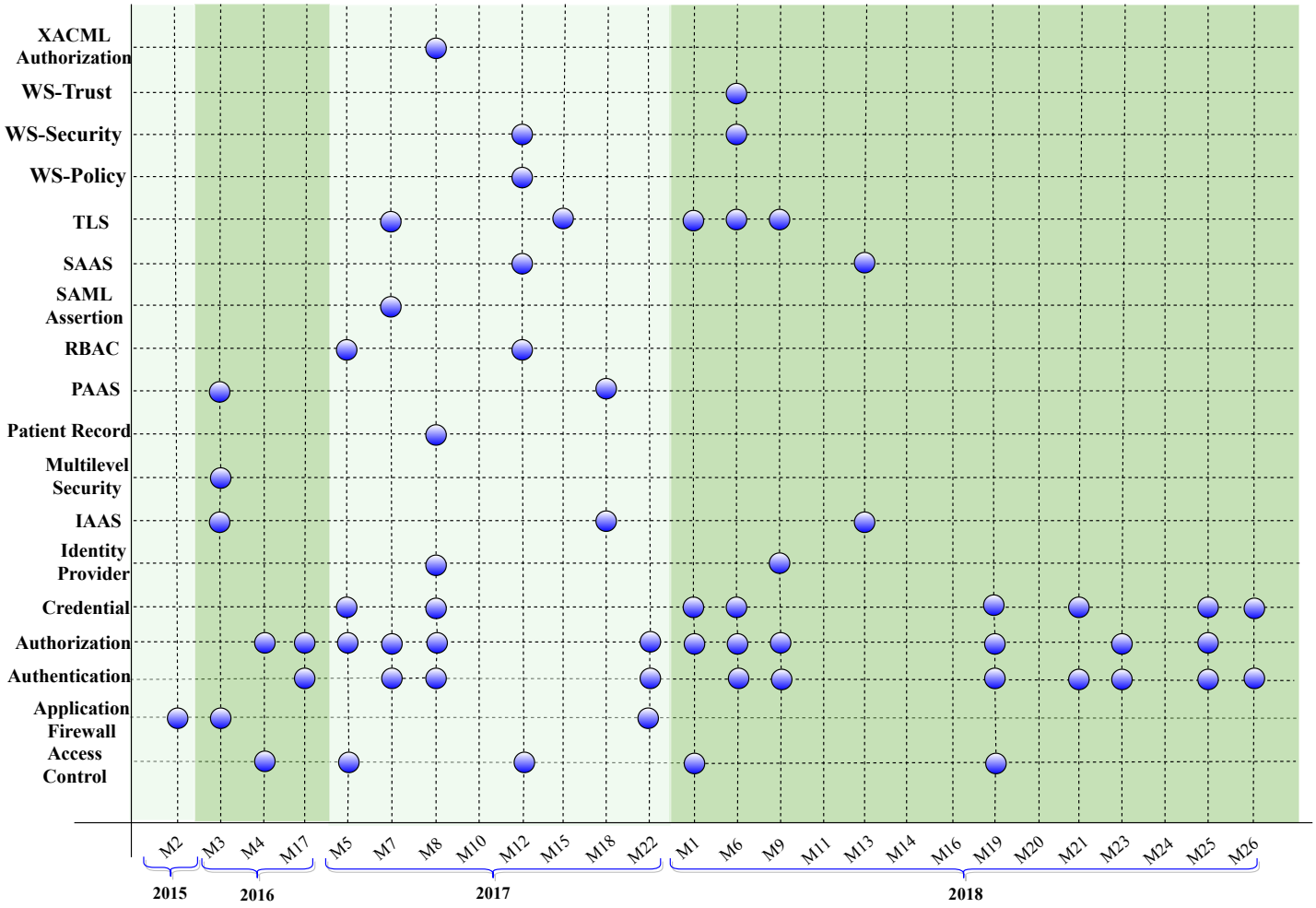

Fig. 7. Security mechanisms and their respective primary studies and publication years.

Our results show that 69% of projects use security frameworks, where the most referenced one is OAuth 2.0[5], an industry-standard protocol for authorization that is used in several domains. Although initially oriented to authorization, it is also used for authentication, client credentials, device codes, and other uses[6].

Therefore, our findings regarding the Authorization, Authentication, and Credentials mechanisms in the academic literature match, at first glance, the security mechanisms used in open source microservices-based systems.

B. Lack of mechanisms for "attack recovery" security strategy

The literature on secure microservices-based systems is clearly concerned first and foremost with stopping/mitigating attacks, and secondarily with detecting them, instead of reacting or recovering from them (see Figure 5) and Section V-A). Stop/mitigate attacks is related to the Authentication, Authorization and Credentials security mechanisms, which are the ones most referenced in the primary studies.

We explored academic sources (the same as section III-B) and industrial sources (InfoQ[7] and StackOverflow[8]) to explore the lack of attack recovery proposals. The key security concerns in microservices architecture we found are:

- *Code reuse*: The use of shared code and libraries can help migration to microservices, but it can also introduce lock-in problems and the need for propagation of patches to included components, possibly up to and including all the system's microservices.
- *Denial of service*: Managing a set of services with multiple entry points from the outside can be difficult. As the number of services grows, the magnitude of this problem is amplified. Proper management of security groups helps to ensure that only the correct ports are exposed.
- *Traffic between microservices*: Microservices exchange information. If traffic happens in a segregated part of the network, it can be assumed that the risk of having a spy is reduced since it is usually behind a corporate firewall, hence less susceptible to man-in-the-middle attacks.

[5]https://oauth.net/2/
[6]https://oauth.net/2/grant-types/
[7]https://www.infoq.com
[8]https://stackoverflow.com

However, when the microservices-based system is in an open cloud environment, this assumption is no longer valid, and besides adding microservices capabilities to handle encrypted traffic, which may affect the performance of the composing microservices.
- *Container implementation*: Usually microservices are implemented using containers. Containers do not have as strong isolation as virtual machines and may be attacked by software in other containers. Also the specific implementations of containers by different vendors have some vulnerabilities that may allow attacks to the application data or to the operating system

These concerns encompass security issues found in the review, for solutions that aim to stop, mitigate or detect attacks at different levels. As mentioned before, we did not identify security concerns associated with reacting to or recovering from attacks. Although Figure 5 shows that 4% of primary studies mention solutions intended to react to an attack, we did not find industrial evidence for it.

On a side note, this gap can be addressed with existing guidelines [10] for attack reaction and recovery:

- *React to attacks*: Alert subjects because ongoing attacks may require actions by operators, other personnel, or cooperating systems which belongs to a particular company. Also, apply any relevant institutional policies.
- *Recover from attacks*: Audit actions through records examination of user and system actions to trace actions of, and to identify, attackers. Also, apply any relevant institutional policies.

C. Increase in publications between 2017 and 2018

Figure 2 illustrates an increment in publications related to security mechanisms in 2017 and 2018. To figure out the reason of this increment, we collected the abstract of each primary study and used TagCrowd[9] to obtain a keywords cloud; see Figure 8.

Fig. 8. Keywords cloud of primary studies.

The most frequent keywords (omitting common words like "applications" and "microservices") "cloud", "data", "containers", "networks", and "attacks". According to the reports released by Nexusguard[10], the number of DDoS attacks recorded a year-on-year growth of 380%. Although DDoS attacks are less frequent in microservices-based systems[11], developers and practitioners are focusing on developing security solutions (mainly) to face this kind of attack.

VI. Threats to validity

In this section, we discuss the threats to the validity [11] of our study.

Threats to internal validity describe factors that could affect the results obtained from the study. To mitigate these threats, we created a strategy in which the search for related work was conducted on the defined keywords, and then applied a snowball process backwards in the selected studies. In addition, we excluded gray literature. Therefore, this potential bias had no significant impact for our study, since the selected articles passed a rigorous peer-review process, which is a solid requirement for high quality publications. Likewise, we apply inclusion and exclusion criteria, which were preliminary validated and correctly defined. We refined these criteria iteratively, based on the initial studies of the review.

Threats to external validity are conditions that limit our ability to generalize the results of our experiment to industrial practice. The potential threat related to external validity is related to whether the primary studies describe security mechanisms or not. We covered this threat by selecting peer-reviewed studies and excluding grey literature (white papers, editorials, and others). In the process of data analysis, we performed validity tests on the extracted information, through a process of cross analysis.

Threats to the conclusions validity are concerned with issues that affect the ability to draw the correct conclusions. In order to mitigate potential threats to the conclusions validity, we followed best practices proposed by Petersen et al. [7] and Wholin et al. [11]. Also, we used the security tactics taxonomy [10] [44] to identify security strategies.

VII. Conclusions

We conducted a systematic literature mapping in order to detect security mechanisms used in microservices-based systems, obtaining 26 primary studies. The results of this study reveal that (i) the high frequency of authentication, authorization, and credentials is not surprising because they are the most basic security mechanisms and any secure system must have them; (ii) detect attacks and stopping/mitigating attacks are the most frequent architectural security strategies used in microservices-based systems research; (iii) most security mechanisms are validated through case studies and experiments; (iv) the most used research strategy is evaluation research; and (v) we found no patterns for microservices-based systems security. This last topic is a good direction for future research; another interesting topic could be providing new mechanisms and techniques to react and recover from attacks in microservices-based systems.

[9]https://tagcrowd.com
[10]https://www.nexusguard.com/
[11]https://medium.com/netflix-techblog/starting-the-avalanche-640e69b14a06

ACKNOWLEDGMENTS

This work was supported by CONICYT (Chile) with grants PCHA/Doctorado Nacional/2016-21161005 and PIA (Basal FB0821 CCTVal), and by DPP scholarship, Universidad Técnica Federico Santa María (UTFSM).

REFERENCES

[1] A. Sill, "The design and architecture of microservices," *IEEE Cloud Computing*, vol. 3, pp. 76–80, Sep. 2016.

[2] M. Dong, K. Ota, and A. Liu, "Preserving source-location privacy through redundant fog loop for wireless sensor networks," in *2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing*, pp. 1835–1842, Oct 2015.

[3] D. Yu, Y. Jin, Y. Zhang, and X. Zheng, "A survey on security issues in services communication of microservices-enabled fog applications," *Concurrency Computation Practice and Experience*, 2 2018.

[4] N. Pathania, *Setting Up Jenkins on Docker and Cloud*, pp. 115–143. Berkeley, CA: Apress, 2017.

[5] D. Yu, Y. Jin, Y. Zhang, and X. Zheng, "A survey on security issues in services communication of microservices-enabled fog applications," *Concurrency and Computation: Practice and Experience*, pp. 1–19, 2018.

[6] M. Ahmadvand, A. Pretschner, and F. Kelbert, "A taxonomy of software integrity protection techniques," *Advances in Computers*, vol. 112, pp. 413–486, 2019.

[7] K. Petersen, R. Feldt, S. Mujtaba, and M. Mattsson, "Systematic mapping studies in software engineering," *International Conference on Evaluation and Assessment in Software Engineering (EASE)*, vol. 8, no. 68-77, 2008.

[8] B. Kitchenham and S. Charters, "Guidelines for performing systematic literature reviews in software engineering," *Keele University and Durham University Joint ReportTech.rep.ebse 2007-001*, 2007.

[9] E. B. Fernandez, *Security patterns in practice: designing secure architectures using software patterns*. John Wiley and Sons, 2013.

[10] E. B. Fernandez, H. Astudillo, and G. Pedraza-García, "Revisting architectural tactics for security," *European Conference on Software Architecture*, pp. 55–69, 2015.

[11] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén, "Experimentation in software engineering," *Springer Science Business Media*, 2012.

[12] R. Wieringa, N. Maiden, N. Mead, and C. Rolland, "Requirements engineering paper classification and evaluation criteria: a proposal and a discussion," *Requirements Engineering*, vol. 11, no. 1, pp. 102–107, 2006.

[13] C. Wohlin, "Guidelines for snowballing in systematic literature studies and a replication in software engineering," *Proceedings of the 18th international conference on evaluation and assessment in software engineering*, p. 38, 2014.

[14] M. Pahl and L. Donini, "Securing iot microservices with certificates," in *NOMS 2018 - 2018 IEEE/IFIP Network Operations and Management Symposium*, April 2018.

[15] Y. Sun, S. Nanda, and T. Jaeger, "Security-as-a-service for microservices-based cloud applications," in *2015 IEEE 7th International Conference on Cloud Computing Technology and Science (CloudCom)*, pp. 50–57, Nov 2015.

[16] A. R. Manu, J. K. Patel, S. Akhtar, V. K. Agrawal, and K. N. B. S. Murthy, "Docker container security via heuristics-based multilateral security-conceptual and pragmatic study," in *2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT)*, March 2016.

[17] M. Ahmadvand and A. Ibrahim, "Requirements reconciliation for scalable and secure microservice (de)composition," in *2016 IEEE 24th International Requirements Engineering Conference Workshops (REW)*, pp. 68–73, Sep. 2016.

[18] C. Esposito, A. Castiglione, C. Tudorica, and F. Pop, "Security and privacy for cloud-based data management in the health network service chain: a microservice approach," *IEEE Communications Magazine*, vol. 55, pp. 102–108, Sep. 2017.

[19] T. Yarygina and A. H. Bagge, "Overcoming security challenges in microservice architectures," in *2018 IEEE Symposium on Service-Oriented System Engineering (SOSE)*, pp. 11–20, March 2018.

[20] V. M. George and Q. H. Mahmoud, "Claimsware: A claims-based middleware for securing iot services," in *2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC)*, vol. 1, pp. 649–654, July 2017.

[21] D. Preuveneers and W. Joosen, "Access control with delegated authorization policy evaluation for data-driven microservice workflows," *Future Internet*, vol. 9, no. 4, p. 58, 2017.

[22] K. Jander, L. Braubach, and A. Pokahr, "Defense-in-depth and role authentication for microservice systems," *Procedia Computer Science*, vol. 130, pp. 456 – 463, 2018.

[23] C. Otterstad and T. Yarygina, "Low-level exploitation mitigation by diverse microservices," in *Service-Oriented and Cloud Computing* (F. De Paoli, S. Schulte, and E. Broch Johnsen, eds.), (Cham), pp. 49–56, Springer International Publishing, 2017.

[24] A. Melis, S. Mirri, C. Prandi, M. Prandini, P. Salomoni, and F. Callegati, "Integrating personalized and accessible itineraries in maas ecosystems through microservices," *Mobile Networks and Applications*, vol. 23, pp. 167–176, Feb 2018.

[25] A. D. Brucker, B. Zhou, F. Malmignati, Q. Shi, and M. Merabti, "Modelling, validating, and ranking of secure service compositions," *Softw. Pract. Exper.*, vol. 47, pp. 1923–1943, Dec. 2017.

[26] A. Oppermann, F. G. Toro, F. Thiel, and J.-P. Seifert, "Secure cloud computing: Reference architecture for measuring instrument under legal control," *Security and Privacy*, vol. 1, no. 3, p. e18, 2018.

[27] A. Martin, S. Raponi, T. Combe, and R. D. Pietro, "Docker ecosystem – vulnerability analysis," *Computer Communications*, vol. 122, pp. 30 – 43, 2018.

[28] S. Brenner, T. Hundt, G. Mazzeo, and R. Kapitza, "Secure cloud micro services using intel sgx," in *Distributed Applications and Interoperable Systems* (L. Y. Chen and H. P. Reiser, eds.), (Cham), pp. 177–191, Springer International Publishing, 2017.

[29] T. Yarygina and C. Otterstad, "A game of microservices: Automated intrusion response," in *Distributed Applications and Interoperable Systems* (S. Bonomi and E. Rivière, eds.), pp. 169–177, Springer International Publishing, 2018.

[30] J. Chelladhurai, P. R. Chelliah, and S. A. Kumar, "Securing docker containers from denial of service (dos) attacks," in *2016 IEEE International Conference on Services Computing (SCC)*, pp. 856–859, 2016.

[31] K. A. Torkura, M. I. H. Sukmana, F. Cheng, and C. Meinel, "Leveraging cloud native design patterns for security-as-a-service applications," in *2017 IEEE International Conference on Smart Cloud (SmartCloud)*, pp. 90–97, Nov 2017.

[32] T. Salman, M. Zolanvari, A. Erbad, R. Jain, and M. Samaka, "Security services using blockchains: A state of the art survey," *IEEE Communications Surveys Tutorials*, vol. 21, pp. 858–880, Firstquarter 2019.

[33] A. Power and G. Kotonya, "A microservices architecture for reactive and proactive fault tolerance in iot systems," in *2018 IEEE 19th International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM)*, pp. 588–599, June 2018.

[34] M. Ahmadvand, A. Pretschner, K. Ball, and D. Eyring, "Integrity protection against insiders in microservice-based infrastructures: From threats to a security framework," in *Software Technologies: Applications and Foundations* (M. Mazzara, I. Ober, and G. Salaün, eds.), pp. 573–588, Springer International Publishing, 2018.

[35] K. A. Torkura, M. I. Sukmana, and C. Meinel, "Integrating continuous security assessments in microservices and cloud native applications," in *Proceedings of the10th International Conference on Utility and Cloud Computing*, UCC '17, pp. 171–180, ACM, 2017.

[36] K. A. Torkura, M. I. H. Sukmana, and A. V. D. M. Kayem, "A cyber risk based moving target defense mechanism for microservice architectures," in *2018 IEEE Intl Conf on Parallel Distributed Processing with Applications, Ubiquitous Computing Communications, Big Data Cloud Computing, Social Computing Networking, Sustainable Computing Communications (ISPA/IUCC/BDCloud/SocialCom/SustainCom)*, pp. 932–939, Dec 2018.

[37] B. Benni, S. Mosser, P. Collet, and M. Riveill, "Supporting microservices deployment in a safer way: A static analysis and automated rewriting approach," in *Proceedings of the 33rd Annual ACM Symposium on Applied Computing*, SAC '18, pp. 1706–1715, ACM, 2018.

[38] A. Bánáti, E. Kail, K. Karóczkai, and M. Kozlovszky, "Authentication and authorization orchestrator for microservice-based software archi-

tectures," in *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pp. 1180–1184, May 2018.

[39] M. Ahmadvand, A. Pretschner, K. Ball, and D. Eyring, "Integrity protection against insiders in microservice-based infrastructures: From threats to a security framework," in *Software Technologies: Applications and Foundations* (M. Mazzara, I. Ober, and G. Salaün, eds.), (Cham), pp. 573–588, Springer International Publishing, 2018.

[40] X. He and X. Yang, "Authentication and authorization of end user in microservice architecture," *Journal of Physics: Conference Series*, vol. 910, p. 012060, oct 2017.

[41] A. Bánáti, E. Kail, K. Karóczkai, and M. Kozlovszky, "Authentication and authorization orchestrator for microservice-based software architectures," in *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pp. 1180–1184, May 2018.

[42] G. Márquez and H. Astudillo, "Actual use of architecture patterns in microservices-based open source projects," *25th Asia-Pacific Software Engineering Conference (APSEC)*, pp. 1–10, 2018.

[43] R. Kazman, "Rapid software composition by assessing untrusted components,," https://insights.sei.cmu.edu/sei_blog/2018/11/rapid-software-composition-by-assessing-untrusted-components.html.

[44] L. Bass, P. Clements, and R. Kazman, *Software Architecture in Practice (3rd Edition)*. SEI Series in Software Engineering, 2013.