Improving the Security of Microservice Systems by Detecting and Tolerating Intrusions

José Flora
University of Coimbra, CISUC, DEI
Coimbra, Portugal
jeflora@dei.uc.pt

***Abstract*—Microservice architectures adoption is growing expeditiously in market size and adoption, including in business-critical systems. This is due to agility in development and deployment further increased by containers and their characteristics. Ensuring security is still a major concern due to challenges faced such as resource separation and isolation, as improper access to one service might compromise complete systems. This doctoral work intends to advance the security of microservice systems through research and improvement of methodologies for detection, tolerance and mitigation of security intrusions, while overcoming challenges related to multi-tenancy, heterogeneity, dynamicity of systems and environments. Our preliminary research shows that host-based IDSes are applicable in container environments. This will be extended to dynamic scenarios, serving as a steppingstone to research intrusion tolerance techniques suited to these environments. These methodologies will be demonstrated in realistic microservice systems: complex, dynamic, scalable and elastic.**

***Index Terms*—Security, Intrusion Detection, Intrusion Tolerance, Microservices, Containers**

I. Introduction

The utilisation of microservice applications to support businesses is growing [1], [2]. This technology allows a faster and easier management and deployment of services which assure several important properties, such as scalability and elasticity. As the adoption of cloud is higher these characteristics are very appealing, allowing developers to use resources as needed and accommodate the dynamics in an on-demand fashion.

There are several concerns stated in multiple reports that are portrayed as obstacles higher microservices adoption. Security concerns inherited by this technology from other areas, such as distributed systems and service-oriented architectures (SOA) in general, slow down the adoption, especially in business-critical systems. These concerns are mainly focused in microservices security and security mechanisms, that are, at the moment, being researched and developed. The fundamental design of microservices consists in segmenting applications into small, independent, and coherent services that perform very well defined tasks and are decoupled from other services, cooperating in order to achieve a common goal [1].

Microservices deployment leverages the small footprint and lightweight of containers, which are a virtualisation technology built upon kernel features [3]. Containers are very easy to create and destroy; making it very applicable to this scenario. However, it also comes with some drawbacks, particularly in terms of security. The fact that every container shares the same physical resources may result in data separation and isolation breaches. This is exacerbated in cloud environments where multi-tenancy is the *de facto* and multiple vendors allocate resources to support their infrastructure [4]. As a result, these concerns need to be carefully researched.

Therefore, this work aims at contributing to detect, tolerate and mitigate such risks. For this, we are focusing our efforts in researching and devising approaches for detecting intrusions in microservices. These techniques will be focused on overcoming their prevailing characteristics and their common mechanisms. Intrusion detection will act as steppingstone for the application of intrusion tolerance measures, contributing to their improved robustness and resiliency to potential attackers.

The main goal of this work is to advance the security of microservices through research and improvement of three action areas: detection, tolerance and mitigation of security intrusions. This can be detailed in the following sub-objectives:

- **Research new effective and efficient intrusion detection measures**, overcoming challenges posed by scalability, elasticity and dynamicity characteristics in heterogeneous and complex microservices. This work will be used as steppingstone for the following aims.
- **Research and propose a generic approach for the design of intrusion tolerant microservices**, to empower the capacity of providing unimpaired service during the occurrence of security intrusions and to cope with microservices features (e.g., scalability, dynamic architecture). The proposed solutions will also be used to protect IDSes to increase their trustworthiness and resilience.
- **Evaluate and compare intrusion mitigation measures**, which reduce the impact of security intrusions detected without necessarily maintaining service availability, based on different scenarios defined according to microservice utilisation, such as business-critical systems.
- **Research and develop an effective integrated approach for intrusion detection and tolerance for microservices**. This will use the work produced during the previous objectives and intends to incorporate the complete contributions to produce a complete approach which applies a defence-in-depth methodology through the identification of compromised services (intrusion detection) and the application of measures that allow to provide the expected service even though breaches take place (intrusion tolerance).

978-1-7281-7735-9/20/$31.00 ©2020 IEEE
DOI 10.1109/ISSREW51248.2020.00051

II. Improving Microservice Applications Security

The work for this PhD aims has two main areas of action (see Fig. 1): the detection and the toleration of security intrusions against microservice applications. Initially, the main focus resided in studying the applicability of state-of-the-art algorithms in the context of container-based systems. Further, the work moved towards defining an approach for evaluation and comparison of such algorithms and techniques. Currently, we are focusing on the application of our work in the broader and more complex context of microservice applications, where the majority of the work will be conducted.

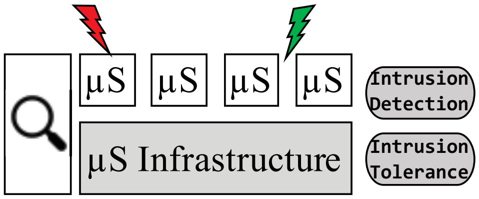

Fig. 1. Overview of the PhD work actuation scope.

The following sections provide further details with regard to the work plan and how we intend to contribute with research work to improve the state of the art of microservices security.

A. Research and Improve Intrusion Detection Systems for Microservices

As mentioned, the majority of the work developed until now has focused on intrusion detection. Our results demonstrated the applicability of behaviour-based intrusion detection in container-based environments [4]. However, there is a need for further work to tackle challenges posed by microservices characteristics such as scalability, elasticity, dynamicity and heterogeneity of the environment besides dealing with continuous integration and continuous delivery which exacerbate the existing concerns.

In order to address scalability challenges, we intend to propose methodologies and approaches to deal with variable number of service replicas and improve IDSes efficiency and detection rate through several methods, such as data aggregation, aiming to decrease the required computation resources to identify the behaviour deviations in microservices complex contexts.

As fast delivery and continuous modification of available services increases, it requires service behaviour models to be retrained. However, it is not feasible to compute a new profile each time an update or new feature is released since, with current methodologies, the environment can change by the minute. Therefore, researching ways to reutilise older profiles, or to update them in an efficient and incremental manner is crucial and may lead to cost-reduction and shorter time-to-secure interval.

Moreover, to take action against an intrusion it is necessary to pinpoint compromised services, even when concerns arise in different points of the architecture, that is, localising the entry-point of the security breach. Machine learning algorithms will be used to establish connections and event propagation across services and perform root-cause analysis through information combination and correlation.

B. Research and Propose Intrusion Tolerance for Microservices and Host-based IDSes for Microservices

Intrusion tolerance measures provide the capacity of service delivery despite intrusions occurring. Besides, these measures can be proactively applied to microservices and we also aim at strengthening intrusion detection systems.

The main goal is to research the application of intrusion tolerance to microservices to provide them with the capacity of guaranteeing reliable and trustworthy service even in the presence of security intrusions. Thus, exploring service proactive re-instantiation measures without harm to the availability of the system, besides geographical replication for redirecting service traffic and overcome security breaches without compromising service security attributes (confidentiality, integrity, availability).

We will develop a novel approach for quantifying systems capacity to tolerate intrusions. Devising a quantitative and meaningful metric which represents the tolerance capacity of a system based on its age (systems become more susceptible to software failure due to aging), its known vulnerabilities (software stack and supporting infrastructure), and previous attempted or successful intrusions. This metric would be continuously computed and monitored allowing the application of proactive measures to increase its security level. For systems, the metric would be computed through the aggregation of the weighted value observed for each individual service according to their criticality level.

Additionally, evaluating and analysing intrusion tolerance techniques effectiveness when introduced in the design of IDSes (e.g., redundancy, multiple sources of information), which would allow to make IDSes more resistant to attacks while producing satisfactory results. Also, research the possibility to protect IDS execution through the utilisation of trusted execution environments.

C. Container-aware Harm Assessment for Automation of Security Mitigation

Attention given to microservices has increased recently [5], nevertheless there is still a lack of effective container-oriented harm assessment measures [6]. Thus, researching the combination of diverse harm assessment techniques to obtain an improved approach for deciding the mitigation measure to apply is relevant. This work will allow the application of more cost-effective mitigation measures according to the level of criticality of each service.

To select the most appropriate measure, it is relevant to evaluate current reactive mitigation measures (e.g., live migration, IP shuffling [5], [7]), according to multiple applicable scenarios to reduce downtime and improve effectiveness. Also, formalise the scenarios identified so that it is possible to automate the decision process based on it.

D. An Integrated Approach for Intrusion Detection and Tolerance for Microservices

To produce a more complete and robust contribution, we aim at integrating both main improvements and develop a cooperating solution.

This task extends the work previously produced and intends to integrate the contributions to produce a complete approach following a defence-in-depth tactic through identification of security breaches and compromised services (intrusion detection) and application of mechanisms which allow to provide the expected service even though security breaches may take place (intrusion tolerance).

Thus, the result of the work conducted during this PhD will be an approach for cooperating detection and tolerance of security intrusions. Such approach will be implemented into a tool in order to experimentally validate it.

III. Current Work and Next Steps

Currently, the focus of the work has been on intrusion detection for container-based systems. We have explored the capacity of state-of-the-art intrusion detection algorithms to generate stable and complete profiles of containers running applications and evaluated their effectiveness in container-based system. At the moment, we are initiating work focused on the application of the devised approach and studied methods in the context of microservice applications.

A. Current Work

The preliminary results focused on studying the convergence capacity of the algorithms Sequence Time-Delaying Embedding (STIDE) and Bags of System Calls (BoSC), commonly used for intrusion detection [4].

In this experiment, benign traces were collected from Docker and LXC containers running MySQL server, which received the workload produced by an implementation of the TPC-C Benchmark (*tpc.org/tpcc*), configured with `100` warehouses and using `50` clients during workload execution. Two runs of collections during `10H` and `24H` were conducted.

Following the collection, we analysed the behaviour database growth through the curve slope computation building on prior work from [8]. We devised a convergence evaluation mechanism in order to conclude training procedures.

In Fig. 2, the blue dots represent the moments where the slope condition is satisfied; however, the sequential number of intervals are not enough to consider the learning process as completed. The green dots represent the four-preceding steady-state intervals to the interval where the classifier reaches the learning steady-state, represented by the red dot where the learning procedure ends.

Further work focused on defining a methodology to evaluate and compare different intrusion detection algorithms in a representative and fair manner. Producing meaningful and representative datasets in the context of container-based applications, which were not available [9].

In sum, our results demonstrate the applicability of state-of-the-art intrusion detection techniques to container-based deployments. Our approach for evaluation of such methods also proves worthwhile and able to portray meaningful results, demonstrating the capability of different algorithms to detect attacks with high detection rates while keeping low false positives, producing consistent outcomes which confirms that alarms are the result of quality profiles [10]. Furthermore, the initial results also show potential of our approach and methodology in the context of microservice applications.

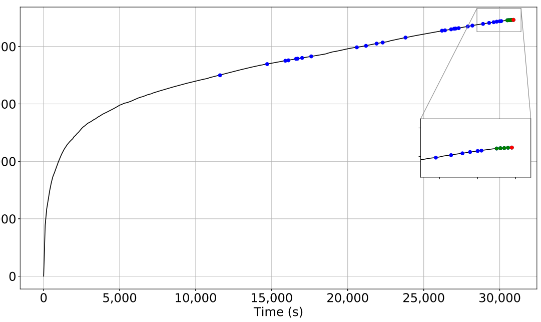

Fig. 2. Training procedure for STIDE with window 4 of 24h collection for Docker container.

B. Next Steps

The current work already allows to identify the potential of the methodology applied; however, there are still some limitations that are required to be resolved. Thus, the next steps will rely on devising and implementing a method that is able to overcome them and improve the precision of the approach utilised, resulting in a more effective technique for intrusion detection in scalability contexts.

Further work will be focused on studying the reutilisation of profiles across different versions and releases of the same service. For this, we will leverage the already developed target infrastructure and adapt the approach to evaluate the effectiveness of profile reuse. For this, we will monitor different versions of a service under the same operating conditions to evaluate the capacity of developing reusable profiles.

Furthermore, our attention will also shift towards studying, defining an approach to evaluate and conduct an evaluation of intrusion tolerance methods for microservices. Constructing a richer and stronger security approach in the scope of microservices.

IV. Related Work

Intrusion detection is the process of monitoring system events and analysing them for possible incidents [11]. Throughout the last decades this technique has been applied to multiple contexts [9], [12], [13], and recently attention has been shifting towards microservices architectures despite existing limitations [5], [7]. Microservices are mostly deployed using containers due to their simplicity and lighter-weight [2], [7]. Some attempts at container-based intrusion detection have been conducted [9], [14], but these works lack the representativeness of more complex and realistic scenarios. Our preliminary work on host-based anomaly detection for container-based systems [4] demonstrated the technique's applicability.

Intrusion mitigation techniques have been studied gradually as reactive measures which come into effect when an intrusion is detected and reported [15] . The high complexity of production microservice-based systems and the lack of diversity during development caused by technology reuse allows attackers to also reuse exploits against several services in the architecture [6] and perform lateral movement within the platform. Thus, mitigation measures intend to reduce and alleviate the effects of an attack. Therefore, mitigation measures are used as a last resource since when applied service availability may be compromised [16].

Intrusion tolerance [17], [18] allows systems to continue providing the expected service even when security intrusions occur [16], [19]. Tolerance can be proactively included into the architectural design of a system [19], through the application of independent replication and redundancy of services and components or data scattering, or by applying additional measures such as service proactive migration of resources when parts of the infrastructure stack is compromised [20].

Protecting the IDS itself is also extremely relevant. Thus, intrusion tolerance has also been used as a manner to increase the levels of trustworthiness of the IDS and protect them from crumbling to attackers [19], [20]. However, these techniques are yet to be applied to IDSes focused on either containers or microservices. Besides, IDSes can also be protected using hardware-based trusted execution environments (e.g. Intel SGX [21]) , which provide integrity and confidentiality guarantees to applications.

V. Conclusions

This project expects to improve the security of container-based microservices through the detection, tolerance and mitigation of security intrusions. The ultimate result of this work is an integrated approach for securing microservices using host-based intrusion detection in microservices and a generic approach of design and additional mechanisms for intrusion tolerant microservice-based systems.

`Tentative Submission Date: 2023`

Acknowledgments

This work is being conducted under the supervision of Nuno Antunes, University of Coimbra. This work is supported by the project AIDA – Adaptive, Intelligent and Distributed Assurance Platform (`POCI-01-0247-FEDER-045907`), co-financed by the European Regional Development Fund (ERDF) through the Operational Program for Competitiveness and Internationalisation (COMPETE2020) and by the Portuguese FCT under CMU Portugal. It is also supported by the project *TalkConnect: Voice Architecture over Distributed Network* (`POCI-01-0247-FEDER-039676`), co-financed by the ERDF, through Portugal 2020 (PT2020), and by COMPETE2020 and FCT. Developed within the scope of `CISUC - UID/CEC/00326/2020` financed by European Social Fund through CENTRO2020.

References

[1] S. Newman, *Building Microservices: Designing Fine-Grained Systems.* "O'Reilly Media, Inc.", 2015.

[2] N. Dragoni, S. Giallorenzo, A. L. Lafuente, M. Mazzara, F. Montesi, R. Mustafin, and L. Safina, “Microservices: yesterday, today, and tomorrow,” in *Present and ulterior software engineering*. Springer, 2017, pp. 195–216.

[3] S. Sultan, I. Ahmad, and T. Dimitriou, “Container Security: Issues, Challenges, and the Road Ahead,” *IEEE Access*, vol. 7, pp. 52 976–52 996, 2019.

[4] J. Flora and N. Antunes, “Studying the Applicability of Intrusion Detection to Multi-Tenant Container Environments,” in *2019 15th European Dependable Computing Conference (EDCC)*, 2019, pp. 133–136.

[5] A. Pereira-Vale, G. Márquez, H. Astudillo, and E. B. Fernandez, “Security Mechanisms Used in Microservices-Based Systems: A Systematic Mapping,” in *2019 XLV Latin American Computing Conference (CLEI)*, 2019, pp. 01–10.

[6] K. A. Torkura, M. I. Sukmana, A. V. Kayem, F. Cheng, and C. Meinel, “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*, 2018, pp. 932–939.

[7] A. Hannousse and S. Yahiouche, “Securing Microservices and Microservice Architectures: A Systematic Mapping Study,” *arXiv:2003.07262 [cs]*, 2020. [Online]. Available: http://arxiv.org/abs/2003.07262

[8] A. Milenkoski, B. D. Payne, N. Antunes, M. Vieira, S. Kounev, A. Avritzer, and M. Luft, “Evaluation of Intrusion Detection Systems in Virtualized Environments Using Attack Injection,” in *Research in Attacks, Intrusions, and Defenses*. Springer International Publishing, 2015, vol. 9404, pp. 471–492.

[9] A. S. Abed, C. Clancy, and D. S. Levy, “Intrusion detection system for applications using linux containers,” in *International Workshop on Security and Trust Management*. Springer, 2015, pp. 123–135.

[10] J. Flora, P. Gonçalves, and N. Antunes, “Using attack injection to evaluate intrusion detection effectiveness in container-based systems,” in *25th IEEE Pacific Rim International Symposium on Dependable Computing (PRDC2020)*, 2020.

[11] R. Bace and P. Mell, “NIST Special Publication on Intrusion Detection Systems,” NIST, NIST Pubs 800-31, 2001.

[12] S. A. Hofmeyr, S. Forrest, and A. Somayaji, “Intrusion detection using sequences of system calls,” *Journal of Computer Security*, vol. 6, no. 3, pp. 151–180, 1998.

[13] S. Bharadwaja, W. Sun, M. Niamat, and F. Shen, “Collabra: A Xen Hypervisor Based Collaborative Intrusion Detection System,” in *2011 Eighth International Conference on Information Technology: New Generations*, 2011, pp. 695–700.

[14] X. Lin, L. Lei, Y. Wang, J. Jing, K. Sun, and Q. Zhou, “A Measurement Study on Linux Container Security: Attacks and Countermeasures,” in *34th Annual Computer Security Applications Conference (ACSAC '18)*. Association for Computing Machinery, 2018, pp. 418–429.

[15] G. M. Sobchuk, S. Subramaniam, K. Zaheer, A. Gelfenshteyn, R. Shetty, M. Brady, G. Donnegan, and R. McGuire, “Network intrusion mitigation,” US Patent US7 676 841B2, 2010. [Online]. Available: https://patents.google.com/patent/US7676841B2/en

[16] V. Stavridou, B. Dutertre, R. Riemenschneider, and H. Saidi, “Intrusion tolerant software architectures,” in *Proceedings DARPA Information Survivability Conference and Exposition II. DISCEX'01*, vol. 2, 2001, pp. 230–241 vol.2.

[17] Y. Deswarte, L. Blain, and J.-C. Fabre, “Intrusion tolerance in distributed computing systems,” in *Proceedings. 1991 IEEE Computer Society Symposium on Research in Security and Privacy*. Oakland, CA, USA: IEEE Comput. Soc. Press, 1991, pp. 110–121.

[18] P. E. Veríssimo, N. F. Neves, and M. P. Correia, “Intrusion-Tolerant Architectures: Concepts and Design,” in *Architecting Dependable Systems*. Springer Berlin Heidelberg, 2003, vol. 2677, pp. 3–36.

[19] I. Welch, J. Warne, P. Ryan, and R. Stroud, “Architectural Analysis of MAFTIA's Intrusion Tolerance Capabilities,” *Deliverable 99*, 2003.

[20] L. Kuang and M. Zulkernine, “An Intrusion-Tolerant Mechanism for Intrusion Detection Systems,” in *2008 Third International Conference on Availability, Reliability and Security*, 2008, pp. 319–326.

[21] V. Costan and S. Devadas, “Intel SGX Explained,” *IACR Cryptology ePrint Archive*, vol. 2016, no. 086, pp. 1–118, 2016.