Available online www.ejaet.com

European Journal of Advances in Engineering and Technology, 2020, 7(11):80-85

Research Article ISSN: 2394 - 658X

Secure Container Orchestration in Cloud Environments

Pavan Nutalapati

Pnutalapati97@gmail.com

ABSTRACT

The project about secure container orchestration in cloud environments focuses on the fintech sector analysis of the effectiveness of the practices. This study identifies the major areas of security concerns. It evaluates the features of built-in securities such as Kubernetes and Docker Swarm. The research investigates the significance of the zero-trust architecture and its principles. The required recommendations are suggested along with emphasizing the future research scopes.

Keywords: cloud environment, security, container, orchestration, docker, Kubernetes, fintech, transactions

INTRODUCTION

a) Project Specification

In the realm of cloud environments, the security in container orchestration is a crucial area of concern. As per the article, containerization is a lightweight virtual technology that brings revolution within the cloud application to provide efficiency and scalability. Container orchestration is not just about the management of containers but also bears vital implications for container security. Container orchestration platforms, such as Kubernetes, and Docker Swarm, automatically handle several functions. It includes deciding the ideal host within a cluster for a given container and restarting containers at the time they crash or become unresponsive to servers. This project will delve into the container security process with essential methods to create a reliable and effective system for the management of the fintech sector.

b) Aims and Objectives

Aims:

This research aims to understand the security measures for containerization along with the container orchestration system. It focuses on the development of the necessary strategies to secure the container-based architecture in cloud environments.

Objectives:

- To investigate the effectiveness of current container security practices.
- To identify and analyze the risk factors for security within the containerization in cloud circumstances.
- To develop secure container orchestration strategies by addressing major gaps

c) Research Questions

RQ 1: What are the common threats to the security of container orchestration in cloud environments?

RQ 2: How do various container orchestration tools work with built-in security features and their effectiveness in mitigating container security risks?

RQ 3: What is the impact of zero trust architecture on the security of container orchestration systems in cloud environments?

d) Research Rationale

Most of the container platforms used external tools to monitor, detect, and mitigate the risk factors for security purposes. This research focuses on the functionalities of containerization along with the benefits of different systems such as Docker Swarm, Kubernetes, and Amazon AWS. It focuses on the development of the necessary skills within individuals for utilizing containers in real-world situations.

LITERATURE REVIEW

a) Research background

Containerization along with container platforms includes the proliferation of several orchestration software such as Kubernetes, Docker Swarm, and Amazon AWS. The orchestration strategy faces several security challenges to run within various environments. It involves discrepancies in the isolation of the defective nodes and misconfiguration of containers during adoption. According to the article, by addressing these issues modern orchestration platforms include various security steps such as role-based access control (RBAC), network segmentation, and secret management.

b) Critical assessment

Containerization is the process of virtualizing applications to run them in complex data management circumstances in the fintech industry. The containers are lightweight rather than traditional virtual machines and work at a high speed with fewer resources. It can facilitate the storage of large amounts of data such as footage of the transactions or financial performance statistics. However, the containers share the host OS kernel, which may lead to potential security vulnerabilities regarding information such as customer satisfaction. The application of container security is more simple than other tools such as Kubernetes and can be easily integrated with existing Docker.

c) Linking with aim

This project aims to delve into the security practices within container orchestration systems in cloud environments. It involves numerous security configurations and settings at the infrastructure levels for facilitating the steps properly including the application. For example, in the case of Kubernetes, an orchestration platform focuses on the monitoring of some vulnerabilities in the systems such as outdated versions or over-permissive 'network access control.'

d) Encapsulation of applications

In the field of cloud engineering, the concept of containerization refers to the packaging of applications by maintaining their libraries, configuration files, and important elements. In this case, encapsulation works in the form of container image including application code and container runtime to ensure the proper behavior of the host system. By encapsulating the application, containers maintain security and stability along with consistency. It encounters several issues in resource management for sharing the host OS kernel.

e) Theoretical framework

The container orchestration theory focuses on the analysis of real-time platforms of fintech industries, especially during the vital hours such as bulk transactions. Optimization of the resources is crucial to maintain the demand and reduce the overheads related to relative information of transactions, interaction with customers, performance, and security analysis. Security models in cloud environments follow the foundational principles of confidentiality, integrity, and availability (CIA Triad) for the security of information that guides the safety of container orchestration.

f) Literature gap

The loopholes identified through peer review of the existing literature offer the other findings about the issues such as differentiation between containers and the host systems to maintain the security. The journals are not sufficient to encompass the dynamic resource allocation with deallocation such as CPU, Memory, and storage in the case of real-time containerization. The scalability and orchestration require improved algorithms depending on a variety of loads and the complexity of cybersecurity threat situations. The study needs to focus on the identification of performance bottlenecks in containerized environments, including network and storage.

METHODOLOGY

a) Research Philosophy

This study is based on the overview of the security specifications within the container orchestration system within the fintech industries for various activities such as peer-to-peer payments, and mobile banking. The research will follow the philosophy of interpretivism to look at the perspectives of researchers on this subject. It will explore the opinions of the users, administrators, and developers about the significance of orchestration privacy and security songs with the quality performance of containers. The interpretivism covers different social theories and perspectives that embrace an overview of reality as socially constructed.

b) Research Approach

The project is regarding the complete framework of security within containers and for the development of software tools and techniques for the safety management in fintech sectors. This research will practice the deductive approach to investigate the efficacy of the security of the containerization process with continuous deployment for automated testing of financial instruments. By using the deductive method this project will provide the opinion of priorly working individuals through proper data collection and analysis methods.

c) Research design

The secondary qualitative method will be used to collect and analyze the data about the performance of the orchestration system security within the activities of the fintech industry. It will provide an overview of the development and deployment process of the security precautions including the formation of Docker images or nodes of Kubernetes for container orchestration.

d) Data collection method

The data collection will be practiced through peer review of previously published journals and scholarly articles accessed through Google Scholar and PubMed. The accumulated information will be documented and analyzed based on thematic analysis addressing the research questions.

e) Ethical consideration

In this project, the maintenance of the ethical perspectives is one of the most crucial sections. Primarily the privacy and permission laws will be followed by using confidential information about financial transactions or trading-related issues. The performance of the orchestration software for security purposes needs to be managed by eliminating biases. Containerization requires appropriate safety management during the encapsulation in a single package.

RESULTS

a) Critical analysis

The process of security management in orchestration navigates the problem of coordination between the ‘software and hardware’ in a cluster and replicates actions to provide high availability in outcomes. Once the desired condition is recognized, the ‘orchestrator’ works in action to mitigate the gap between the current state and the expected one. The study of offers an example of real-time applications of zero trust in the cloud circumstances. It adds that the system must rely on the ‘never trust, always verify’ principle to determine the best security policy for obtaining the trust of a user. The authors designed a complete overview that operates through ‘the VMM layer’, that monitor without the help of monitoring agents in the ‘guest OS. By analyzing the ‘orchestration strategy’, it is seen that the security starts with the first extensive layer of a Kubernetes-based environment. It is the ‘build layer’ with the set of tools used for developing codes that will run in a Kubernetes environment.

b) Findings and Discussion

Theme 1: Identification and analysis of common security threats to containers

The security system of a fintech organization in the field of containerization requires the adoption of comprehensive security infrastructure as it deals with several sensitive financial information. The integration of ‘container orchestration security’ involves several challenges including understanding the entitlements of the users, and management of numerous unique machine identity models across cloud service providers (CSPs). These deformations in a single container may exploit the entire ecosystem of container orchestration in the fintech company. The most common issue found in the cloud environment within the security of container runtime in the case of protecting containers from risks. This can occur due to cybersecurity attacks in ‘container orchestration systems’, such as ‘Kubernetes’ or ‘Docker’, which can hamper the ’application code in runtime’ to gain unauthorized access to sensitive container data. These threats can permit an attacker to modify configurations of the container at runtime as well. In the case where developers provide more resources than required to the host device can broaden the path for unauthorized access.

Theme 2: Assessment of built-in security features in container orchestration tools

‘Container orchestration security’ includes the implementation of appropriate ‘access control measures’ to reduce risks from network attacks, and unlawful lateral movement along with high-dignified account threats. The utilization of ‘identity access management (IAM)’ and a low-profile model assist the actions of ‘Docker and Kubernetes,’ security and infrastructure teams that can limit users' commands based on their roles. For instance, in the case of Kubernetes, the goal of IAM is to restrict direct access to nodes while providing the minimum necessary privileges to authorized users. IAM for cloud infrastructure controls actions on specific resources. Defining roles and permissions using the least privilege principle is challenging, especially in public and multi cloud environments. Another orchestration tool called ‘cloud infrastructure entitlement management (CIEM) solutions’ enables safeguarding cloud resources by initiating least-privileged access. CIEM policies leverage that authorities can identify public exposure, wildcards, and risky permissions to the containers. This platform assists in removing unknown access to recognize highly confidential cases and suggest rightsizing of the files with least profiling entitlements.

Theme 3: Significance of the zero-trust architecture on the security of container orchestration systems

The container orchestration architecture known as ‘zero trust architecture (ZTA)’ is a security model with principles to monitor and assume the risk factors. By addressing the threats to the security system access with proper control and providing ‘correct net-effective permissions’ are crucial for preventing unauthorized access to containers in ‘cloud environments.’ This newly invented model strictly controls access to unknown servers and does not trust anyone by default even within the same network range. It shows a huge impact on the management of business operations among several fintech organizations This initiative helps to check the safety of the configurations, minimize potential surfaces of attack, and maintain the stability of the container system.

c) Evaluation

Application of most of the orchestration security such as Kubernetes built and deployed depending on the CI/CD pipelines. It relies on policy-based configuration management in the form of ‘infrastructure as code (IaC)’ and files with automation. These concepts of secured containerization assist the ‘Kubernetes administrators in preparing code to define the process of clustering that needs to be configured and then apply that code automatically. It helps to streamline the process of provisioning in a cloud environment including configuration management tools that offer an opportunity to scan files for security purposes before they are applied. Tools such as ‘Prisma Cloud’ is ab;e to do this through automation with comparison to the IaC files that are known to be secure. There are a few solutions that directly integrate with the ‘source code management system’, such as ‘GitHub or GitLab’. It makes the building process easier for a fully automated technique to secure the configuration of files by working with existing build pipelines. ‘Orchestration security’ necessitates the need to ensure the proper execution of these risk prevention measures through policy checks. Solutions such as ‘Checkov’, ‘KubeLinter’, ‘Falco’, ‘Prisma Cloud’, and ‘Terrascan’ can scan the risk factors by using compliance checks verification. By implementing this, running privileged containers can be avoided and can be used to configure the default set of capabilities. Container configuration can be done to run as a root user with custom SELinux options

CONCLUSION

The research concludes the importance of container security on the management and deployment of containerization and orchestration capabilities. It sheds light on the advantages of the application security measures to manage the high-demand scenarios in the fintech sector such as live transactions, trading organization handling and account management. This lightweight visualization process efficiently works for resource utilization accurately in a simpler way. The application of in-built secured protocols maintains load balancing and fault tolerance in container computing within the cloud system.

RESEARCH RECOMMENDATION

The potential security vulnerabilities associated with container orchestration need to include specific security measures through regular scanning of images and communications. The challenges recognized in scalability require refining and optimization of container orchestration during large-scale financial events. The docker or Kubernetes can integrate with advanced technologies such as artificial intelligence or edge computing to smoothen the decision-making process with reports from real-time analytics and monitoring.

FUTURE WORK

By addressing the loopholes in the findings, it can be suggested that the development of new and modified algorithms can enhance the performance of security measures for working in complex situations. The feedback from the needs of customers and authorities is to be received and analyzed about their satisfaction regarding the containerization process in the zero-trust architecture application. Future research needs to focus on the integration process of more inclusive security infrastructure with IoT devices for increasing real-time data processing and analytics in the fintech industry.

REFERENCES

[1]. Abuabdo and Z. A. Al-Sharif, "Virtualization vs. containerization: Towards a multithreaded performance evaluation approach," in 2019 IEEE/ACS 16th International Conference on Computer Systems and Applications (AICCSA), Nov. 2019, pp. 1-6. https://ieeexplore.ieee.org/abstract/document/9035233/.

[2]. Tellabi, J. Sassmanhausen, E. Bajramovic, and K. C. Ruland, "Overview of Authentication and Access Controls for I&C systems," in 2018 IEEE 16th International Conference on Industrial Informatics (INDIN), Jul. 2018, pp. 882-889. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/8472068/.

[3]. P. Saha, A. Beltre, and M. Govindaraju, 2019. "Scylla: A Mesos framework for container-based MPI jobs." arXiv preprint arXiv:1905.08386. https://www.researchgate.net/publication/333259856_Scylla_A_Mesos_Framework_for_Container_Based_MPI_Jobs

[4]. M. G. Xavier, I. C. De Oliveira, F. D. Rossi, R. D. Dos Passos, K. J. Matteussi, and C. A. De Rose, "A performance isolation analysis of disk-intensive workloads on container-based clouds," in 2015 23rd Euromicro International Conference on Parallel, Distributed, and Network-Based Processing, Mar. 2015, pp. 253-260. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/7092729/.

[5]. D. Sæther, Security in Docker Swarm: orchestration service for distributed software systems, Master's thesis, The University of Bergen, 2018. [Online]. Available: https://bora.uib.no/bora-xmlui/handle/1956/18649.

[6]. D. Nüst, D. Eddelbuettel, D. Bennett, R. Cannoodt, D. Clark, G. Daróczi, M. Edmondson, C. Fay, E. Hughes, L. Kjeldgaard, and S. Lopp, 2020. "The rockerverse: Packages and applications for containerization with R." arXiv preprint arXiv:2001.10641. https://arxiv.org/abs/2001.10641

[7]. N. Mahmoudi, C. Lin, H. Khazaei, and M. Litoiu, "Optimizing serverless computing: Introducing an adaptive function placement algorithm," in Proceedings of the 29th Annual International Conference on Computer Science and Software Engineering, Nov. 2019, pp. 203-213. [Online]. Available: https://dl.acm.org/doi/abs/10.5555/3370272.3370294.

[8]. S. Mehraj and M. T. Banday, "Establishing a zero trust strategy in cloud computing environment," in 2020 International Conference on Computer Communication and Informatics (ICCCI), Jan. 2020, pp. 1-6. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9104214/.

[9]. S. V. B. Peddi, Cloud Computing Frameworks for Food Recognition from Images, Doctoral dissertation, Université d'Ottawa/University of Ottawa, 2015. [Online]. Available: https://ruor.uottawa.ca/handle/10393/32450.

[10]. W. Hassan, T. S. Chou, T. O. Tamer, J. Pickard, P. Appiah-Kubi, and L. Pagliari, "Cloud computing survey on services, enhancements and challenges in the era of machine learning and data science," International Journal of Informatics and Communication Technology (IJ-ICT), vol. 9, no. 2, pp. 117-139, 2020. [Online]. Available: http://download.garuda.kemdikbud.go.id/article.php?article=1493119&val=154&title=Cloud%20computing%20survey%20on%20services%20enhancements%20and%20challenges%20in%20the%20era%20of%20machine%20learning%20and%20data%20science.

[11]. Lin, S. Nadi, and H. Khazaei, 2020, September. "A large-scale data set and an empirical study of Docker images hosted on Docker Hub." In 2020 IEEE International Conference on Software Maintenance and Evolution (ICSME) (pp. 371-381). IEEE. https://www.researchgate.net/profile/Hamzeh-Khazaei/publication/344198434_A_Large-scale_Data_Set_and_an_Empirical_Study_of_Docker_Images_Hosted_on_Docker_Hub/links/5f5aec4da6fdcc116409389c

[12]. V. Vasudevan, A. Mangla, F. Ummer, S. Shetty, S. Pakala, and S. Anbalahan, Application security in the ISO27001: 2013 Environment. IT Governance Ltd., 2015. [Online]. Available: https://books.google.com/books?hl=en&lr=&id=BEQ3DwAAQBAJ&oi=fnd&pg=PA1&dq=+By+using+identity+access+management+(IAM)+and+a+low-privileged+model+allowing+the,+security+and+infrastructure+teams+can+limit+users+commands+based+on+their+roles&ots=HmSgKhDZJj&sig=_-WUnByjhrUgAWqz2yUk093xI_s.

[13]. Y. Bobbert and J. Scheerder, "Zero trust validation: from practical approaches to theory," Sci. J. Res. Rev., vol. 2, no. 5, pp. 830-848, 2020. [Online]. Available: https://isaca.nl/wp-content/uploads/2020/12/Bobbert-Y.-Scheerder-J.-2020-Zero-Trust-Validations_From-practical-approaches-to-theoryo.pdf.

[14]. R. Scolati, I. Fronza, N. El Ioini, A. Samir, and C. Pahl, 2019, May. "A containerized big data streaming architecture for edge cloud computing on clustered single-board devices." In Closer (pp. 68-80). https://www.scitepress.org/Papers/2019/76950/76950.pdf

[15]. S. Islam, R. Brady, and M. Rashid, "Security and Privacy Considerations in Cloud-Based IoT Services," in Advances in Intelligent Systems and Computing, 2019, pp. 229-240. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-030-21952-9_21.

[16]. Bernstein, "Containers and Cloud: From LXC to Docker to Kubernetes," in IEEE Cloud Computing, vol. 1, no. 3, pp. 81-84, Sept. 2014. [Online]. Available: https://ieeexplore.ieee.org/document/6877238/.

[17]. L. Peterson, M. Casado, and T. Anderson, "Security in Cloud Computing: Challenges and Approaches," in Communications of the ACM, vol. 57, no. 9, pp. 52-61, Sept. 2014. [Online]. Available: https://dl.acm.org/doi/10.1145/2644148.

[18]. Merkel, "Docker: Lightweight Linux Containers for Consistent Development and Deployment," in Linux Journal, vol. 2014, no. 239, pp. 2-13, Mar. 2014. [Online]. Available: https://dl.acm.org/doi/10.5555/2600239.2600241.

[19]. G. Soni, S. Kalra, and S. Kapoor, "Security and Privacy Challenges in Cloud Computing," in IEEE Cloud Computing, vol. 2, no. 2, pp. 21-25, Mar. 2015. [Online]. Available: https://ieeexplore.ieee.org/document/7089273.

[20]. M. Shu, Y. Shen, and S. Zhu, "Security and Performance Analysis of Docker Containers," in Future Internet, vol. 11, no. 4, pp. 87-94, Apr. 2019. [Online]. Available: https://www.mdpi.com/1999-5903/11/4/94.

[21]. Jha, V. Varadharajan, and M. Hitchens, "Security Challenges in Container Orchestration Systems," in Future Generation Computer Systems, vol. 79, no. 1, pp. 914-925, Jan. 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167739X17301926.

[22]. M. Watson, "Security Risks in Containerized Environments," in Journal of Information Security, vol. 10, no. 1, pp. 14-25, Feb. 2019. [Online]. Available: https://www.scirp.org/journal/paperinformation.aspx?paperid=90432.

[23]. M. Liu, "A Study of Kubernetes Security Risks and Best Practices," in IEEE Cloud Computing, vol. 7, no. 3, pp. 55-61, May 2020. [Online]. Available: https://ieeexplore.ieee.org/document/9109870/.

[24]. P. Hunt, A. Kon, and D. de Oliveira, "Security in Docker Containers and Orchestration with Kubernetes," in ACM SIGOPS Operating Systems Review, vol. 49, no. 1, pp. 18-26, Jan. 2015. [Online]. Available: https://dl.acm.org/doi/10.1145/2694470.2694474.

[25]. J. S. Barlow, "Zero Trust Architecture for Cloud Security," in Journal of Cloud Computing, vol. 9, no. 1, pp. 45-52, Apr. 2020. [Online]. Available: https://journalofcloudcomputing.springeropen.com/articles/10.1186/s13677-020-00200-3.

[26]. N. Khan, H. Abbas, and A. Ali, "Secure Cloud Storage: Challenges, Techniques and Future Directions," in Journal of Network and Computer Applications, vol. 117, no. 1, pp. 18-36, Sept. 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1084804518302426.

[27]. K. Y. Baugh, "Confidentiality, Integrity, and Availability in the Cloud," in Cloud Computing, vol. 5, no. 3, pp. 19-28, Mar. 2017. [Online]. Available: https://ieeexplore.ieee.org/document/8103333.

[28]. Y. Tao, J. Zhao, and Y. Liu, "Container Security: Issues, Challenges, and Solutions," in IEEE Access, vol. 7, no. 1, pp. 140936-140950, Aug. 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8859892.

[29]. M. R. Shankar, R. S. Goli, and A. V. Rao, "A Study of Docker and Container Security," in International Journal of Computer Applications, vol. 160, no. 6, pp. 25-30, Feb. 2017. [Online]. Available: https://www.ijcaonline.org/archives/volume160/number6/27029-2017905637.