Cloud Based Environmental Security Analysis: An Evaluation of Techniques Employed to Secure Cloud Environments, their Effectiveness and the Risks Posed by Malicious Software

Finlay Reid
School of Design and Informatics
Abertay University
DUNDEE, DD1 1HG, UK

ABSTRACT

Context: With the emergence of cloud computing, a revolutionary approach to sharing resources, the hyperbole around its security issues has only increased. This evolution of computing technology which relies on both legacy and existing technologies has seen a dramatic increase in both users and businesses move to its service. With this shift to cloud-based environments, the risks posed by malicious software has remained the same. Containers, a lightweight alternative to virtual machines have seen great uptake from users although container-based security has remained archaic.
Aim: The following project aims to complete a thorough exanimation of cloud based environmental security including containers, compare different hardening methods and assess the risks posed by malicious software. In completing this project users will become more educated on the subject through a in depth container security plan. Furthermore, the completion of this project will hopefully prompt further discussion pertaining to container and cloud security.
Method: To examine and understand container-based security, test containers with different properties will be created. This will enable an analysis of several hardening techniques. Following on from the initial testing, malicious exploits used to negate security measures will be both demonstrated and analysed allowing for a better understanding of container security. The data from the results will be used to develop and implement a recommended security plan for container-based environments.
Results: The expected results of the work should illustrate the security weaknesses found within cloud computing namely with containers. An evaluation of malicious exploits used to disrupt cloud software will demonstrate the ease in which attackers can infect cloud-based systems. This will enable a discussion of comparing traditional malware with cloud-based malware.
Conclusion: In conclusion, the following project will demonstrate the vulnerable nature of cloud computing in its current state and seek to minimize the risks posed by malicious exploits. This will be achieved through the meticulous analysation of both techniques used to secure and exploit, cloud-based systems.

Keywords

Cloud malware, Virtualisation, Cloud based systems, Cloud security, Containers, container Security

1. INTRODUCTION

Cloud computing at its core is a way of sharing resources over the internet. It's a combination of previous technologies to enable convenient network access to a shared pool of resources which can be customised and deployed with ease. The introduction of concepts that would form the basis of cloud computing began during the 1960s, this revolutionary approach to information technology was known as utility computing. Its similarity to modern day cloud computing cannot be underestimated, with its primary goals of coalescing applications, storage and servers to share with users. Just like its predecessor, cloud-based systems suffer from the same weaknesses found in traditional computing, the vulnerability of malicious exploits infecting the machine/network compromising critical information. Figure 1 below shows the evolution of cloud computing.

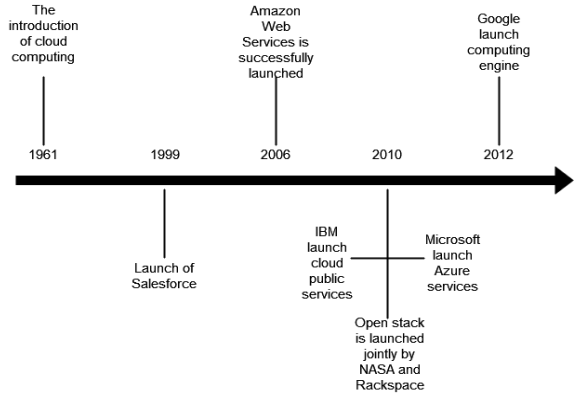

Figure 1 : Cloud Computing History

Currently businesses use an average of 1,181 cloud services with a shocking 92.7% of them not being secured or fit for use according to a survey done by Netskope (netskope, 2021). This research demonstrates why cloud security is such an important topic in modern computing: as the popularity of the model continues to increase more challenges will present themselves. As cloud computing has evolved, the model has progressed to suit different needs for different users. Both deployment and service models allow the cloud to be tailored to that specific business or user, deployment models describe where the core infrastructure is located and who controls this. While service models enable users to choose their billing systems, basic functionality and business model.

Containers, a lightweight alternative to virtual machines significantly increased the use of virtualization by users, allowing software to be ran without worrying about platform compatibility. Within the cloud, containers can be deployed with relative ease, a number of companies provide both cloud-based services and a container-based hosting solution. Virtualization shares the same basic principles as the cloud, it can be implemented at the server level or operating system level, these terms are sometimes wrongly used interchangeably. The technologies and protocols of virtualization are what allow most modern cloud-based services to function.

1.1 Containers

Linux container-based virtualization functions through four key components: userspacetools, cgroups, images and namespaces. A container essentially uses the host operating system and kernel to ensure isolation, this is achieved through the use of two kernel components namespaces and cgroups. Responsibility of recording resource usage falls to the Linux feature cgroups, which was created by google engineers in 2006. Cgroups enable users to restrict the number of resources used by the CPU, memory and disk. Namespaces are closely related to the cgroup component of the kernel, they slice the operating system into virtual sections, working hand in hand to allow the container to function. The kernel in the host machine is shared by the container allowing for a significant performance increase when compared with hypervisor-based virtualization. As the instructions are not wastefully passed through extra layers and the machine has to complete less work when doing system calls. Another benefit when using container-based virtualization includes the scalability, whether done horizontality or vertically. This allows users to add more resources like upgrading the CPU or add additional machines to the pool of computing resources. With the rising popularity of container-based virtualization there has been host of papers published concerned over the weaknesses found within containers.

1.2 Kubernetes

Kubernetes, also known as K8s, was developed for the purpose of managing virtualized container environments. The software debuted on the 7th of June 2014, one year after its competitor docker. This open-source management system has its own terminology and architecture, for example pods – a group of containers are run by nodes – a machine which collects entire pods which work in unison. Kubernetes is currently used by a host of major companies ranging from Amazon to VMware, displaying how crucial container-based security is in today's technological climate.

2. BACKGROUND

With the significant growth of container-based virtualization the secureness of these isolated environments has become a topical subject in recent years. However there has been a lack of substantial literature relating to container security. Furthermore, the documentation presented has been for specific use cases. Concerns over how containers share the host OS have been raised by security professionals as it gives malicious users a springboard to infect the host machine (S. Sultan, I. Ahmad and T. Dimitriou, 2019). An assessment over the potential risks posed by malware breaking out of its container was undertaken by the Beijing School of Cyber Security. In this experiment exploits classified as Privilege escalation, denial of service and others were deployed within an isolated container. A total of 88 exploits were employed to assess a containers ability of ensuring complete isolation from its host. The results from this experiment reinforce the concerns held by security professionals on container security, 56.2 % of the exploits successfully executed on the container platform. This highlights the risks for the host system posed by weak container security as the container fails to provide any extra safeguards for the applications inside of it (Lin and Sun, 2018).

The previously mentioned cgroup component of a container helps in protecting the host from Dos attacks with its ability to limit resources. However, the protection is far from sufficient as malicious exploits could still make use of intensive access for that amount of memory (Chen, Feng and Wen, 2018). As containers continue to share the Linux kernel of its host system, privilege escalation will remain a point of weakness within container-based virtualization. In fact, it can be seen as the greatest flaw within the container eco system as the isolation provided by containers could be completed negated. Overall, the inner security workings of a container mainly the c group and namespace mechanisms managed to defend against 21.62% "privilege escalation" exploits (Chen, Feng and Wen, 2018). This demonstrates the necessity of this project in assessing container security, presenting the best practices for users to remain secure against Dos and privilege escalation exploits.

Kubernetes, a provider of container-based virtualization technologies has had their own set of security issues. As Recent as 2018, users with ill intentions gained unauthorized access to Teslas Amazon Web Services resources by exploiting a vulnerability in a Kubernetes console (Goodin, 2021). Hardening orchestration systems such as Kubernetes require users to take the initiative to proactively seek weaknesses found within. The software has inbuilt mechanisms to protect against exploits, but these are not sufficient, the onus is placed on the user to protect themselves when using this container orchestration software. An example of a small but substantial security vulnerability includes the failure of Kubernetes to encrypt its key to the internal userbase. Without additional configuration, Kubernetes saves critical information in 'etcd' – Kubernetes internal database that stores its configuration data, its state, and its metadata (Shamim and Bhuiyan, 2020). A malicious user could potentially use this to acquire crucial information such as passwords, usernames and database queries. Leveraging an exploit from this data would be straight forward, possibly infecting the whole network. From committing extensive research of Kubernetes and analyzing related papers to the subject area. It is clear there is not a robust plan of action for users to ensure their Kubernetes systems are secure and have the best possible protection against malicious exploits. The objective of this project is to put an end to this, creating an in-depth methodology tasked with hardening Kubernetes containers.

Cloud based systems is a topic this project will cover due to the close relationship between cloud computing and containerization. Whether using Saas or another service, each one is susceptible to their own security issues. This technology can also be used as a platform of attack, due to powerful nature of cloud networks. The following paper gives the opinion that the benefits of cloud systems are outweighed due to the security, safety and privacy challenges. It continues stating that complex cloud services are normally configured through web interfaces in which users are likely to make errors (Shaikh, 2021). Therefore, opening the system up to potential exploits. There is also a great variation in weaknesses ranging from Data loss to hijacking of sessions. Further on in the paper the author recommends The Cloud Security Alliance to organizations in order to minimize the risk posed by container weaknesses. Security within the cloud has improved over the years, however it is still a new technology with many faults. This project aims assess the current security used in modern cloud-based systems, giving organizations a base of knowledge in order to assist them when securing their cloud network.

3. METHOD

3.1 Research

The initial phase of the project will include a period of thorough research into the specifics of cloud and container exploits. Relevant literature including journals, papers and web articles will be reviewed to ensure the author has a firm grasp on the topic of choice. The work presented requires a good base knowledge of both container and cloud security before an attempt to exploit these technologies. To ensure the project has a chance of success, material applicable to the legality of the proposed work will be researched to ensure the test is carried out in a safe and legal manner. One of the main areas of research includes how to break into and out of containers. What vulnerabilities can be exploited to enable unauthorized access to the host or the container?

Specifics of the research involve looking at exploits that bypass container security mechanisms such as cgroups. Just from light research there is numerous papers and ways in which malicious users can exploit Kubernetes through privilege escalation. For example, by simply modifying a test pod resource YAML file and copying a snippet of code, users are able to exploit a weakness found in Kubernetes pods (Artem, Tetiana, Larysa and Vira, 2020) . Older exploits will still be covered as they often give clues to new or unfound exploits.

As this project covers cloud security in addition to containers, relevant security policies and exploits used to disrupt/negate the security measures of containers will be explored.

3.2 Set Up

Information regarding the specifics of setting up the project will be gathered during the initial research phase. However, a rough idea of the process is known. The set up for this project will involve the author first creating a Kubernetes cluster through Google Kubernetes Engine (GKE). Next serval container images with varying security configurations will be downloaded or created and then implemented onto the cluster. Necessary security measures will be put into place to ensure the home machine is not infected in process of testing. After successfully setting up the test containers, the cloud-based service will be set up. A service like AWS or a personal cloud system will be configured for the attacks. Software including VMware, Linux and Next cloud will be used to construct the cloud computing environment.

3.3. Implementation

After successfully researching the relevant topics and setting up the test environment, the implementation phase of the project will begin. Running exploits capable of targeting weaknesses found within containers and cloud computing will be analyzed. Discussing how they work, their countermeasures and the implications if this happened in a real-world situation. With the knowledge of frailties within these technologies an exploit will created, if the exploit is unable to be developed a script will be crafted with the intention of hardening container/cloud security.

3.4 Evaluation

Quantitative evaluation of this project will be completed by the creation of a in depth guide that aims to secure an organizations container and cloud computing environment. Using knowledge gained from the projects previous phases the most severe weaknesses of the technologies will be documented and discussed. A tier ranking system of the most pertinent exploits and weaknesses will be made to ensure users have a good understanding of the challenges faced when using these systems. After the creation of the python script or malicious exploit, it will be executed to determine the success of the program. The results will be thoroughly analysed, and relevant data will be evaluated to see if any improvements can be made or with additional time/resources what could be achieved. The main aim of the project is to accentuate the weaknesses found within cloud computing and containerization. Hopefully the project brings motivation to security professionals to further test and document ensuring these technologies can remain as secure as possible.

4. Summary

In summary, with the continuous growth of organizations switching to containerization and the cloud computing model. It remains paramount these new technologies are as secure as possible; through this project a complete security assessment of cloud-based systems and containers will allow users to harden their systems. As mentioned before one outcome of the project will be a program developed with the aim of exploiting or securing containers. The significant lack of literature relating to the security of containers was one of the main prompts to undertake this project. Finally, the data gathered regarding the results of the project and its induvial phases will be presented in a dissertation. Which will allow the work to be replicated, and to provide a critical analysis of the threats and weaknesses.

5. REFERENCES

Chen, J., Feng, Z., Wen, J.Y., Liu, B. and Sha, L., 2019, March. A container-based dos attack-resilient control framework for real-time uav systems. In 2019 Design, Automation & Test in Europe Conference & Exhibition (DATE) (pp. 1222-1227). IEEE.

Netskope. 2021. *Cybercriminals Find Cover in the Cloud: New Netskope Research Finds 44% of Threats are Cloud-Enabled* . [online] Available at: <https://www.netskope.com/press-releases/cloud-threat-report> [Accessed 15 October 2021].

Goodin, D., 2021. *Tesla cloud resources are hacked to run cryptocurrency-mining malware*. [online] Ars Technica. Available at: <https://arstechnica.com/information-technology/2018/02/tesla-cloud-resources-are-hacked-to-run-cryptocurrency-mining-malware/> [Accessed 19 October 2021].

Lin, X., Lei, L., Wang, Y., Jing, J., Sun, K. and Zhou, Q., 2018, December. A measurement study on linux container security: Attacks and countermeasures. In Proceedings of the 34th Annual Computer Security Applications Conference (pp. 418-429).

Shamim, S. and Bhuiyan, F., 2020. XI Commandments of Kubernetes Security: A Systematization of Knowledge Related to Kubernetes Security Practices.

Shamim, M.S.I., Bhuiyan, F.A. and Rahman, A., 2020, September. XI Commandments of Kubernetes Security: A Systematization of Knowledge Related to Kubernetes Security Practices. In 2020 IEEE Secure Development (SecDev) (pp. 58-64). IEEE.

Artem, L., Tetiana, B., Larysa, M. and Vira, V., 2020. Eliminating privilage escalation to root in containers running on kubernetes. Scientific and practical cyber security journal.