Webology, Volume 18, Number 4, 2021
ISSN: 1735-188X
DOI: 10.29121/WEB/V18I4/141

SECURITY IMPLICATIONS IN DOCKER BASED VIRTUAL ENVIRONMENT

Noor Mohd[1], Vivekanand Kuriyal[1], Deepak Upadhyay[2]

[1]Department of Computer Science & Engineering Graphic Era Deemed to be University, Dehradun, Uttarakhand, India
[2]Department of Computer Science & Engineering, Graphic Era Hill University, Dehradun, Uttarakhand India, 248002

ABSTRACT

There has been a significant growth in the use of virtualization technology in recent years. It raises the need for effective and stable approaches for virtualization are becoming more evident. Virtualization based on containers and based on hypervisors virtualization are two main forms of virtualization technology that have appeared on the market of the two grades, virtualization based on containers will provide more digital world lightweight and secure but not without security risk. There is no contact inside the VMs between the applications and the host kernel as the applications can interact only with the virtual machines kernel. To attack the kernel host, the attacker must bypass the hypervisor and the VM kernel host. This is not the case for Docker and all other methods of virtualization. Since the host kernel can be directly accessed by the applications in all of them, thereby making an attacker who targets the host kernel directly .That is one of the main reasons for this to answer questions and concerns on protection for Docker. And since the defense is such a big concern thus a range of attacks can occur in Docker such as SQL injection, exploitation of privilege, DOS, e.t.c In this Report, we analyze the security of Docker .To explain all of the possible conditions in which attacks can be carried out in Docker, a threat model for single host is proposed and along with an attack classification that describes what attacks can take place on which layers of Docker.

Keywords: Intrusion Detection System, Virtualization, VM

INTRODUCTION

This is the era of digital computing because everyone is using digital gadgets for their daily life activity. Virtualization increase usability of digital recourse very interesting. Container virtualization is generally used runtime execution and application covering. Containers application can service equipment all the organization libraries, code and files appropriate to support the object situation [1]. Using programmers, containers can expand their commodity more accurately and use them freely. Use of Docker is increase very fast compare to Virtualization technology.

Security is a most required thing in every field, In Virtualization security issue is also important. Virtualization (i.e., hypervisor-based technique) insists on being more secure as a contrast to container-based techniques as they measure an additional layer of confinement between the application or host. The application running in a virtual environment (VMs) can communicate only with VM hypervisor and not with the host hypervisor. So the attacker is mainly responsible for circumventing the hypervisor and the host kernels may target host kernels in a process.

However, applications can accurately get input and interact with host kernels with all container technologies, and then allow an intruder may precisely target a server, thereby allows the attacker could save a massive number of attempt while partitioning inside the target host. That creates a security problem with docker-container [2]. We test Docker 's internal protection because if a specific host system runs an amount of Docker containers to which a container subgroup agrees and the attacker has complete control over those, even if the extreme container subgroup is under authentic clients discipline.

The intruder can carry out various models of attacks with DOS and privilege escalation attack in this model category [2]. Although the kernel protection systems are exiting to be able to enhance the protection of a Linux host machine such as LSM (Linux Security component) and Linux capabilities. The LSM provides a system to support restricted security models in the Linux kernel.

Docker is a light weighted container technology with the potential ―to construct, run applications and export‖ [2].A few prominent applications used the Docker such as Spotify, EBay and Yelp.

Docker is almost recent and more managing applicant since it appears not to have acquired previous container technologies including the latest capabilities. In the case of Docker there are several advantages such as speed, fast delivery, density, flexibility [4]. However, there are still a few disadvantages. Docker is composed of two basic components: Docker Engine or Docker Server. One is a lightweight package tool focused on container-based virtualization, while the other is a Saas (Software-as-a-service) framework for custom Docker image allocation [2].

Over the last few decades, the development or importance of virtualization technologies has increased significantly. The need for useful and safe virtualization technologies that could provide an optimized, reliable, thick and secure environment has further expanded. A broad variety of these findings emerges that can be restricted to two crucial classes, i.e. container- based (also to be defined as OS-level virtualization) and hypervisor-based virtualization [3][4]. Container-based virtualization serves as a portable replacement for virtual machines or provides higher portable and usable user environments.

In addition, containers take less start-up time in 50milliseconds, while a VM will start in 30–40seconds, so virtualization based on containers is more powerful than VMs. Containers will use the same Kernel as a substitute for receiving a custom copy for each as in VMs. In the last few years the container platform rate has been rising. There are various container developments to access, such as the newest or predominant applicant being Linux-V server, LXCs, Open VZ, and Docker. Clearly Docker is the imminent creator of virtualization technologies, therefore users need to restore traditional virtualization effectively. There is no communication inside the virtual machines between the devices and the host kernel, as applications could only interact with kernel of the virtual machine. As talked about IDS in [20].

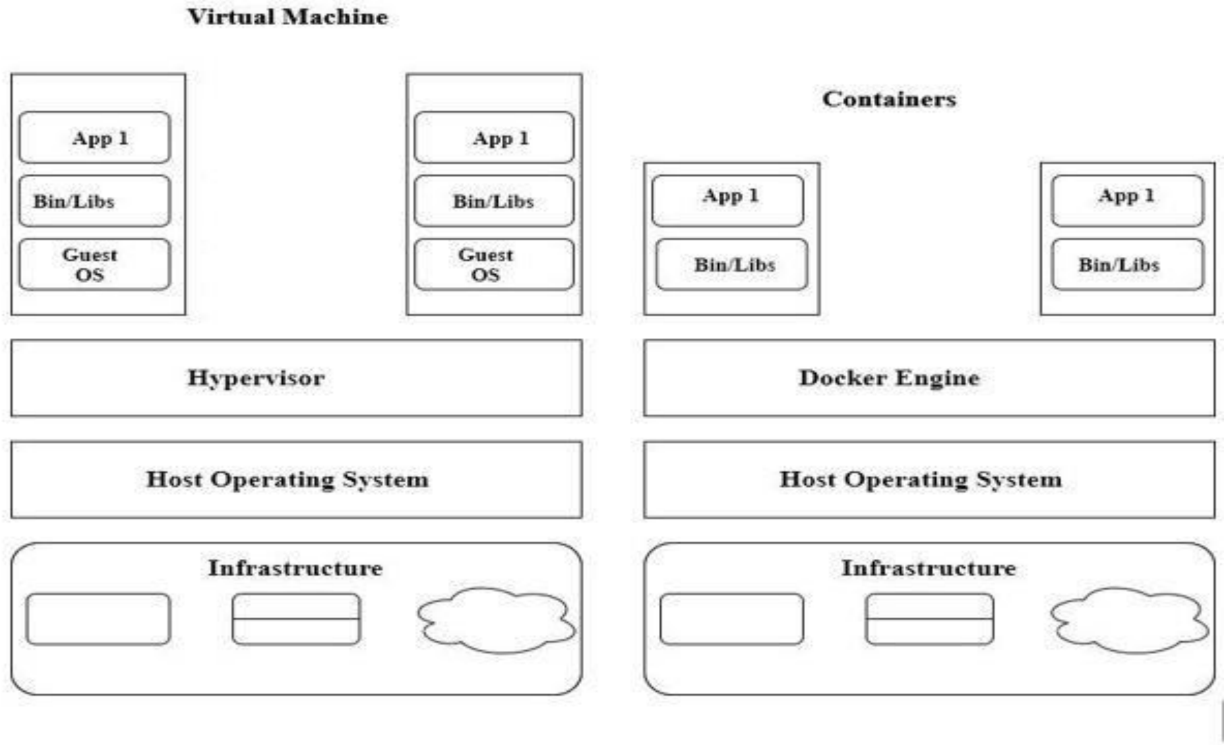

Figure1: Virtual Machine Application Vs Container

Since safety is a big issue, a number of attacks may occur in Docker such as Denial of Service (DOS), SQL injection, privilege escalation, poisoned images, etc. Among all these attacks DOS (Denial of Service) and DDOS (Distributed Denial of Service) are becoming so frequent as they can be easily launched due to the availability of multiple attacking tools, resulting in enormous financial losses for the organizations.

A Increases in the demand for Linux containers could be seen in recent years. Docker containers provided the benefits and became popular day by day [2]. The docker Container Advantages are listed below:-

i) Speed

The time it takes for a container to be developed is extremely fast testing, creation, and easy installation as the containers is lightweight. Containers will be delivered for testing after they are made. Flexibility inside docker containers applications are highly portable. The performance of the portable applications remains the same and easily moved to a single element.

ii) Scalability

Docker can run on every Linux machine has the capability that it can be expand in some substantial servers, cloud platforms and data servers,. Containers can freely move from a cloud domain to local host.

iii) Rapid Delivery

The authority of the controller is to maintain and deploy the server with containers, Containers can work in each domain.

iv) Density

Containers can be run on a particular Kernel host compared to VMs and because of the higher the overall work density of the Docker Containers is higher.

DISADVANTAGE OF DOCKER CONTAINER

- Docker may not run on previous devices (only supports 64bit machines)[4].
- Docker not provided complete virtualization and depends on the Linux kernel
- Security issue in Docker container [3].

Malware Attacks IN DOCKER BASED ENVIRONMENT

In a very Simple term, any computer program, Software or tool intentionally designed to harm your computer, personal data, hardware or network comes under the category of malwares. Malwares does not start its working as soon as they introduced in a target machine until and unless they are explicitly converts into some executable binary code, running scripts, embedded with some runnable software or become active. This type of files can be known by plenty of terms like SQL injection ransom ware, Computer Viruses, Worms, Adware, Trojan Horses and Scare ware.

Malwares runs in a targeted computer without the consent of computer user and tries to harm it in anyway but contrary those software which cause unintentional loss to the user because of some flaw, doesn`t come under Malware category and can be called a Software Bug. Many a times, companies intentionally provide software or program which secretly work without the knowledge of users. The only way we can keep our computers safe by Malwares is by avert malware programs to gain access to our computers. We can use antiviruses, firewalls and other strategies to defend our system against the Malwares.

TYPES OF MALWARES

Now we will discuss different categories of malwares but these categories are not 100% mutually exclusive because a single malware can be used in different types to harm the user.

a) Virus

It is a software generally embedded in some another apparently harm less program that can spreading by itself and intrude into another running processes or files and performs some undesirable actions like disrupting normal execution of other processes or deleting confidential data.

b) Screen locking ransom ware

The attacks of these categories can be seen frequently. It is a common scenario that a user opens its system and found itself locked out of its own computer system. Just a threatening message is displayed on computer screen somewhat like your data has been encrypted or your PC has been locked and you cannot recover it by any means but only by paying a fine money. Displayed message also contains information about how to pay fine money and a procedure to get a decryption key or password. If you do not pay the amount within a deadline then sometimes they actually delete all your data permanently and leave you with just a dumb box.

c) Trojan horses

These are the harmful programs that mostly take a form of some benign program or utility so that they can easily invade into a target computer. They usually have some hidden destructive task which they execute when activated with help of some other process. These programs need some special social engineering to be spread. These cannot be easily detected because they do not leave any footprints behind but definitely affect the performance of the computer system, mostly slow down it. Major difference between Trojan horses and computer viruses are that they didn‘t try to intrude themselves into another programs and files.

d) Root kit

One of the most important properties to be possessed by the malwares is that after being installed in the target system, their existence and their working should be must hide by the user. They should not be easily detectable. Malicious software‘s like Root kit allows this privacy to the malicious programs by performing slight modification in the system`s operating system so that they cannot be detected. Generally Root kit impose read protections or you can say view protection to the malicious process so that it is not visible in the process list and cannot be read by the user anyway. Because of Root kits, malicious programs get complete control over the system means that already installed software working can be changed which might include those software which are used to stop malicious programs.

e) Backdoors

As the name suggests these are used to bypass the front security to enter into the confidential area or to enter into a private network and viewing their files by bypassing normal authentication check procedures. Once a single backdoor enters into a system, it can make a path for other backdoors to be installed in the future unknowingly to user.

f) SQL injection

SQL injection attack where the malicious script was inserted into strings which later brought to the processing or implementation of database backend. Technology development is increasing day by day, while the network security issues are also increasing due to these security issues that no web application can be safe in this era, most attackers focus on the web application to attack [1].

LITERATURE REVIEW

Research work on Docker’s security.

Scheepers [2] did a comparison between Linux Containers (LXC) and XEN and provided the advantages or disadvantages for both. It was concluded that XEN is better when it comes to equal distribution of resources whereas LXC is better in terms of resource utilization as it wastes fewer resources but the time taken to complete the tasks is more in the case of LXC.

Rad et al. [4] provided the pros and cons of Docker along with a short description of the four constituents of Docker namely containers, images, Docker client and server, and Docker registries. A comparison of Docker with other virtualization techniques like XEN, LXCs, and KVM has also been made.

Sultan et al. [3] made a comparison between containers and VMs. A threat model having four use cases was also proposed for the containers. Of the four, only one use case takes advantage of hardware-based solution while the other three use cases have software-based solution. Within software-based solutions, Linux kernel features and Linux Security Modules is addressed while Intel SGX and virtual Trusted Framework Modules are addressed within hardware dependent solutions. Finally, several open questions are raised and the possibilities for future research.

Martin et al. [5] presented and described three typical Docker usage cases, namely widespread use case, recommended use case, and use case for CaaaS cloud providers. An adversary model was defined for the three use 7 cases, and a vulnerability analysis was also conducted. In addition, an overview of the Docker ecosystem was provided along with a brief description of all its constituents such as the Docker hub, the Docker daemon, etc. compared KVM and Docker and found out that if we talk about the resources getting waste then there is no major difference between these two as they both waste resources equally.

Seo et al. [6] also made a comparison bet in terms of performance and came to a conclusion that Docker has a better performance than KVM.

Combe et al. [7] gave a brief overview of the docker Security or Docker environment including a list of the risks associated with docker use. In addition, the challenges faced during the use of docker were identified and categorized into three groups, namely usage challenges for PaaS providers, widespread usage challenges, and suggested usage challenges.

Andreou et al. [8] provided Docker security architecture with the description of their components and overview of Docker-security .They also present the two mechanisms for the security of Docker container components which is protected with App Armor Further, discussed the two use-cases.

Bui [9] made a comparison and provided a description of the two main virtualization techniques namely container-based virtualization and hypervisor-based virtualization. Docker security level was analyzed on the basis of two areas i.e. security at the internal level and interaction with the safety attributes of the Linux kernel. Different constituents of Docker such as containers, Docker hub, and Docker engine were also described under Docker overview.

Yasrab [10] presented the overview and security analysis of Docker along with a brief comparison between virtual machines and Docker. A solution has been proposed for a safe Docker environment under which deployment guidelines and policy modules for access control were described. Lastly, a description of the attacks that can take place in Docker and how to tackle these attacks has also been given.

Abed et al. [11] described system overview using Linux kernel. They also discussed the system call bag BOSC (Bag of system call) approach which is anomaly detection technique based on frequency. They also discussed malicious attack using SQL map on the container to create malicious dataset by targeting the container-organized Mysql database, and the author also discussed Linux strace to trace all system calls.

Tien et al. [1] presents a method known as KubAnomaly which is a cloud container orchestration framework that can be used to detect anomalies in Kubernetes. System logs were monitored to suggest an extraction method for features. To evaluate the efficiency of the proposed program, three separate datasets (publicly accessible, private, and data from the experiments performed in the real-world) were used. The potency of the model that has been proposed was illustrated by making a comparison between its accuracy and accuracy of other 8 machine learning algorithms and the accuracy was found to be 96%. The proposed model successfully identified four real attacks that were launched by attackers in September 2018.

Security Solution In Docker Environment

Different researcher has purpose distinguished solution for Dockers environment, some of them are described below.

Run Containers as a Non-Root User

Run containers process as a non-root user for other, because all process in container use root permission by default.

Use Your Own Private Registry

A registry is a fully independent catalog of container images which is set by the organization and used it. You can host it on your own on-premises infrastructure or on a third-party registry service such as Amazon ECR, Azure Container Registry, Google Container Registry, Red Hat Quay and JFrog's own container registry service

Keep Your Images Lean and Clean

If your dockers image is larger than attack surface is also large. So surface should according to need a simple as possible. In the case of a fully-fledged VM, you have no choice but to use an entire operating system. But with Docker workloads, your containers only have to provide the resources your application needs.

CONCLUSION and scope of future work

The use of virtualization technologies has increases adequately after a long time and Docker is a comparatively new and the most presiding applicant since it comes with new competencies that previous techniques did no longer own. Flexibility, speed, thickness and rapid transmission are the advantages of Docker but there exist some drawbacks too. The disadvantage of Docker involve no complete virtualization, its supports only 64bit machines or no longer being able to run on older machines, its security and privacy of users is the most critical concern. The major usage of computer systems and the rising dependency of individuals, trade, commerce, governments on them make it more prone to attacks. Therefore, we focused on network detection for container security.

Machine Learning based Detection techniques for Securing Docker Container has been proposed for detection in Docker-based environment. Earlier approaches were not giving efficient results due to lack of ability to remember long sequences of system call. In addition for extensive or limitless traces, the approach cannot work accurately. In this Report, a threat model for all the possible attacks that can take place in a Docker-based environment. Attack classification has also been proposed which discusses all the major attacks that can take place on four layers of Docker i.e. container, application, Docker engine, and host. An architectural framework has been presented which describes the entire process of attack launched.

In future, we will going to enlarge the threat model for multiple host and will also propose more efficient technique for the Docker container, to find the pattern more accurately in the system calls log in order to detect the malware in Docker environment.

References

1. Parvez, A. Rahmati, I. Guvenc, A. I. Sarwat and H. Dai, "A Survey on Low Latency Towards 5G: RAN, Core Network and Caching Solutions," in IEEE Communications Surveys & Tutorials, vol. 20, no. 4, pp. 3098-3130, Fourthquarter 2018, doi: 10.1109/COMST.2018.2841349.
2. C. W. Tien, T. Y. Huang, C. W. Tien, T. C. Huang, and S. Y. Kuo, ―Kub, "Anomaly: Anomaly detection for the Docker orchestration platform with neural network approaches",*Engineering Reports*, vol. 1, no. 5, 2019.
3. M. J. Scheepers, "Virtualization and Containerization of Application Infrastructure: A Comparison", 2014.
4. S. Sultan, I. Ahmad, T. Dimitriou, ‖Container Security: Issues Challenges and the Road Ahead‖, IEEE Access, vol. 7, pp. 52976-52996, 2019.
5. B. B. Rad, H. J. Bhatti, M. Ahmadi, "An Introduction to Docker and Analysis of its Performance", IJCSNS International Journal of Computer Science and Network Security, vol. 17, no. 3, March 2017.
6. A. Martin, S. Raponi, T. Combe, R. Di Pietro, "Docker ecosystem vulnerability analysis", Computer Communications, vol. 122, pp. 30-43, 2018.
7. K.-T. Seo, H.-S. Hwang, I.-Y. Moon, O.-Y. Kwon, B.-J. Kim, "Performance comparison analysis of linux container and virtual machine for building cloud", Advanced Science and Technology Letters, vol. 66, no. 105111, pp. 2, 2014.
8. T.Combe, A.Martin, R.DiPietro,"To docker or not to docker: Asecurity perspective", IEEE Cloud Comput., vol. 3, no. 5, pp. 54-62, Sep./Oct. 2016.
9. Loukidis-Andreou, I. Giannakopoulos, K. Doka, and N. Koziris, "Docker-Sec: A Fully Automated Container Security Enhancement Mechanism," 2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS), 2018.
10. T. Bui,"Analysis of Docker Security" CoRR, vol. abs/1501.02967, 2015. [Online]. Available: http://arxiv.org/abs/1501.02967
11. R. Yasrab and I. Technology,"Mitigating Docker Security Issues".
12. A. S. Abed, T. C. Clancy, and D. S. Levy, "Applying Bag of System Calls for Anomalous Behavior Detection of Applications in Linux Containers" 2015 IEEE Globecom Workshops (GC Wkshps), 2015.
13. Mohd, N., Singh, A., & Bhadauria, H. S. (2021). Intrusion Detection System Based on Hybrid Hierarchical Classifiers. *Wireless Personal Communications*, *121*(1), 659-686.