Container and container orchestration security

Joakim Croona - jcroona@kth.se
Philip Strömberg - phistr@kth.se

April 2019

1 Introduction

Containerization is a fairly new technology, that has grown in popularity since Docker was released. The goals of containers is to isolate programs and dependencies from the underlying host, with lower overhead than virtual machines. As containers have grown in popularity the importance of container security has grown, as more organizations will be affected by issues in containers.

Important questions to ask for organizations that use containers are:

- What are the security issues that arise when using containers,
- what can done to secure containers
- What can be done to enhance container security

1.1 What is a container?

A container is a small piece of standalone software. The point of it is to be able to execute in the same way on any hardware, infrastructure and operating system. This gives containers powerful flexibility, as both the developer and the end user will not have to consider each others setup to use the software. It enables an agile transformation that synergizes with microservices, as it is both small in size and fast in execution.

Containers can in many ways be compared to virtual machines, but the key difference is that while virtual machines simulates hardware, containers simulate the operating system. This makes containers the more lightweight and agile alternative.

The primary container runtime that is used is Docker[1], which is developed by Docker inc. They also provide a service called Docker Hub[2], from which it is possible to download predefined images and run them as containers. Anyone can create and upload their own docker image to Docker Hub. This simplifies the use of containers, as there are predefined images that contain commonly used services and applications for development, such as databases and specific language runtimes. Many of those images are provided by reputable companies as well, and they are verified and clearly marked as official images.

Using docker Hub, creating an instance of mysql on a development machine is just a command away. Containers can be created that only contain the dependencies that is needed for a given application, and the containers that are used for development of the application can be the same as the containers used in production. Containers therefore creates a consistent runtime environment where dependencies are explicitly specified, and that should make arguments like "Works on my machine" irrelevant, as the same environment is used when developing and deploying. Containers can therefore solve certain problems that can be faced when developing software, but with a lower performance penalty than competing solutions like virtual machines.

1.2 Container orchestration

As containers usage has grown, the need to automate deployment and management of the has become apparent. To manage the creation, deployment and destruction of containers at scale container orchestration software has been developed. Using container orchestration software it is possible to automatically scale up container deployments to handle increased customer demand. The use of container orchestration software becomes important as the use of containers increases in an organization, as it might be difficult to manage 4000 containers manually.

Currently there are two major container orchestration software that is used:

- Kubernetes
- Docker Swarm

1.2.1 Kubernetes

Kubernetes is an open source container orchestration framework, that was developed by Google, and is inspired by the Borg system that Google uses internally, that has not been open sourced. It has been described as the Gold standard for container orchestration[3], and is now being developed under the umbrella of the Cloud Native Computing Foundation. The CNCF is backed by many large technology companies that has an interest in containers, such as Google, Red hat, and Intel. Kubernetes is also supported by major cloud vendors, such as Amazon web services, Google cloud platform, and Microsoft Azure all provide managed Kubernetes services.

1.2.2 Docker Swarm

Docker Swarm is developed by Docker, inc, that also develops Docker. The service is a part of the docker product, so if docker is installed, Docker Swarm is also installed. Docker Swarm has been described as the easier alternative to get started with compared to Kubernetes, as it is less configurable and less complex, compared to Kubernetes[3][4]. In certain environments, having a less complex tool, that requires less configuration, can be good for beginners and

smaller scale businesses. This, however, does make Docker Swarm less flexible, which might make it unsuitable for certain use cases.

2 What is a container oriented attack?

A container only has access to read, write and execute within its container. That is at least how it is meant to be during normal circumstances, however, there have been several known exploits and other instances of using containers in malicious ways.

2.1 Escape attacks

Normally a container is, as expected, contained. It has no access or even concept of there being anything outside of its vision, but with the help of various exploits it is possible to break out and attack its host system. When a container breaks out of its shell with malicious intents, it is called a container escape attack. This can, for example, be done by social engineering, such as the exploit of the vulnerability (CVE-2019-5736), tricking people into executing malicious containers [5].

2.2 Side-channel attacks

There are many more types of attacks, but the other common, relevant, and just as dangerous one is the side-channel attack. This type of attack can read from containers, exposing their contents. This is not necessarily the fault of the container or its implementation, but the fault of its context. The operating system or even hardware can give off the contents by revealing small bits of information unknowingly, giving the attacker enough at least hints of information of what the container could hold.

2.3 Docker hub attacks

Anyone can upload their own container image to docker. This is usually a good thing, but as not everyone is equally interested in the advancement of society, that is not always the case. Container breaches can be very hard to detect, with 43% of IT security professionals [6] saying that for them it can take at least days to detect a compromised container.

This threat has become a reality several times with malicious container images still on the platform. A notably large detection happened last year in June, where 17 images with a combined download number of at least 5 million was discovered [7]. The company uncovering the malicious images has stated that security is not in DockerHub's focus [7] and that *"For ordinary users, just pulling a Docker image from the DockerHub is like pulling arbitrary binary data from somewhere, executing it, and hoping for the best without really knowing what's in it"*. This is not a good review of a software distribution platform,

especially as Docker provided a paid service that scans images for security risks, which might affect their incentive to fix the real issues that was found.

3 History of security breaches

Containers have a rather bad reputation when it comes to security. Its history is rich in breaches and exploits, and according to a Tripwire survey from January this year [6], 94% of IT security professionals are concerned, with almost half of them being "very concerned", with security in container environments in general, and the vast majority thinking that security issues will increase in the coming year. As many as 60% of them also had some kind of security breach revolving containers during the year as well. That number is even larger when it comes to companies with more than 100 containers, rising to 70%.

4 Measures for making containers more secure

Escape attacks can be used against virtualization based isolation, and against containers, but containers have more angles to attack [8]. When virtualization is used for separation, the tools used to isolate virtual machines running on a shared host are the hypervisor, and hardware isolation methods. Those methods are not perfect, as there have been escape attacks against virtual machines, but such technology has been used for quite some time now. The big issue with virtualization is that it has more overhead than containers. Containers, on the other hand are secured using the Linux kernel, and a mandatory access control system, like SELinux. This has a bigger attack surface, as issues in the Linux kernel and in SELinux or some other mandatory access control system can be used to break out of the container [8].

The concept has major flaws, however, methods to mitigate those risks have been presented. One can use a container co-location strategy to reduce the security risks in security demanding environments. If the containers that are running on a single host do not need to be isolated from each other, then the risks of escape attacks are limited [8]. Kubernetes has tools to specify on what node a given pod runs, which makes it possible to specify such requirements. Using pod co-location strategies to enhance security is therefore possible. Containers do increase the security risks somewhat, but there are methods to mitigate those risks to some extent, but that depends on what the security requirements are in a given environment. Those requirements can differ between public and private organizations, and between projects in the same organization. The most secure thing to do is to have a 1:1 relation between pods and containers, giving an escaped container less resources and more resistance before breaching completely [8]. This could also leave a lot of resources unused however, as the pod loses the flexibility of swapping in and out containers.

5 The effect of best practices

Containers are a rather new area in computer science and that shows in its best practices and the awareness of its users. According to a survey by Tripwire [6] directed at IT security professionals, 47% of them says that their containers are vulnerable, 46% of them do not even know whether or not they have vulnerable containers, leaving only 7% confident in their active containers. Some best practices exists however, such as the organization developing Kubernetes and container standards, Cloud Native Computing Foundation with their recent article from the beginning of this year explaining 9 ways to improve the security of your Kubernetes systems together with some general tips and good advice [9]. Some of the advice in that article is obvious, like making sure that you are running the latest, patched Kubernetes version, but most other advice is to enable certain features, and using namespaces to separate sensitive workloads.

6 The competitor: Virtual machines

6.1 Why use virtual machines instead?

Containers are more flexible, more light weight and easier to use than virtual machines, so what are the downsides? While both virtual machines and containers are possible targets for escape attacks, as they have very similar properties, containers are far more vulnerable [8]. While virtualization is secured by both hardware and the hypervisor, the only thing standing between you and a container is the container runtime and the Linux kernel.

6.2 Past security breaches in virtual machines

There has been relatively few known breaches in virtual machines [8], but in the hacking contest Pwn2Own there have been several breaches detected accompanied with or during the competition. Notable examples are the 2017 [10] and 2019 [11], both years managing to successfully complete an escape attack. The 2019 Fluoroacetate team was especially successful, managing to hack both Oracle's VirtualBox and VMWare workstation[11], two of the biggest virtual machine systems. As the competition lasted for several days, it is also notable that the team managed to breach both products on the very first day of the contest, winning them several hundred thousand dollars.

7 Conclusion

As we have seen in this essay, containers are very flexible and can be used in most contexts. They have many advantages over their competitor, virtual machines, by having less overhead, giving them faster execution times and take up less space on hardware. On the security side there are many issues with containers however. They are a lot more susceptible to attacks compared to virtual

machines due to their nature and the majority of its big users are experiencing security breaches and general suspicion towards the the concept as a whole. Its users also don't seem very positive about the future of containers, but as they have so many advantages and potential of improving, the future of containers is anything but dark. While it might take a couple of innovations in the area to make more serious organizations like the police [8] or banks to start using them, there will always be a constant stream of smaller or less serious tech companies standing in line to partake in the fastest and most efficient alternative possible.

References

[1] *Docker*. https://www.docker.com. Accessed: 2019-04-30.

[2] *Docker Hub*. https://hub.docker.com. Accessed: 2019-04-30.

[3] *New relic What Is Container Orchestration?* https://blog.newrelic.com/engineering/container-orchestration-explained/. Accessed: 2019-04-30.

[4] *The New Stack Kubernetes vs. Docker Swarm: What's the Difference?* https://thenewstack.io/kubernetes-vs-docker-swarm-whats-the-difference/. Accessed: 2019-04-30.

[5] *Threat Post Major Container Security Flaw Threatens Cascading Attacks*. https://threatpost.com/container-security-flaw-runc/141737/. Accessed: 2019-04-30.

[6] *Tripwire State of Container Security Report*. https://3b6xlt3iddqmuq5vy2w0s5d3-wpengine.netdna-ssl.com/state-of-security/wp-content/uploads/sites/3/Tripwire-Dimensional-Research-State-of-Container-Security-Report.pdf. Accessed: 2019-04-30.

[7] *The register UK Docker Hub security dissed, dodgy container image data damned*. https://www.theregister.co.uk/2018/06/14/docker_security_dodgy_container_images/. Accessed: 2019-04-30.

[8] Christian Abdelmassih. *Container Orchestration in Security Demanding Environments at the Swedish Police Authority*. 2018.

[9] *CNCF 9 Kubernetes Security Best Practices Everyone Must Follow*. https://www.cncf.io/blog/2019/01/14/9-kubernetes-security-best-practices-everyone-must-follow/. Accessed: 2019-04-30.

[10] *VMWare The Security Landscape: Pwn2Own 2017*. https://blogs.vmware.com/security/2017/03/security-landscape-pwn2own-2017.html. Accessed: 2019-04-30.

[11] *BleepingComputer Safari, Virtualbox, VMware Get Hacked During First Day of Pwn2Own 2019*. https://www.bleepingcomputer.com/news/security/safari-virtualbox-vmware-get-hacked-during-first-day-of-pwn2own-2019/. Accessed: 2019-04-30.