UNIVERSITÄT PASSAU

Docker Security

To Docker or Not to Docker - A Security Perspective

Felix Klement

System Security - 5622V

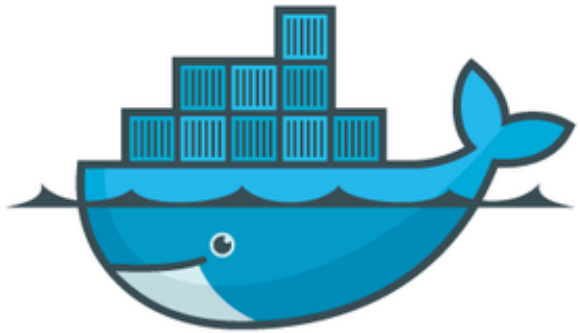
Chair of IT-Security
University of Passau, Germany

08.06.2020

UNIVERSITÄT PASSAU

What is Docker?

UNIVERSITÄT PASSAU



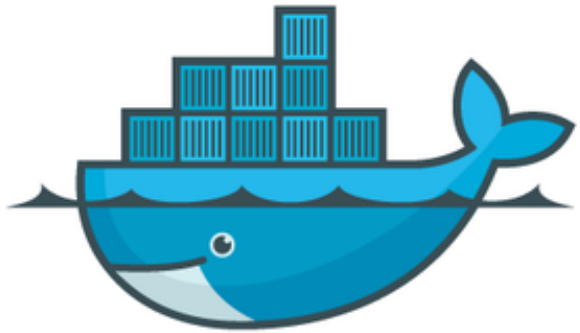
Movement in the cloud

Everyone wants to

- Migrate workloads to cloud
- Be portable across multiple environments
- Avoid cloud vendor lock-in

What is Docker?

UNIVERSITÄT PASSAU

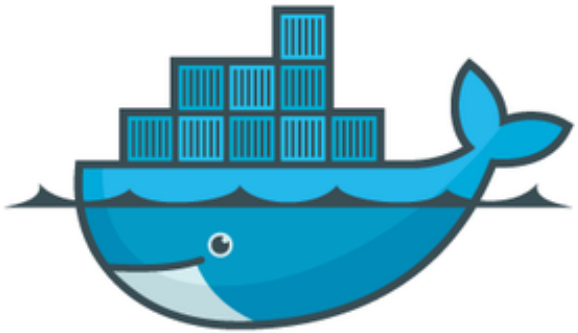


Movement in the cloud

Everyone wants to

- Migrate workloads to cloud
- Be portable across multiple environments
- Avoid cloud vendor lock-in

What is Docker?

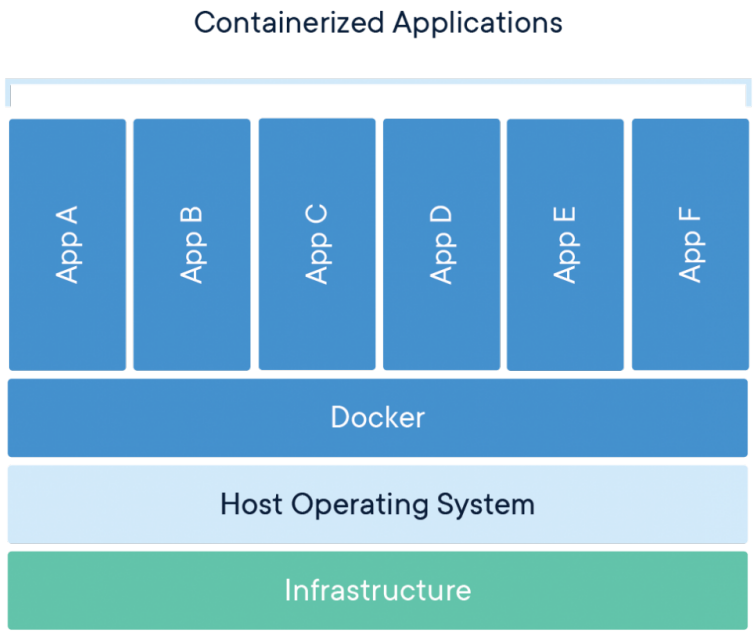


Movement in the cloud

Everyone wants to

- Migrate workloads to cloud
- Be portable across multiple environments
- Avoid cloud vendor lock-in

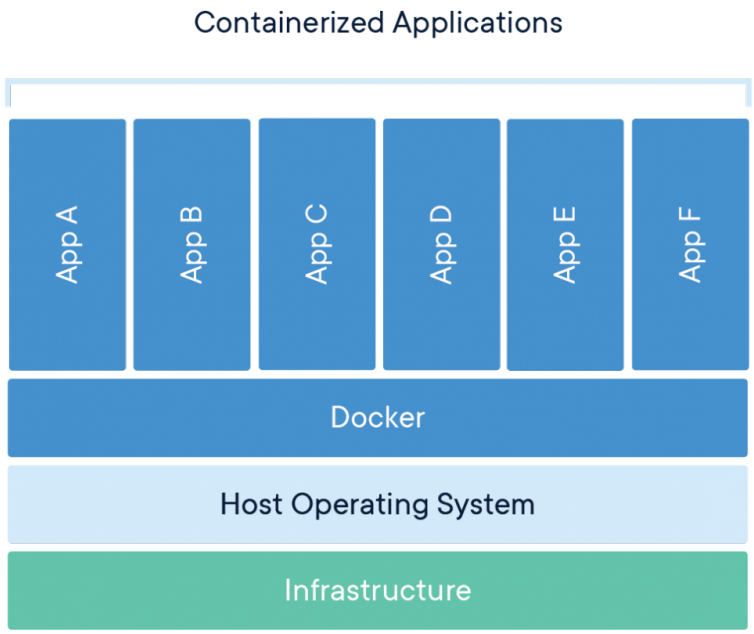
What is Docker?



What is a container?

- Standardized packaging for software and dependencies
- Isolate apps from each other
- Share the same OS kernel
- Works with all major Linux and Windows sServer

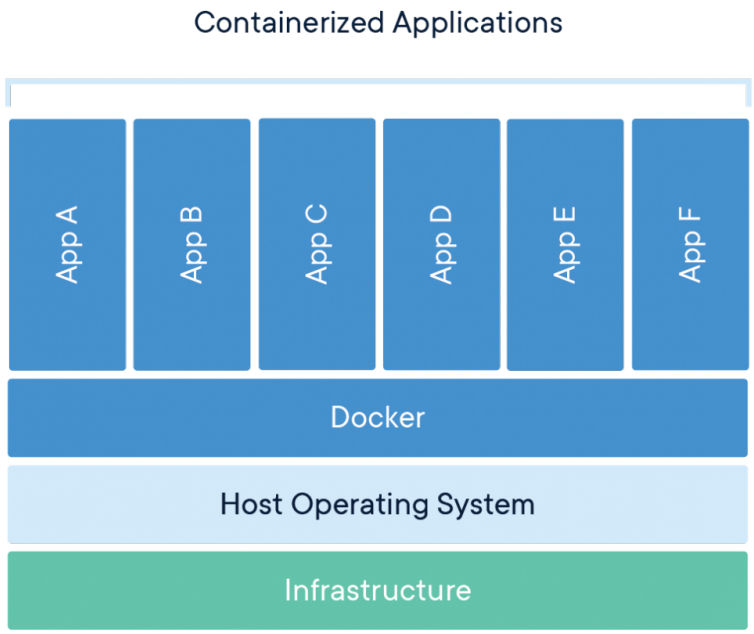
What is Docker?



What is a container?

- Standardized packaging for software and dependencies
- Isolate apps from each other
- Share the same OS kernel
- Works with all major Linux and Windows sServer

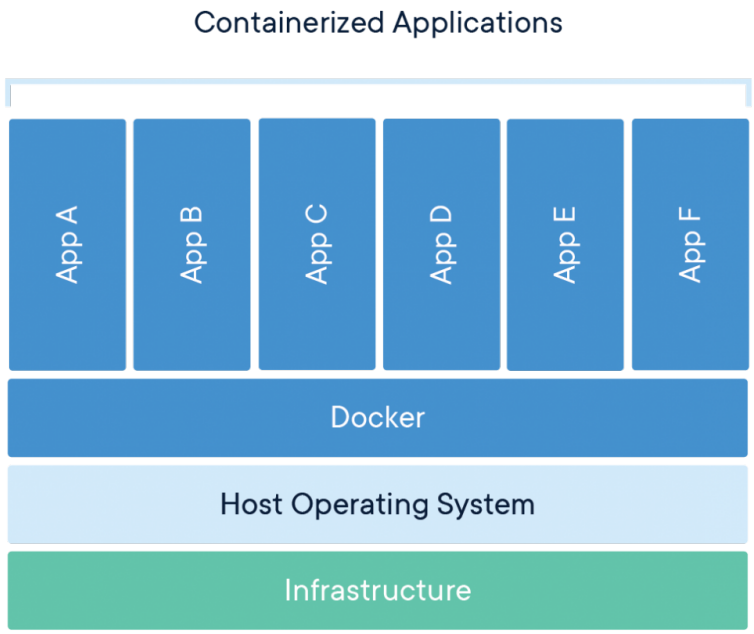
What is Docker?



What is a container?

- Standardized packaging for software and dependencies
- Isolate apps from each other
- Share the same OS kernel
- Works with all major Linux and Windows sServer

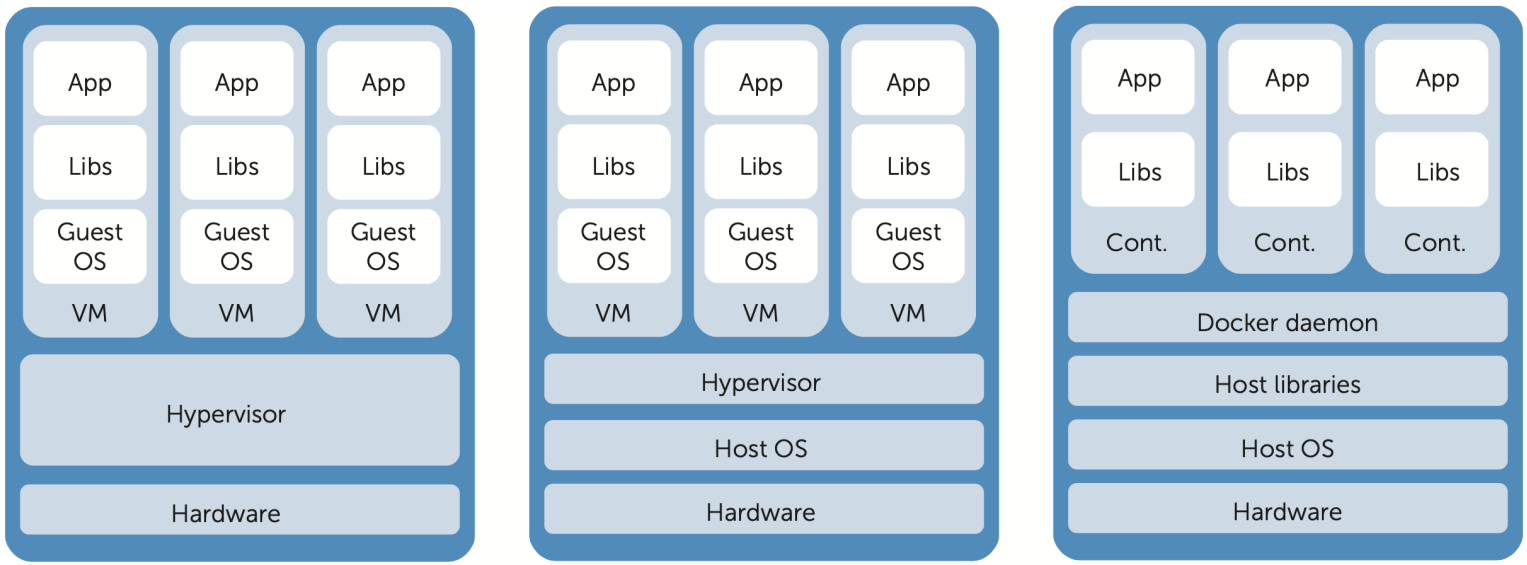
What is Docker?



What is a container?

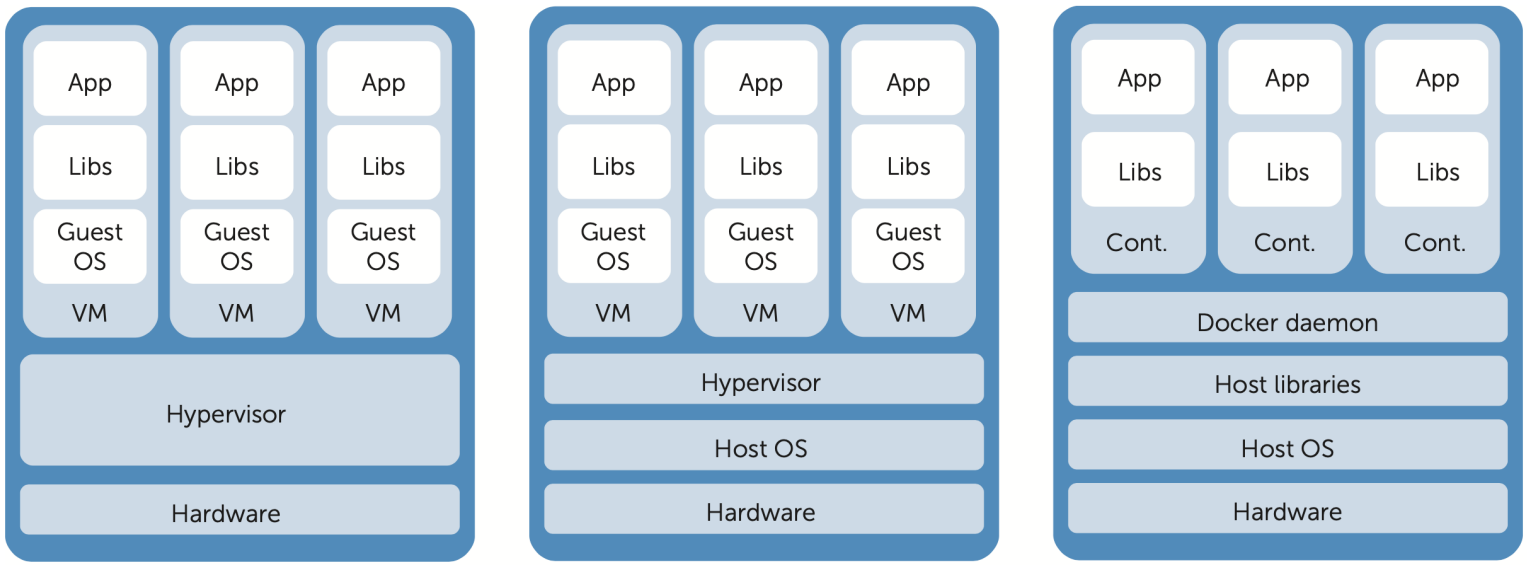
- Standardized packaging for software and dependencies
- Isolate apps from each other
- Share the same OS kernel
- Works with all major Linux and Windows sServer

Comparing application runtime models



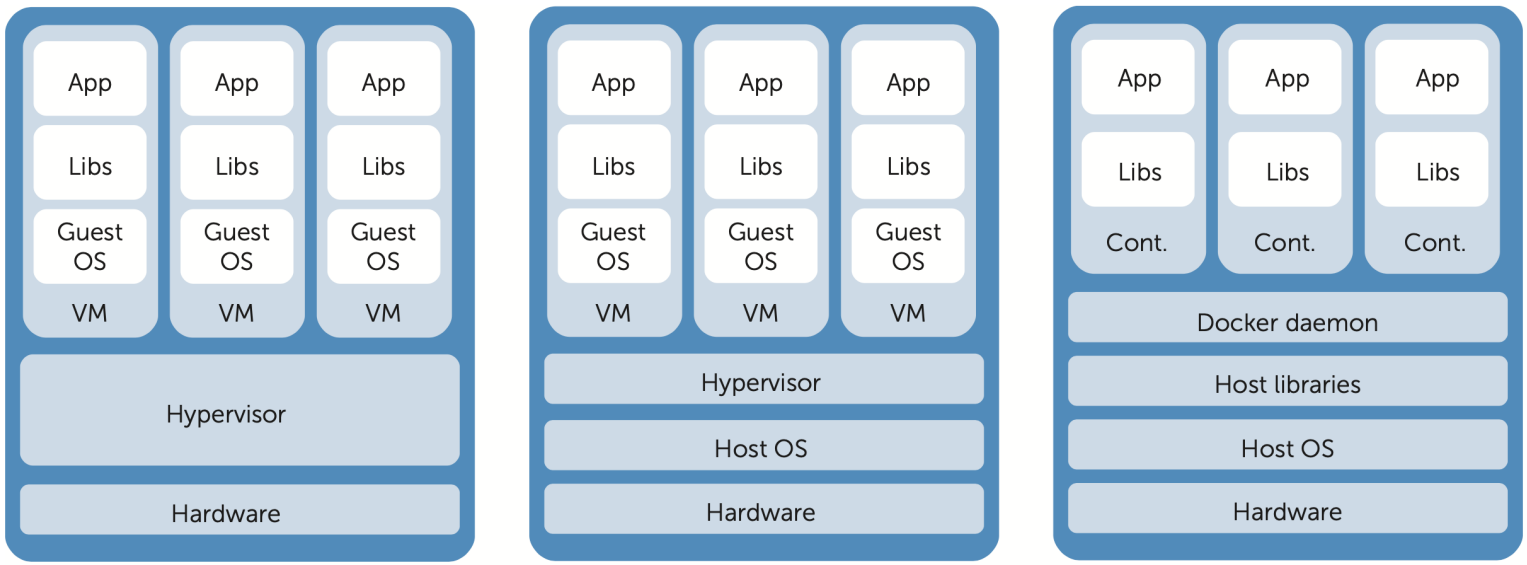
1. Type 1 hypervisor
2. Type 2 hypervisor
3. Container

Comparing application runtime models



1. Type 1 hypervisor
2. Type 2 hypervisor
3. Container

Comparing application runtime models



1. Type 1 hypervisor
2. Type 2 hypervisor
3. Container

UNIVERSITÄT PASSAU

UNIVERSITÄT PASSAU

Security relies on three factors:

- Isolation of processes at the userspace level (managed by Docker deamon)
- Enforcement of isolation by the kernel
- Network operations security

UNIVERSITÄT PASSAU

Security relies on three factors:

- Isolation of processes at the userspace level (managed by Docker deamon)
- Enforcement of isolation by the kernel
- Network operations security

UNIVERSITÄT PASSAU

Security relies on three factors:

- Isolation of processes at the userspace level (managed by Docker deamon)
- Enforcement of isolation by the kernel
- Network operations security

Adversary models according to Combe et al.

UNIVERSITÄT PASSAU

Direct adversaries

- Can sniff, block, inject or modify network and system communications
- They directly target the production machines
- Can compromise several system components (locally or remotely)

Indirect adversaries

- Same capabilities as direct adversaries
- Leverage the Docker ecosystem

Adversary models according to Combe et al.

UNIVERSITÄT PASSAU

Direct adversaries

- Can sniff, block, inject or modify network and system communications
- They directly target the production machines
- Can compromise several system components (locally or remotely)

Indirect adversaries

- Same capabilities as direct adversaries
- Leverage the Docker ecosystem

Adversary models according to Combe et al.

UNIVERSITÄT PASSAU

Direct adversaries

- Can sniff, block, inject or modify network and system communications
- They directly target the production machines
- Can compromise several system components (locally or remotely)

Indirect adversaries

- Same capabilities as direct adversaries
- Leverage the Docker ecosystem

Adversary models according to Combe et al.

UNIVERSITÄT PASSAU

Direct adversaries

- Can sniff, block, inject or modify network and system communications
- They directly target the production machines
- Can compromise several system components (locally or remotely)

Indirect adversaries

- Same capabilities as direct adversaries
- Leverage the Docker ecosystem

Adversary models according to Combe et al.

UNIVERSITÄT PASSAU

Direct adversaries

- Can sniff, block, inject or modify network and system communications
- They directly target the production machines
- Can compromise several system components (locally or remotely)

Indirect adversaries

- Same capabilities as direct adversaries
- Leverage the Docker ecosystem

Targets according to Combe et al.

UNIVERSITÄT PASSAU

- Containers
- Code repositories
- Image repositories
- Management network

Targets according to Combe et al.

UNIVERSITÄT PASSAU

- Containers
- Code repositories
- Image repositories
- Management network

Targets according to Combe et al.

UNIVERSITÄT PASSAU

- Containers
- Code repositories
- Image repositories
- Management network

Targets according to Combe et al.

UNIVERSITÄT PASSAU

- Containers
- Code repositories
- Image repositories
- Management network

UNIVERSITÄT PASSAU

1. Introduction

2. Security Overview

3. Vulnerabilities

4. Conclusion

UNIVERSITÄT PASSAU

Insecure local configuration

- Default configuration on local systems is relatively secure
- Examples:
  - *-net=host*
    Full access to host network stack (enabling network sniffing, reconfiguration, and so on)
  - *-cap-add=¡CAP¿*
    Specified capabilities e.g. with *SYS_ADMIN*, a container can remount */proc* and */sys* subdirectories and change the host's kernel parameters

UNIVERSITÄT PASSAU

Insecure local configuration

- Default configuration on local systems is relatively secure
- Examples:
  - *-net=host*
    Full access to host network stack (enabling network sniffing, reconfiguration, and so on)
  - *-cap-add=¡CAP¿*
    Specified capabilities e.g. with *SYS_ADMIN*, a container can remount */proc* and */sys* subdirectories and change the host's kernel parameters

UNIVERSITÄT PASSAU

Insecure local configuration

- Default configuration on local systems is relatively secure
- Examples:
  - *-net=host*
    Full access to host network stack (enabling network sniffing, reconfiguration, and so on)
  - *-cap-add=¡CAP¿*
    Specified capabilities e.g. with *SYS_ADMIN*, a container can remount */proc* and */sys* subdirectories and change the host's kernel parameters

UNIVERSITÄT PASSAU

Other possible vulnerabilities are:

- Image distribution vulnerabilities
- Weak local access control

UNIVERSITÄT PASSAU

Other possible vulnerabilities are:

- Image distribution vulnerabilities
- Weak local access control

UNIVERSITÄT PASSAU

1. Introduction

2. Security Overview

3. Vulnerabilities

4. Conclusion

Conclusion

UNIVERSITÄT PASSAU

- Briefly highlighted Docker in general
- We had a overview at the Docker Security including the adversary models and possible targets
- Shortly investigated the possible vulnerabilities

Contact: fk@sec.uni-passau.de

Conclusion

UNIVERSITÄT PASSAU

- Briefly highlighted Docker in general
- We had a overview at the Docker Security including the adversary models and possible targets
- Shortly investigated the possible vulnerabilities

Contact: fk@sec.uni-passau.de

Conclusion

UNIVERSITÄT PASSAU

- Briefly highlighted Docker in general
- We had a overview at the Docker Security including the adversary models and possible targets
- Shortly investigated the possible vulnerabilities

Contact: fk@sec.uni-passau.de

Conclusion

UNIVERSITÄT PASSAU

- Briefly highlighted Docker in general
- We had a overview at the Docker Security including the adversary models and possible targets
- Shortly investigated the possible vulnerabilities

Contact: fk@sec.uni-passau.de