ISX QII/21

IT-Security Virtual Conference

9. Juni

HERZLICH WILLKOMMEN

isxconference.de | #isxconference

VOGEL IT AKADEMIE

carmasec

security. done. right.

How to: Container Security

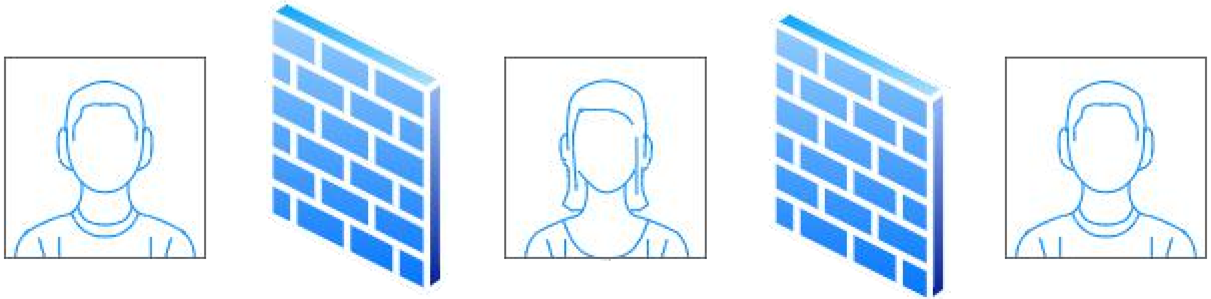
You can do more than image scanning

whoami

Kevin Kloft

- E-Mail: kevin.kloft@carmasec.com
- Senior Security Solution Architect
- carmasec GmbH & Co. KG since 2019
- twitter: @kevsecops

Shift Left

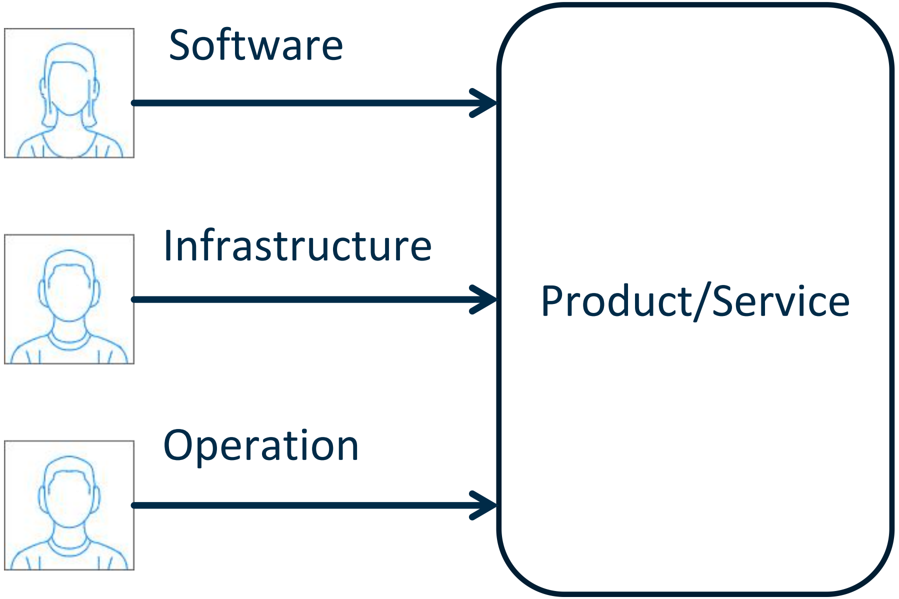


Developer

Release Engineer

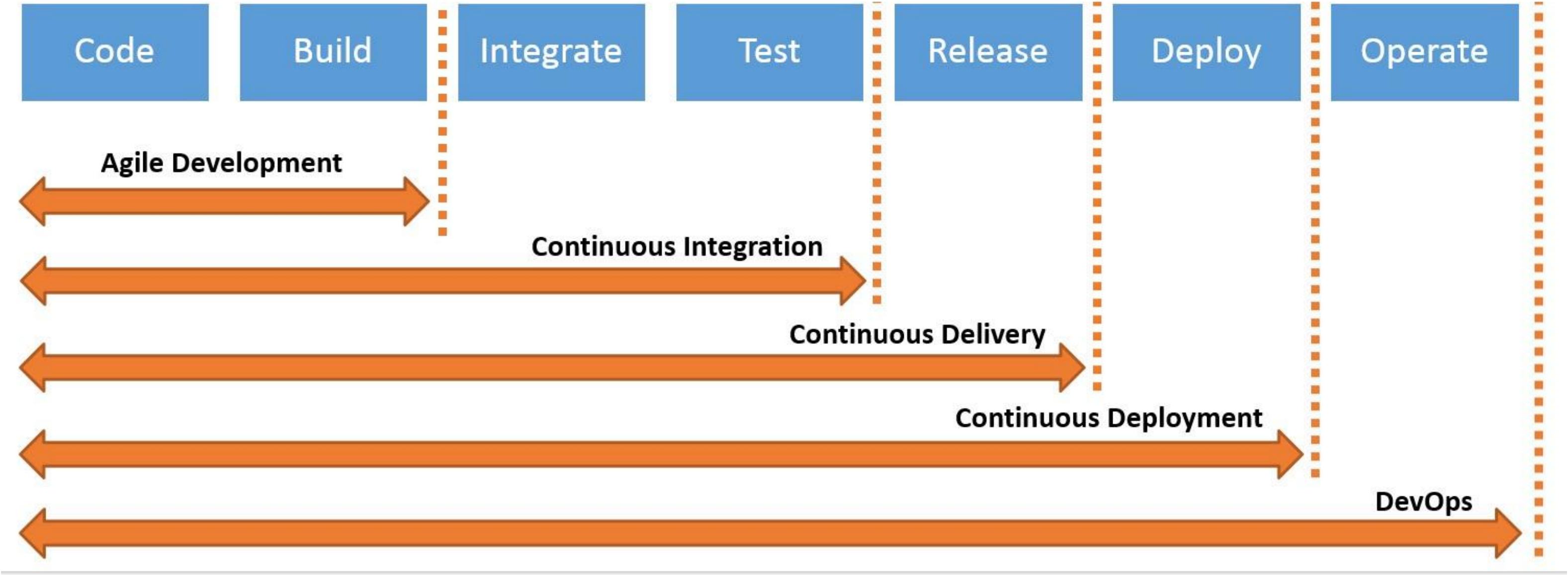
Operation Engineer

Shift Left



DevOps

What is DevOps/DevSecOps?

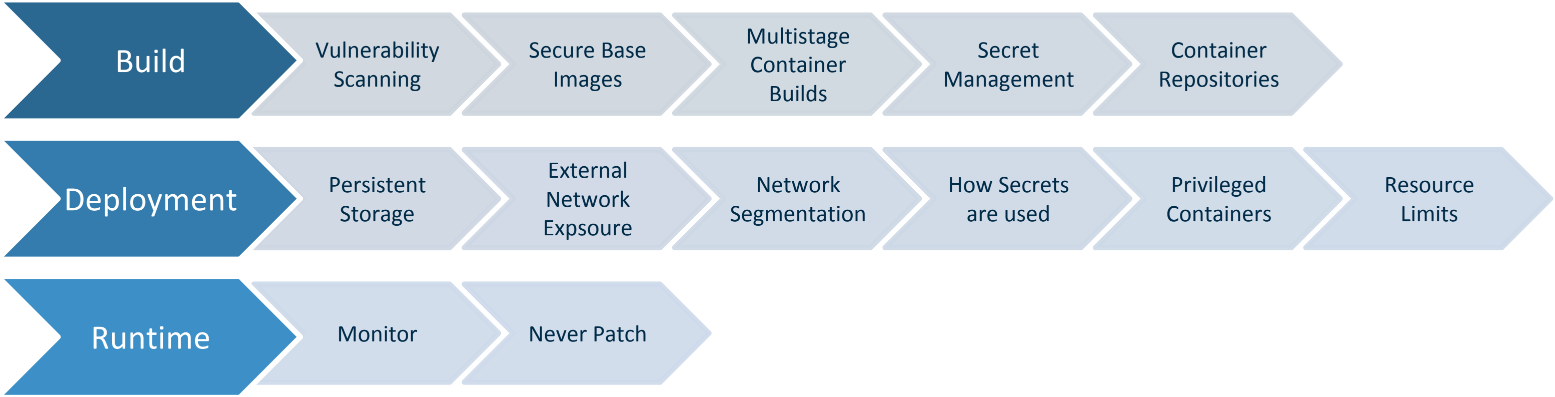


Container

- Images: From Scratch, Repository (e.g. DockerHub), Dockerfile
- Runtime: Docker, lxc, cri-o, containerd
- Orchestration: k8s (Kubernetes), Docker Swarm

Container Lifecycle

Where to do some Security Magic?



Build
- Vulnerability Scanning
- Secure Base Images
- Multistage Container Builds
- Secret Management
- Container Repositories

Deployment
- Persistent Storage
- External Network Expsoure
- Network Segmentation
- How Secrets are used
- Privileged Containers
- Resource Limits

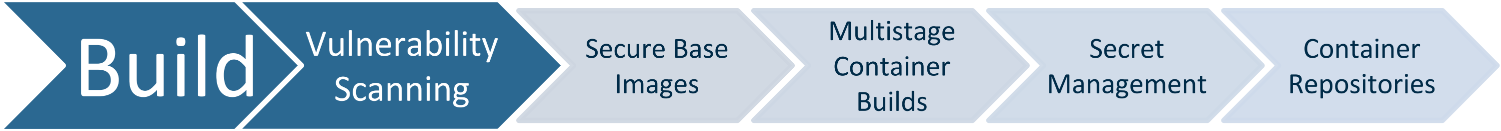
Runtime
- Monitor
- Never Patch

Build Process

Build
Vulnerability Scanning
Secure Base Images
Multistage Container Builds
Secret Management
Container Repositories

- Vulnerability Scanning
- Secure Base Images
- Multistage Container Builds
- Secret Management
- Container Repositories

Vulnerability Scanning



Build

Vulnerability Scanning

Secure Base Images

Multistage Container Builds

Secret Management

Container Repositories

- Use a Vulnerability Scanner
- Scans Artifacts and Containers
  - Including OS Packages
  - Programming Language Dependencies

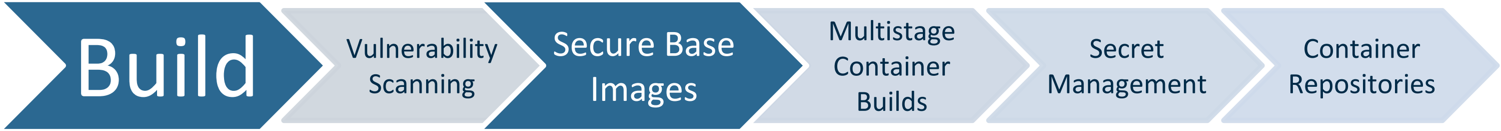
trivy

anchore

clair

Secure Base Images

carmasec



Build

Vulnerability Scanning

Secure Base Images

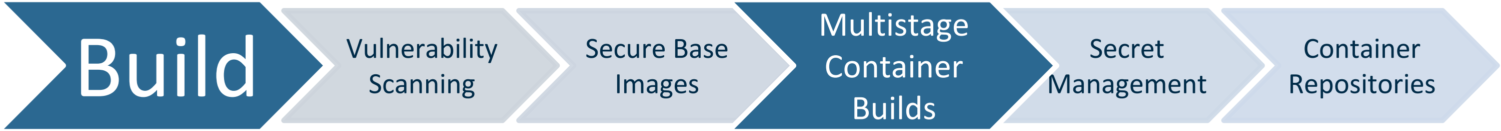
Multistage Container Builds

Secret Management

Container Repositories

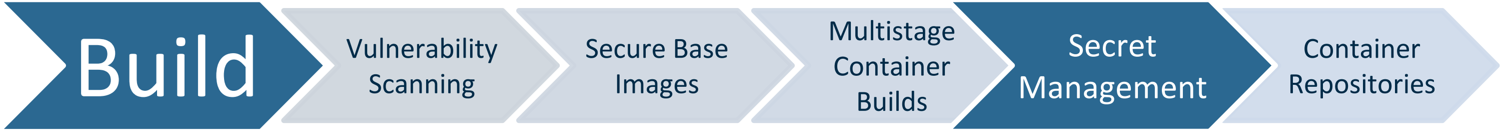
- Create Awareness to use secure base images
- Provide Secure Base Images and patch it frequently
- Provides some peace of mind
- Should be foundation for:
  - Applications
  - Other images

Multistage Container Builds



- One Container with all dependencies and libraries to build the application
- Transfer to a new Container (e.g. „.jar“ file)
- Delete all not necessary dependencies
- Ship only necessary + „.jar“ file to Production

Secret Management


Build

Vulnerability Scanning

Secure Base Images

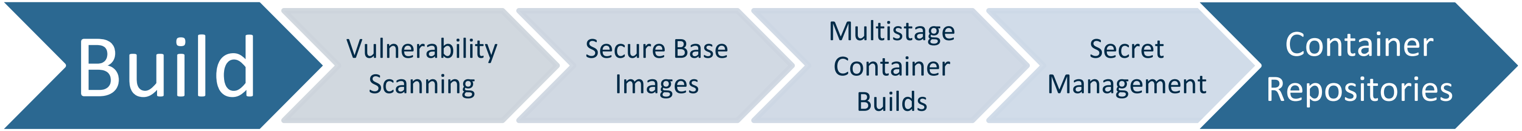
Multistage Container Builds

Secret Management

Container Repositories

- Never store secrets inside an image
  - TLS certificates
  - Credentials
  - SSH keys
  - Database passwords

Container Repositories

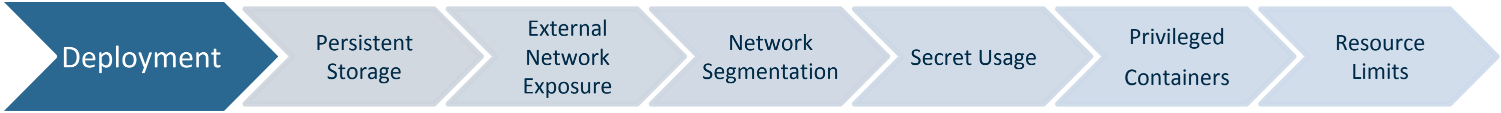


- Provide an own managed Container Repository
- Store already scanned and hardened images
- Reduces the risk of malicious images
- Single Point of Truth for Developers
- e.g. Open Source Container Repository Harbour has Image Scanners included

JFrog ARTIFACTORY

nexus repository

Deployment



Deployment

Persistent Storage

External Network Exposure

Network Segmentation

Secret Usage

Privileged Containers

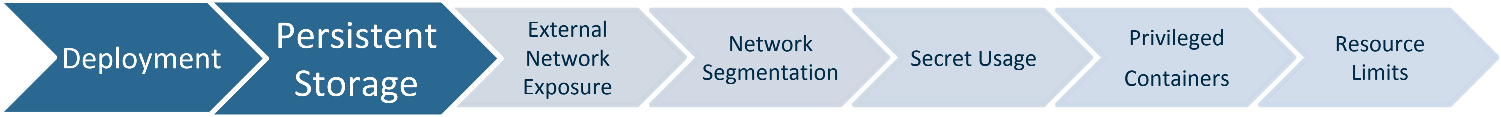
Resource Limits

- Some basic questions to ask
- Persistent Storage
- External Network Expsoure
- Network Segmentation
- Secret Usage
- Privileged Containers
- Resource Limits

Some basic questions to ask

- What it is
- Where it came from
- How it‘s deployed
- What can it access
- Whether it complies

Persistent Storage



Deployment

Persistent Storage

External Network Exposure

Network Segmentation

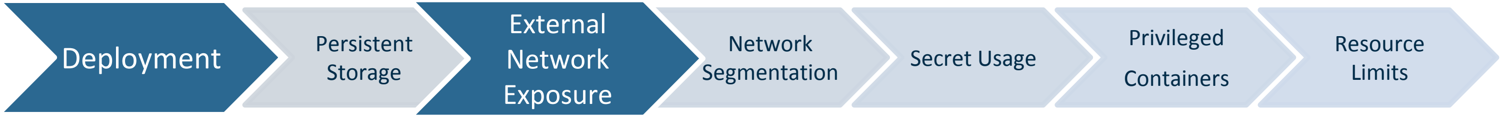
Secret Usage

Privileged Containers

Resource Limits

- Understand how persistent storage is configured and used
- Same goes for Host Mounts
- Specify the access restrictions
- Make read only, if write is not necessary
- Control the users rights, if write is necessary

External Network Exposure



Deployment

Persistent Storage

External Network Exposure

Network Segmentation

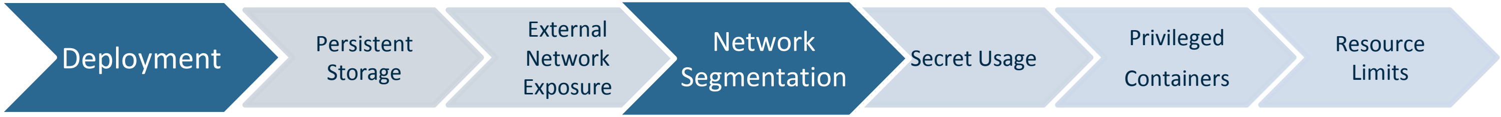
Secret Usage

Privileged Containers

Resource Limits

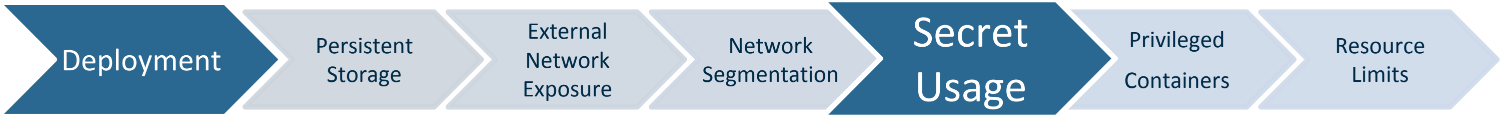
- Have a good overview of the ingress network flows
- Reduce external network exposure where not necessary
- Understand what input comes from external users & services

Network Segmentation



- By default K8s allows every pod to contact every other pod
- Establish Network Segmentation Policies
- This limits the ability of an attacker to move laterally

Secret Usage

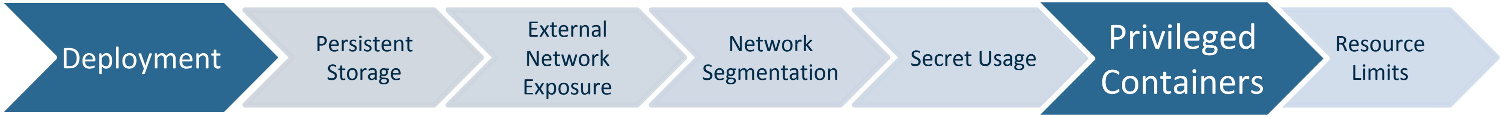


- Understand how secrets are being used
- Enforce Access Controls
- Monitor priviliged Access
- Create more visibility of secrets themself

git-secret

HashiCorp Vault

Privileged Containers



Deployment

Persistent Storage

External Network Exposure

Network Segmentation

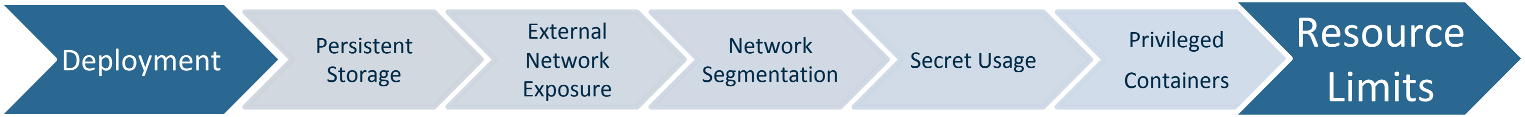
Secret Usage

Privileged Containers

Resource Limits

- Don‘t run privileged Containers, where not needed
- If necessary, understand the capabilities, user identity and privileges granted
- Privileged Container = Privileged Host Process

Resource Limits


Deployment

Persistent Storage

External Network Exposure

Network Segmentation

Secret Usage

Privileged Containers

Resource Limits

- Without resource limits can cause availability issues
- Monitor the ressources → could be indicator of compromise

Prometheus

INSTANA

elastic + logstash

kibana

Grafana

Runtime

Runtime

Monitoring

Never Patch

- Monitoring: What to watch?
- Never Patch

Monitoring - What to watch?

carmasec

Runtime

Monitoring

Never Patch

- Monitor running deployments for newly discovered Vulnerabilities
- Get visibility ...
    - ... of active network traffic between running container
    - ... between container and external clients / servers / services

Never Patch

Runtime
Monitoring
Never Patch

- Never patch/update a running container
- Scale to zero and let restart
  - Faster
  - Easier
  - More Secure

Questions and Discussions

Kevin Kloft

- E-Mail: kevin.kloft@carmasec.com
- twitter: @kevsecops

- https://www.carmasec.com/ISX

carmasec

security. done. right.

Hauptsitz:
carmasec GmbH & Co. KG
Ruhrallee 185
45136 Essen

Niederlassung:
carmasec GmbH & Co. KG
Im Mediapark 5
50670 Köln

Telefon:	+49 (0) 201 426 385 900
Fax:	+49 (0) 201 426 385 909
Web:	www.carmasec.com
Email:	contact@carmasec.com