

Agendavorschlag Docker Container

Grundlagen der Containerisierung

- Motivation: warum Containerisierung? Vor- und Nachteile
- Einführung in Images
- Begriffsklärung: Image, Container, Swarm, Registry, Engine, Hub (ohne tief ins Detail zu gehen, kommt später)
- Speziell Docker: Zusammenhang von Container Image und Gastbetriebssystem
- Docker CE vs. EE

Docker-Grundlagen

Alles immer mit Kommandozeile (docker-Kommando) und praktischen Übungen!

- Docker-Basisinstallation
 - Docker Agent (Engine)
 - docker command
 - Globale Konfiguration von Docker: Systemkonfigurationsdateien
 - Administration lokal mit CLI vs. TCP
- Erstellung von Images
 - Dockerfile: Aufbau und Syntax
 - Base Images
- Managen von Images
 - Image-Repositories (Docker Registry)
 - Docker Hub vs. interne Registries (Nexus, VMWare Harbor)
- Docker Continuous Integration
- Volumes und persistente Datenspeicherung

Orchestrierung: Application Stacks mit Containern

Docker Networking-Konzepte sind wichtig und müssen verstanden werden. Docker Swarm nur anreißen und ein paar einfache Beispiele machen, weil wir bei Orchestrierungslösungen voraussichtlich eher auf Kubernetes/Pivotal setzen werden.

- Docker Networking
- Docker Compose
 - docker-compose.yml, Sprachversionen
 - docker-compose vs. docker stack ...
- Swarm Mode
- Swarm Operations
- Managing Secrets

Docker Container Security

- Isolation Kernel Namespaces and Control Groups
- Benutzermanagement
 - Mapping containerinterne/externe User
 - (Quasi-)Root-Berechtigung: welche Privilegien hat ein Mitglied der docker-Gruppe?
- Intra-Plattform-Kommunikation
- Netzwerke
- Image Construction und Scanning
- Content Trust
- Seccomp
- Linux Sicherheits-Module

Containerorchestrierung: Kubernetes

- Orchestrierung mit Kubernetes (K8s)
 - Einführung in K8s: Motivation, Architektur und Komponenten
 - Möglichkeiten der Orchestrierung
- Installation und CLI Tools
- Hochverfügbarkeit, Ressource Sharing, Load Balancing
- Automatische Skalierung, Rolling Upgrades
- Health Check und Self Healing, Service Discovery
- Praxisbeispiele
- Kommerzielle Lösungen, z.B. Pivotal