



KATEDRA
INFORMATIKY

UNIVERZITA PALACKÉHO V OLMOUCI

Docker

David Votočka

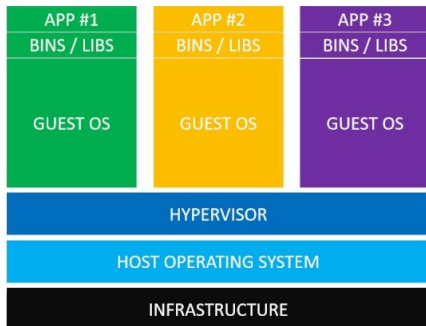
Obsah

- Co je Docker?
- Virtuální stroj vs Docker
- Výhody Dockeru
- Jak Docker funguje?
- DEMO
- Ostatní

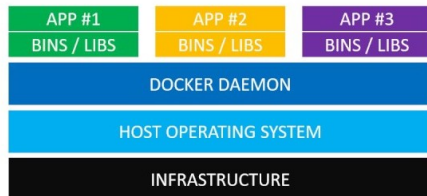
Co je Docker?

- open source platforma pro vývoj, distribuci a spouštění aplikací
- Umožňuje vytváření a běh v Docker kontejnerech se všemi závislostmi
- Aplikaci zabalí do kontejneru včetně frameworků, knihoven, kódu, běhového prostředí, ...
- podobný klasické virtualizaci
- Dostupný na většině Linux distribucí, macOS i Windows

Virtuální stroj vs. Docker



Virtual Machines



Docker Containers

Výhody Dockeru

- odstraňuje klasický problém se závislostmi při přenosu
- Díky virtualizaci OS se sdíleným kernelem z hostovaného OS je výrazně rychlejší
- Samotné aplikace jsou izolovány na úrovni procesů což stále zajišťuje vysokou bezpečnost
- přenositelnost (na stroji je nutné mít nainstalovaný pouze Docker)
- rozdělení HW prostředků
- sjednocení prostředí při vývoji, testování i v produkci
- rychlost nasazení a testování

Jak Docker funguje?

Základní pojmy

- **Image** je neměnná šablona aplikace (statický snapshot)
- **Container** je běžící instance image (izolovaný proces, sdílí kernel hosta)
- **Dockerfile** je soubor obsahující instrukce jako FROM, RUN, COPY, ... (jak sestavit prostředí dané aplikace)
 - **Docker compose**
- **Networking**
 - Každý container má přístup k síti a může komunikovat s dalšími containery i mimo docker síť
- **Registry** Centralizované úložiště Docker images
 - Docker hub (veřejné)
 - Možnost vytváření soukromých
 - nahrávání a stahování images pomocí příkazů push / pull

Jak Docker funguje?

Dockerfile

```
1 FROM node:18-alpine
2
3 WORKDIR /usr/src/app
4
5 COPY package*.json ./
6
7 RUN npm install --production
8
9 COPY . .
10
11 EXPOSE 3000
12
13 CMD ["node", "index.js"]
14
```

Jak Docker funguje?

Docker compose

```
1  services:
2    backend:
3      build:
4        context: ./server
5        container_name: backend
6      ports:
7        - "3000:3000"
8      env_file:
9        - .env
10
11    frontend:
12      build:
13        context: ./client
14        container_name: frontend
15      depends_on:
16        - backend
17      ports:
18        - "8080:8080"
19
```


Jak Docker funguje?

Architektura

- Docker je client-server služba
- Docker client (Prostředí pro uživatele Docker CLI / Docker Desktop)
- Docker daemon (stará se o běh kontejnerů, přiřazování zdrojů, ...)
- Komunikace mezi klientem a serverem pomocí REST API
- celek obsahující CLI, daemon a REST API = Docker engine

Základní příkazy

- `docker version` – verze klienta a serveru
- `docker images` – seznam lokálních image
- `docker ps -a` – běžící i zastavené kontejnery
- `docker pull image` – stáhne image z registry
- `docker build -t name:tag .` – postaví image z Dockerfile
- `docker run -d -name c -p host:cont image` – spustí kontejner
- `docker logs -f c` – sleduje logy běžícího kontejneru.
- `docker exec -it c sh` – interaktivní shell uvnitř kontejneru.
- `docker stop c` / `docker start c` – zastaví/spustí kontejner
- `docker rm -f c` – smaže kontejner
- `docker save img:tag -o file.tar` – uloží image do tar
- `docker load -i file.tar` – načte image z tar
- `docker compose up -d` / `down -v` – spustí/smaže multi-slужby

DEMO

- Ukázka použití

Ostatní

- Docker je dnes využíván například firmami Spotify, Netflix, PayPal
- Docker + Kubernetes
 - Kubernetes je open-source platforma pro nasazování, škálování a správu kontejnerizovaných
- Docker Swarm
 - nativní orchestrátor vestavěný v Docker Engine