

Docker příklady

Příklady užití

images

- docker images
- docker images -a
- docker images -q
- docker images --no-trunc
- docker images --digests
- docker images --help

docker run

- `docker run --name="JindrichKana" --privileged=true -it -v mynfs:/app centos bash`
- `docker run -d -v /usr/sbin:/target maruska`
- `docker run -d -p 82:8080 --network user0-net3 csasdemo`
- `docker run --name=jindrovo -it -v /usr/sbin:/target centos bas`

docker ps

```
[jindrich.kana@jindrovo ~]$ docker ps --help
```

Usage: docker ps [OPTIONS]

List containers

Options:

- a, --all Show all containers (default shows just running)
- f, --filter filter Filter output based on conditions provided
 - format string Pretty-print containers using a Go template
- help Print usage
- n, --last int Show n last created containers (includes all states) (default -1)
- l, --latest Show the latest created container (includes all states)
 - no-trunc Don't truncate output
- q, --quiet Only display numeric IDs
- s, --size Display total file sizes

pull & push

- `docker pull tomcat`
- `docker push kanaajin/csas/webcentos:latest`

storage

```
docker volume create myvol
docker volume ls
docker volume create --driver local --opt type=nfs --opt o=addr=IP_ADDRESS,rw --opt device=:/path/to/dir foo
docker volume create --driver local --opt type=tmpfs --opt device=tmpfs --opt o=size=100m,uid=1000 netcat1
docker service create -d --replicas=4 --name devtest-service --mount source=myvol2,target=/app nginx:latest
docker volume create --driver local --opt type=nfs --opt device=:/nfsexport --opt "o=addr=localhost,rw" mynfs
docker run --privileged=true -it -v mynfs:/app centos bash
```

Připojení NFS serveru

```
[jindrich.kana@jindrovo termin-2]$ cat /etc/exports
/nfsexport *(fsid=0,rw)
docker volume create --driver local --opt type=nfs --opt device=:/nfsexport --opt "o=addr=localhost,rw" csasnfs
[root@jindrovo ~]# docker run -d --name csasplainlinux --privileged=true -v csasnfs:/app centos
bcb8f10bc160c635c84d767cb2220173be950b14c172ec133a117539dd503208
```

docker network



Vytvoření třech kontejnerů s aplikací netcat - a) spuštění kontejneru na pozadí, b) expose kontejneru na portu 808{1,2,3}, c) připojení docker host adresáře /sbin v kontejneru do /sbin, d) pojmenování kontejneru.

```
docker run -d -p 8081:8081 -v /sbin:/sbin --name netcat1 netcat1
docker run -d -p 8082:8082 -v /sbin:/sbin --name netcat2 netcat2
docker run -d -p 8083:8083 -v /sbin:/sbin --name netcat3 netcat3
```

Vytvoření třech sítí - a) v módu bridge, b) attachable/možno připojit kontejner za běhu, c) definice sítě, d) specifikace brány, e) jméno sítě

```
docker network create -d bridge --attachable --subnet 172.0.1.0/24 --gateway 172.0.1.1 netcat1
docker network create -d bridge --attachable --subnet 172.0.2.0/24 --gateway 172.0.2.1 netcat2
docker network create -d bridge --attachable --subnet 172.0.3.0/24 --gateway 172.0.3.1 netcat3
```

Připojení dvou kontejnerů do sítě - každý do své vlastní.

```
docker network connect netcat1 netcat1
docker network connect netcat2 netcat2
```

Připojení třetího kontejneru do všech třech sítí.

```
docker network connect netcat1 netcat3
docker network connect netcat2 netcat3
docker network connect netcat3 netcat3
```

Více....

```
docker-network-connect (1) - connect a container to a network
docker-network-create (1) - create a new network
docker-network-disconnect (1) - disconnect a container from a network
docker-network-inspect (1) - inspect a network
docker-network-ls (1) - list networks
docker-network-rm (1) - remove one or more networks
```

inspect

```
docker inspect csasapp --format '{{  
.NetworkSettings.Networks.bridge.IPAddress }}'
```