1 Docker CheatSheet Cloud

- PDF Link: cheatsheet-docker-A4.pdf, Category: Cloud
- Blog URL: `https://cheatsheet.dennyzhang.com/cheatsheet-docker-A4`
- Related posts: Kubernetes Yaml, #denny-cheatsheets

File me Issues or star this repo.

1.1 Container Runtime

Name	Summary
dockerd	
containerd	
cri-o	From Redhat
rkt	a pod-native container engine for Linux from CoreOS. Stopped maintainance now
Amazon ACS	supports DC/OS, Swarm, Kubernetes
CoreOS Fleet	
Cloud Foundry Diego	Not actively maintained any more
Reference	CheatSheet: Docker, CheatSheet: CRI-O, CheatSheet: rkt, CheatSheet: containerd

1.2 Docker Trouble Shooting

Name	Summary
Docker push: manifest invalid	Re-push a new version of the same docker tag may fail, due to permis
Docker pull: missing signature key	Docker push again to resolve the issue
Docker cp: Error response from daemon: not a directory	container folder is in a symbol link
Find process id by container name	`docker top $container_id`, or `docker top $container_name`
Get dockerd storage driver	`docker info`, then check `Storage Driver`
docker-containerd-shim	The Docker four components: `Docker engine`, `containerd`, `contain`

1.3 Docker Basic

Name	Summary
Install docker on Ubuntu	`apt-get install docker.io`
Install docker on CentOS	Use docker repo `https://download.docker.com/linux/centos/docker-ce.repo`
Install docker in Debian 10	Link: How To Install and Use Docker on Debian 10
Install old docker version	GitHub: install-old-docker.md

1.4 Docker start service

Name	Summary
Start a ubuntu test env	`docker run ubuntu:16.04 /bin/echo hello world`
Start a ubuntu 18.04 test env	`docker run ubuntu:18.04 /bin/echo hello world`
Start a debian9 test env	`docker run debian:9 /bin/echo hello world`
Start a centos test env	`docker run centos:centos6 /bin/echo hello world`
Start a jenkins server	`docker run -p 8080:8080 -p 50000:50000 jenkins/jenkins:lts`
Start a nginx server	`docker run -t -d -p 8080:80 --name nginx-test nginx`
Start a nexus server	`docker run -d -p 8082:8081 --name nexus -v /data/nexus-data:/nexus-data sonatype/`
Start a sshd server	`docker run -t -d --privileged -p 5022:22 denny/sshd:latest /usr/sbin/sshd -D`
Start a ftp server	`docker run -t -d -p 21:21 -p 20:20 -e USERNAME=${username} -e PASSWORD=${password`

1.5 Container Basic

Name	Summary
Start docker container	`docker run -p 4000:80 imgname`
Start docker container in detached mode	`docker run -d -p 4000:80 imgname`
Start container with entrypoint changed	`docker run -t -d --entrypoint=/bin/sh "$docker_image"`
Enter a running container	`docker exec -it <container-id> sh`
Upload local file to container filesystem	`docker cp /tmp/foo.txt mycontainer:/foo.txt`
Download container file local filesystem	`docker cp mycontainer:/foo.txt /tmp/foo.txt`
Stop container	`docker stop <hash>`
Remove container	`docker rm <hash>`
Remove all containers	`docker rm $(docker ps -a -q)`
Force shutdown of one given container	`docker kill <hash>`
Login to docker hub	`docker login`
Tag <image>	`docker tag <image> username/repo:tag`
Docker push a tagged image to repo	`docker push username/repo:tag`
Run image from a given tag	`docker run username/repo:tag`
Create docker image	`docker build -t denny/image:test .`

1.6 Docker Cleanup

Name	Summary
Delete all containers	delete-all-containers.sh
Remove unused docker images	delete-unused-images.sh
Docker prune images	`docker image prune -f`
Docker prune volumes	`docker volume prune -f`
Remove the specified image	`docker rmi <imagename>`
Remove all docker images	`docker rmi $(docker images -q)`
Remove orphaned docker volumes	`docker volume rm $(docker volume ls -qf dangling=true)`
Remove dead containers	`docker rm $(docker ps --filter status=dead -qa)`
Remove exited containers	`docker rm $(docker ps --filter status=exited -qa)`

1.7 Dockerfile

Name	Summary
Change entrypoint to run nothing	`entrypoint: ["tail", "-f", "/dev/null"]`
Set timezone in Dockerfile	`RUN ln -snf /usr/share/zoneinfo/$TZ /etc/localtime && echo $TZ > /etc/timezon`
Define multiple line command	GitHub: Dockerfile-example-multiline

1.8 Docker Compose

Name	Summary
Change restart policy	`restart: always`, Link: Compose file version 3 reference
Mount file as volume	`$PWD/httpd/httpd.conf:/usr/local/apache2/conf/httpd.conf:ro` GitHub: sample-mount-file.yml
Start compose env	`docker-compose up`, `docker-compose up -d`
Stop compose env	`docker-compose down`, `docker-compose down -v`
Check logs	`docker-compose logs`

1.9 Docker Containers

Name	Summary
Start docker container	`docker run -p 4000:80 <imgname>`
Start docker container in detached mode	`docker run -d -p 4000:80 imgname`
Start docker container and remove when exit	`docker run -rm -it <imgname> sh`
Enter a running container	`docker exec -it [container-id] sh`
Stop container	`docker stop <hash>`
List all containers	`docker ps`, `docker ps -a`
Remove container	`docker rm <hash>`, `docker rm $(docker ps -a -q)`
Force shutdown of one given container	`docker kill <hash>`
Login to docker hub	`docker login`
Run image from a given tag	`docker run username/repo:tag`
Tail container logs	`docker logs --tail 5 $container_name`
Check container healthcheck status	`docker inspect --format '{{.State.Health}}' $container_name`
List containers by labels	`docker ps --filter "label=org.label-schema.group"`

1.10 Docker Images

Name	Summary
List all images	`docker images`, `docker images -a`
Create docker image	`docker build -t denny/image:<tag> .`
Docker push a tagged image to repo	`docker push denny/image:<tag>`
Show the history of an image	`docker history <image_name>`
Export image to file	`docker save <image_name> > my_img.tar`
Load image to local registry	`docker load -i my_img.tar`
Tag <image>	`docker tag <image> username/repo:tag`

1.11 Docker Socket file

Name	Summary
Run container mounting socket file	`docker run -v /var/run/docker.sock:/var/run/docker.sock -it alpine sh`
A different docker socket file	`export DOCKER_HOST=unix:///my/docker.sock`
List containers	`curl -XGET --unix-socket /var/run/docker.sock http://localhost/containers/js`
Stop container	`curl -XPOST --unix-socket /var/run/docker.sock http://localhost/containers/<`
Start container	`curl -XPOST --unix-socket /var/run/docker.sock http://localhost/containers/<`
List events	`curl --unix-socket /var/run/docker.sock http://localhost/events`
Create container	`curl -XPOST --unix-socket /var/run/docker.sock -d '{"Image":"nginx:alpine"}'`
Links	Link: Develop with Docker Engine SDKs and API

1.12 Docker Conf

Name	Summary
Docker files	`/var/lib/docker`, `/var/lib/docker/devicemapper/mnt`
Docker for Mac	`~/Library/Containers/com.docker.docker/Data/`

1.13 Ubuntu docker: Install missing packages

Name	Summary
Pull ubuntu docker image	`docker pull ubuntu`
man: command not found	`apt-get update`, `apt-get install man`
ping: command not found	`apt-get update`, `apt-get install iputils-ping`
dig: command not found	`apt-get install dnsutils`

1.14 Check Status

Name	Summary
Tail container logs	`docker logs --tail 5 $container_name`
Check container healthcheck status	`docker inspect --format '{{.State.Health}}' $container_name`
List containers	`docker ps`
List all containers	`docker ps -a`
List containers by labels	`docker ps --filter "label=org.label-schema.group"`
List all images	`docker images -a`

1.15 Resource Reference

Name	Summary
Docker SDK	`https://docs.docker.com/develop/sdk/examples/`
Docker REST API	`https://docs.docker.com/engine/api/v1.27/#tag/Container`
Docker Hub auto build	`https://docs.docker.com/docker-hub/builds/#build-statuses-explained`

1.16 More Resources