Шпаргалка с командами Docker

[Блог компании Флант](https://habr.com/ru/company/flant/blog/)[Системное администрирование\*](https://habr.com/ru/hub/sys_admin/)[Серверное администрирование\*](https://habr.com/ru/hub/s_admin/)[DevOps\*](https://habr.com/ru/hub/devops/)

Перевод

[Автор оригинала: Aymen El Amri](https://github.com/eon01/DockerCheatSheet" \t "_blank)

***Прим. перев.****: Неделю назад Aymen El Amri, руководящий компанией eralabs и создавший обучающий курс «Безболезненный Docker»,*[*опубликовал*](https://medium.com/statuscode/dockercheatsheet-9730ce03630d)*свой Docker Cheat Sheet — шпаргалку по основным командам Docker. [Git-репозиторий](https://github.com/eon01/DockerCheatSheet) этого документа на GitHub уже набрал 1000+ stars и несколько сторонних контрибьюторов, что подтвердило его актуальность и пользу.*  
  
  
  
*Представленные здесь команды описаны минимально (с акцентом на читаемость как есть) и включают в себя установку Docker, работу с реестрами и репозиториями, контейнерами, образами, сетью, Docker Swarm. Ниже представлен перевод шпаргалки в её состоянии на 2 сентября с дополнениями из комментариев ниже.*

Установка

Linux

curl -sSL https://get.docker.com/ | sh

Mac

Скачайте dmg по этой ссылке:

https://download.docker.com/mac/stable/Docker.dmg

Windows

Используйте MSI-инсталлятор:

https://download.docker.com/win/stable/InstallDocker.msi

Реестры и репозитории Docker

Вход в реестр

docker login

docker login localhost:8080

Выход из реестра

docker logout

docker logout localhost:8080

Поиск образа

docker search nginx

docker search nginx -- filter stars=3 --no-trunc busybox

Pull (выгрузка из реестра) образа

docker pull nginx

docker pull eon01/nginx localhost:5000/myadmin/nginx

Push (загрузка в реестр) образа

docker push eon01/nginx

docker push eon01/nginx localhost:5000/myadmin/nginx

Первые действия с контейнерами

Создание контейнера

docker create -t -i eon01/infinite --name infinite

Первый запуск контейнера

docker run -it --name infinite -d eon01/infinite

Переименование контейнера

docker rename infinite infinity

Удаление контейнера

docker rm infinite

Обновление контейнера

docker update --cpu-shares 512 -m 300M infinite

Запуск и остановка контейнеров

Запуск остановленного контейнера

docker start nginx

Остановка

docker stop nginx

Перезагрузка

docker restart nginx

Пауза (приостановка всех процессов контейнера)

docker pause nginx

Снятие паузы

docker unpause nginx

Блокировка (до остановки контейнера)

docker wait nginx

Отправка SIGKILL (завершающего сигнала)

docker kill nginx

Отправка другого сигнала

docker kill -s HUP nginx

Подключение к существующему контейнеру

docker attach nginx

Получение информации о контейнерах

Работающие контейнеры

docker ps

docker ps -a

Логи контейнера

docker logs infinite

Информация о контейнере

docker inspect infinite

docker inspect --format '{{ .NetworkSettings.IPAddress }}' $(docker ps -q)

События контейнера

docker events infinite

Публичные порты

docker port infinite

Выполняющиеся процессы

docker top infinite

Использование ресурсов

docker stats infinite

Изменения в файлах или директориях файловой системы контейнера

docker diff infinite

Управление образами

Список образов

docker images

Создание образов

docker build .

docker build github.com/creack/docker-firefox

docker build - < Dockerfile

docker build - < context.tar.gz

docker build -t eon/infinite .

docker build -f myOtherDockerfile .

curl example.com/remote/Dockerfile | docker build -f - .

Удаление образа

docker rmi nginx

Загрузка репозитория в tar (из файла или стандартного ввода)

docker load < ubuntu.tar.gz

docker load --input ubuntu.tar

Сохранение образа в tar-архив

docker save busybox > ubuntu.tar

Просмотр истории образа

docker history

Создание образа из контейнера

docker commit nginx

Тегирование образа

docker tag nginx eon01/nginx

Push (загрузка в реестр) образа

docker push eon01/nginx

Сеть

Создание сети

docker network create -d overlay MyOverlayNetwork

docker network create -d bridge MyBridgeNetwork

docker network create -d overlay \

--subnet=192.168.0.0/16 \

--subnet=192.170.0.0/16 \

--gateway=192.168.0.100 \

--gateway=192.170.0.100 \

--ip-range=192.168.1.0/24 \

--aux-address="my-router=192.168.1.5" --aux-address="my-switch=192.168.1.6" \

--aux-address="my-printer=192.170.1.5" --aux-address="my-nas=192.170.1.6" \

MyOverlayNetwork

Удаление сети

docker network rm MyOverlayNetwork

Список сетей

docker network ls

Получение информации о сети

docker network inspect MyOverlayNetwork

Подключение работающего контейнера к сети

docker network connect MyOverlayNetwork nginx

Подключение контейнера к сети при его запуске

docker run -it -d --network=MyOverlayNetwork nginx

Отключение контейнера от сети

docker network disconnect MyOverlayNetwork nginx

Очистка Docker

Удаление работающего контейнера

docker rm nginx

Удаление контейнера и его тома (volume)

docker rm -v nginx

Удаление всех контейнеров со статусом exited

docker rm $(docker ps -a -f status=exited -q)

Удаление всех остановленных контейнеров

docker container prune

docker rm `docker ps -a -q`

Удаление контейнеров, остановленных более суток назад

docker container prune --filter "until=24h"

Удаление образа

docker rmi nginx

Удаление неиспользуемых (dangling) образов

docker image prune

docker rmi $(docker images -f dangling=true -q)

Удаление неиспользуемых (dangling) образов даже с тегами

docker image prune -a

Удаление всех образов

docker rmi $(docker images -a -q)

Удаление всех образов без тегов

docker rmi -f $(docker images | grep "^<none>" | awk "{print $3}")

Остановка и удаление всех контейнеров

docker stop $(docker ps -a -q) && docker rm $(docker ps -a -q)

Удаление неиспользуемых (dangling) томов

docker volume prune

docker volume rm $(docker volume ls -f dangling=true -q)

Удаление неиспользуемых (dangling) томов по фильтру

docker volume prune --filter "label!=keep"

Удаление неиспользуемых сетей

docker network prune

Удаление всех неиспользуемых объектов

docker system prune

По умолчанию для Docker 17.06.1+ тома не удаляются. Чтобы удалились и они тоже:

docker system prune --volumes

Docker Swarm

Установка Docker Swarm

curl -ssl https://get.docker.com | bash

***Прим. перев.****: в Docker версий 1.12.0+ ничего дополнительно устанавливать не требуется, т.к. Docker Swarm встроен в Docker Engine в виде специального режима (Swarm mode).*

Инициализация Swarm

docker swarm init --advertise-addr 192.168.10.1

Подключение рабочего узла (worker) к Swarm

docker swarm join-token worker

Подключение управляющего узла (manager) к Swarm

docker swarm join-token manager

Список сервисов

docker service ls

Список узлов

docker node ls

Создание сервиса

docker service create --name vote -p 8080:80 instavote/vote

Список заданий Swarm

docker service ps

Масштабирование сервиса

docker service scale vote=3

Обновление сервиса

docker service update --image instavote/vote:movies vote

docker service update --force --update-parallelism 1 --update-delay 30s nginx

docker service update --update-parallelism 5--update-delay 2s --image instavote/vote:indent vote

docker service update --limit-cpu 2 nginx

docker service update --replicas=5 nginx

P.S.

***Прим. перев.****: Напомню, что оригинальная (англоязычная) версия Docker Cheat Sheet доступна и обновляется в [Git-репозитории](https://github.com/eon01/DockerCheatSheet). Автор будет рад исправлениям/пополнениям от сообщества.*