

Docker вступление

Тимур Татаршаов

2018

Нижний Новгород

Обо мне

Тимур Татаршаов

tatarshaov@gmail.com

<https://famer.github.io>

Czech Technical University MSc in Software and Web development

Ex RedHat-er

Содержание

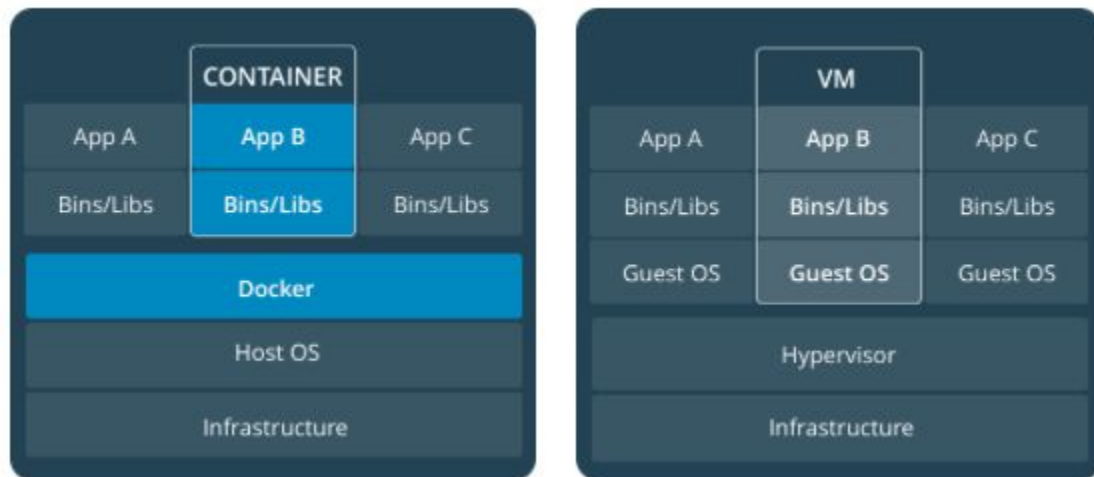
- История становления
- Контейнеризация в сравнении с виртуализацией
- Устройство образа и контейнера
- Dockerfile, база для сборки образа
- Дистрибуция приложений с помощью Docker
- Для чего хорош и плох Docker

История

- Linux containers
 - Появились в 2008 (LXC)
 - Позволяли запускать дерево процессов в изолированной, на системном уровне, “виртуализации”
 - используют меньше ресурсов и дискового пространства, чем традиционная виртуализация
- Docker
 - Используют новые функции ядра(cgroups, namespaces)
 - Позволяет собирать и делиться образами
 - Репозиторий с большим количеством образов

Виртуализация в сравнении с контейнеризацией

Контейнеры изолированы, но переиспользуют ОС и, где допустимо, программы и библиотеки



Виртуализация против контейнеризации

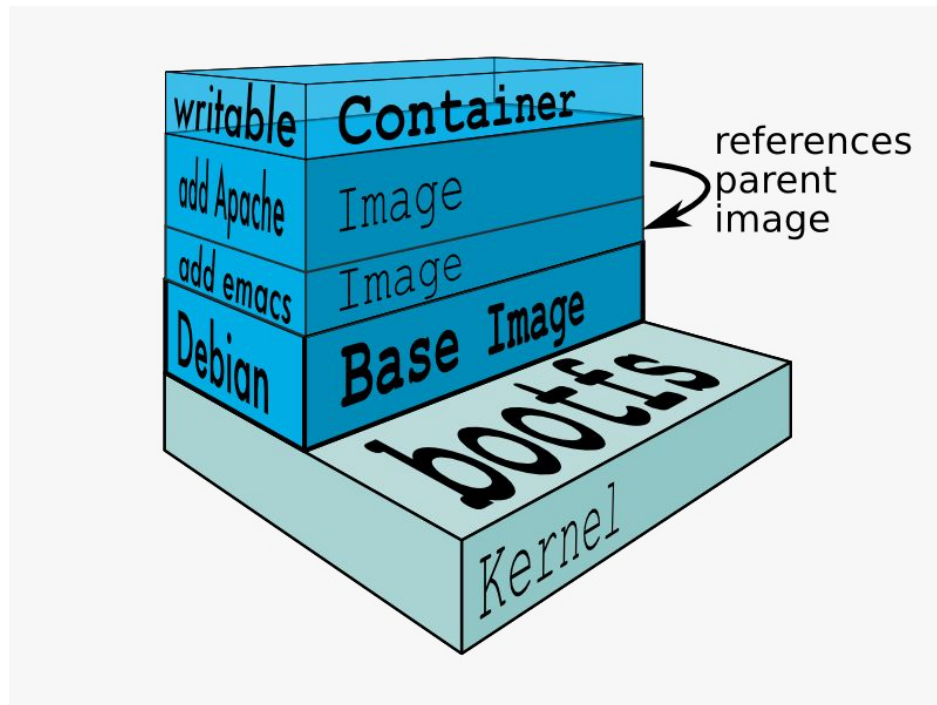
- Утилизация ресурсов и дискового пространства
- Время старта
- Время деплоя образа

Терминология

- Образ(image) — база для контейнера
 - содержит объединение слоёв файловых систем, наложенных друг на друга
 - не изменяется и не имеет состояния(кроме одного)
- Контейнер(container) — запущенный образ
 - имеет временное хранилище
 - эфемерен, сбрасывает состояние и временное хранилище при следующем запуске

Docker образ и контейнер

Dockerfile -> Образ -> Контейнер



Docker контейнер

- приспособлен для запуска одного приложения
- упаковка для приложения с зависимостями
- эфемерен, постоянные данные хранятся в примонтированных папках или контейнерах данных
- консистентен между различными средами выполнения и машинами

Dockerfile

- Dockerfile

- FROM oraclelinux:7
MAINTAINER Timur Tatarshaov <tatarshaov@gmail.com>
RUN yum install -q -y httpd
EXPOSE 80
CMD httpd -X
- Одна строка в Dockerfile-е -- один промежуточный слой образа
- \$ docker build -t famer/httpd:v1 .
Sending build context to Docker daemon 2.048 kB
Step 1 : FROM oraclelinux:7
---> 4c357c6e421e
Step 2 : MAINTAINER Timur Tatarshaov <tatarshaov@gmail.com>
---> Running in 35feebb2ffab
---> 95b35d5d793e
Removing intermediate container 35feebb2ffab Step 3 : RUN yum install -q -y httpd ...

Слоение образов

- Загрузка образа из реестра

```
$ sudo docker pull ubuntu
```

```
Using default tag: latest
```

```
latest: Pulling from library/ubuntu
```

```
5ba4f30e5bea: Pull complete
```

```
9d7d19c9dc56: Pull complete
```

```
ac6ad7efd0f9: Pull complete
```

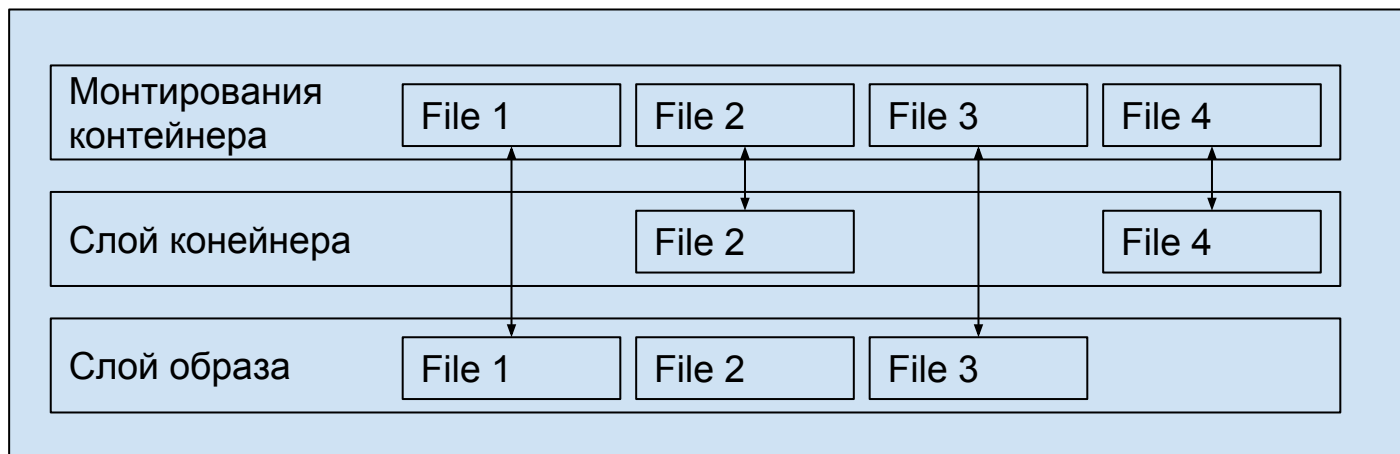
```
e7491a747824: Pull complete
```

```
a3ed95caeb02: Pull complete
```

```
Digest: sha256:46fb5d001b88ad904c5c732b086b596b92cfb4a4840a3abd0e35dbb6870585e4
```

```
Status: Downloaded newer image for ubuntu:latest
```

Файловая система контейнера



Запуск

```
$ docker run -v /data:/var/lib/postgresql/data postgres
```

Дистрибуция приложений

- Реестр(Registry)
- Dockerfile + хранилище с приложением
- HTTP сервер
- Образ в виде *.tar файла

Плюсы использования

- приложения консистентны
- одинаковое окружение работы приложения в различных системах (требуется лишь наличие Docker Engine)
- дистрибуция приложения с зависимостями
- изоляция приложения
- 12factor

Для чего плох(пока еще) Docker

- не (микро)сервисной архитектуры приложений
- дополнительное дисковое пространство(переиспользуется за счет слоев)
- приложений GUI

Вопросы?