

docker 技术分享

冯海雄

May 5, 2015

目录

docker 技术分
享
冯海雄

目录

docker 是什么

docker 的基本
内容

docker 的安装

docker 的使用
实例

docker 集群

docker 是什么

docker 的基本内容

docker 的安装

docker 的使用实例

docker 集群

docker

docker 技术分

享

冯海雄

目录

docker 是什么

why docker

docker 的基本
内容

docker 的安装

docker 的使用
实例

docker 集群

Docker 是一个方便开发人员和系统管理员开发、打包、运行程序的开放式平台, Docker 是 PaaS 提供商 dotCloud 开源的一个基于 LXC 的高级容器引擎, 2013 年将源代码托管在 Github 上, 使用 go 语言编写遵从 Apache2.0 协议开源, Docker 目前的版本是 1.6.0。dotCloud 主要从事于 PaaS 的服务开发, PaaS 是 Platform As a service 的缩写

docker

docker 技术分
享
冯海雄

目录

docker 是什么

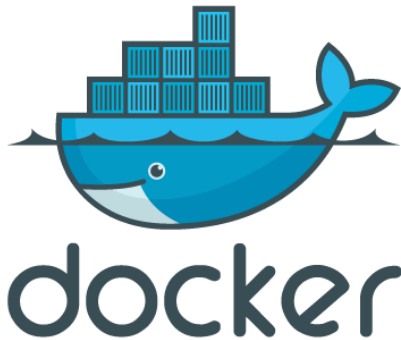
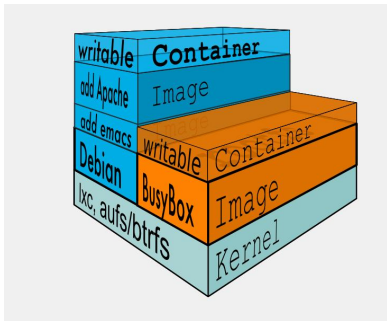
why docker

docker 的基本
内容

docker 的安装

docker 的使用
实例

docker 集群



docker 与传统虚拟机的区别

docker 技术分

享

冯海雄

目录

docker 是什么

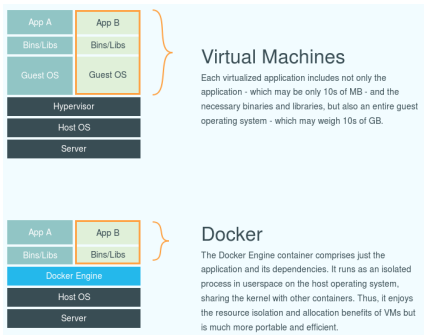
why docker

docker 的基本
内容

docker 的安装

docker 的使用
实例

docker 集群



为什么使用 docker

docker 技术分
享

冯海雄

目录

docker 是什么

why docker

docker 的基本
内容

docker 的安装

docker 的使用
实例

docker 集群

1. docker 可以提供一个轻量级的程序运行环境
2. docker 可以简化程序的安装部署
3. docker 可以提高硬件资源的利用率
4. docker 可以方便管理减轻运维压力

docker 的常用命令

docker 技术分

享

冯海雄

目录

docker 是什么

docker 的基本内容

docker 常用命令概览

docker 的镜像

docker 容器

docker 的仓库

用 docker 共享我们的工作

docker 的特点

docker 的安装

docker 的使用实例

- attach Attach to a running container
- build Build an image from a Dockerfile
- commit Create a new image from a container
- cp Copy files/folders from a container
- images List images
- kill Kill a running container
- load Load an image from a tar archive
- logs Fetch the logs of a container
- port Lookup the public-facing port that is NAT-ed to PRIVATE_PORT
- ps List containers
- pull Pull an image or a repository from server
- push Push an image or a repository to server
- restart Restart a running container
- rm Remove one or more containers
- rmi Remove one or more images
- run Run a command in a new container
- save Save an image to a tar archive
- start Start a stopped container
- stop Stop a running container
- tag Tag an image into a repository

镜像

docker 技术分

享

冯海雄

目录

docker 是什么

docker 的基本
内容

docker 常用命令概览

docker 的镜像

docker 容器

docker 的仓库

用 docker 共享我们的
工作

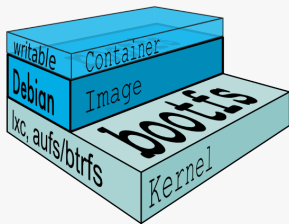
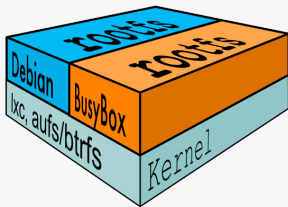
docker 的特点

docker 的安装

docker 的使用

实例

镜像是一个只读的容器模板,一个镜像可以实例化多个容器,Union file systems, or UnionFS, are file systems that operate by creating layers, making them very lightweight and fast.



获得镜像

docker 技术分

享

冯海雄

目录

docker 是什么

docker 的基本
内容

docker 常用命令概览

docker 的镜像

docker 容器

docker 的仓库

用 docker 共享我们的
工作

docker 的特点

docker 的安装

docker 的使用
实例

docker 上的一切操作都基于镜像, 获得镜像有以下几种方法

1. 用 docker pull 从 docker Hub 获取
2. 基于当前系统构建基础镜像
3. 使用 dockerfile 构建镜像
4. 使用 commit 将容器固化成镜像
5. tag 为镜像打上不同的标签

容器

docker 技术分

享

冯海雄

目录

docker 是什么

docker 的基本
内容

docker 常用命令概览

docker 的镜像

docker 容器

docker 的仓库

用 docker 共享我们的
工作

docker 的特点

docker 的安装

docker 的使用
实例

容器是一个动态的运行进程，容器依赖于指定的镜像，容器可以看作一个轻量级动态镜像

- docker run 启动容器
- docker ps 列出正在运行的容器
- docker logs 查看指定容器的日志
- docker stop 停止一个正在运行的容器
- docker start 启动一个被启动过的容器

容器的网络设置

docker 技术分

享

冯海雄

目录

docker 是什么

docker 的基本
内容

docker 常用命令概览

docker 的镜像

docker 容器

docker 的仓库

用 docker 共享我们的
工作

docker 的特点

docker 的安装

docker 的使用
实例

- host 模式, 使用 `-net=host` 指定。
- container 模式, 使用 `-net=container:name`
- none 模式, 使用 `-net=none` 指定。
- bridge 模式, 使用 `-net=bridge` 指定, 默认设置。
- 容器中的 `/etc/hosts` 可以通过 `-add-hosts` 来添加内容
- 容器可以通过 `-dns` 来指定 dns 服务器

容器的重启策略

docker 技术分

享

冯海雄

目录

docker 是什么

docker 的基本
内容

docker 常用命令概览

docker 的镜像

docker 容器

docker 的仓库

用 docker 共享我们的
工作

docker 的特点

docker 的安装

docker 的使用
实例

-restart 选项可以设置容器的重启策略

- no 表示不重启
- onfailure 表示容器如果以失败状态结束则重启容器
- always 表示只要退出就重启容器
- -rm 选项表示运行完毕就删除容器
- -rm 和 -restart 参数不能同时设置

容器对 cpu 和内存的配置

docker 技术分

享

冯海雄

目录

docker 是什么

docker 的基本
内容

docker 常用命令概览

docker 的镜像

docker 容器

docker 的仓库

用 docker 共享我们的
工作

docker 的特点

docker 的安装

docker 的使用
实例

- `-m=""`: Memory limit (format: `<number><optional unit>`, where unit = b, k, m or g)
- `-memory-swap=""`: Total memory limit (memory + swap, format: `<number><optional unit>`, where unit = b, k, m or g)
- `-c, -cpu-shares=0` CPU shares (relative weight)

容器的日志

docker 技术分

享

冯海雄

目录

docker 是什么

docker 的基本
内容

docker 常用命令概览

docker 的镜像

docker 容器

docker 的仓库

用 docker 共享我们的
工作

docker 的特点

docker 的安装

docker 的使用
实例

docker logs 可以查看指定容器的日志 -log-driver 选项可以指定容器的日志格式

- -log-driver= none 禁用日志, 容器运行过程中不会有日志输出
- -log-driver= json-file 容器日志以 json 格式的形式输出, 默认为 json 格式
- -log-driver= syslog 容器日志直接以系统日志的形式记录, 这种情况下 docker logs 指令不可用。

关联容器在一起

docker 技术分

享

冯海雄

目录

docker 是什么

docker 的基本
内容

docker 常用命令概览

docker 的镜像

docker 容器

docker 的仓库

用 docker 共享我们的工作

docker 的特点

docker 的安装

docker 的使用
实例

-link 可以将运行了不同服务的容器关联在一起容器可以使用被关联过来的容器的服务实例如下：

1. 启动一个运行了 mysql 的容器

- `sudo docker start sqlubuntu`

2. 新建一个关联了 mysql 容器的终端容器

- `sudo docker run -it -rm -link sqlubuntu:db base:6.6 /bin/bash`

3. 通过新建的终端访问容器中的数据库

- `mysql -h db -uroot -p"letmein" -e"show databases"`

容器中的数据管理

docker 技术分

享

冯海雄

目录

docker 是什么

docker 的基本
内容

docker 常用命令概览

docker 的镜像

docker 容器

docker 的仓库

用 docker 共享我们的
工作

docker 的特点

docker 的安装

docker 的使用

实例

在启动容器时 `-v` 选项可以添加数据卷到容器

1. 数据卷数据卷通过 `unionfs` 实现数据的共享机制

- 数据卷在容器创建的时候初始化
- 数据卷可以在容器之间共享和复用
- 数据卷的变化不会写入到镜像中去
- 容器消失之后数据卷上的数据不会被删除

2. 数据卷容器是一个添加了数据卷的容器, 这样的容器可以被其他容器共享

3. 数据卷容器共享的数据目录对所有的容器开放, 在操作数据卷时容器中的 `root` 用户具有和系统 `root` 用户一样的超级权限

仓库

docker 技术分

享

冯海雄

目录

docker 是什么

docker 的基本
内容

docker 常用命令概览

docker 的镜像

docker 容器

docker 的仓库

用 docker 共享我们的
工作

docker 的特点

docker 的安装

docker 的使用

实例

仓库用来存储和分享镜像

1. 公有仓库 Dockerhub

- `docker pull ubuntu:latest` 从 DockerHub 获取镜像
- `docker push base:6.6` 推送镜像至 Dockerhub

2. 私有仓库可使用时需要指定镜像标签为私有库地址否则会推送到公有仓库

docker 镜像的备份与还原

docker 技术分

享

冯海雄

目录

docker 是什么

docker 的基本
内容

docker 常用命令概览

docker 的镜像

docker 容器

docker 的仓库

用 docker 共享我们的
工作

docker 的特点

docker 的安装

docker 的使用
实例

1. docker save 可以把镜像保存到本地的归档文件中

- `sudo docker save base >base.tar`

2. docker load 可以加载保存好的归档文件

- `sudo docker load <base.tar`

dockerfile

docker 技术分
享

冯海雄

目录

docker 是什么

docker 的基本
内容

docker 常用命令概览

docker 的镜像

docker 容器

docker 的仓库

用 docker 共享我们的
工作

docker 的特点

docker 的安装

docker 的使用
实例

dockerfile 是一种便捷的镜像分享的方式
dockerfile 有自己的编写规则：

- FROM 指定以来的镜像
- MAINTAINER 命令用来指定维护者的姓名和联系方式
- RUN 命令在 build 阶段执行命令
- COPY 在 build 阶段拷贝文本本地文件到容器中
- USER 指定容器中运行服务的用户
- EXPOSE 设置容器运行时开放的端口
- CMD 设置容器运行时启动的程序
- ENTRYPOINT 设置容器的入口与 CMD 类似

docker 的特点

docker 技术分

享

冯海雄

目录

docker 是什么

docker 的基本
内容

docker 常用命令概览

docker 的镜像

docker 容器

docker 的仓库

用 docker 共享我们的
工作

docker 的特点

docker 的安装

docker 的使用
实例

- 将应用和系统“容器化”，不添加额外的操作系统
- 你会拥有足够的“操作系统”，仅需添加或减小镜像即可。
在一台服务器上可以布署 100 1000 个 Containers 容器。
- 开源的, 免费的, 低成本的
- docker 可以使部署变得简单
- 缩短软件的开发周期

docker 的局限

docker 技术分

享

冯海雄

目录

docker 是什么

docker 的基本
内容

docker 常用命令概览

docker 的镜像

docker 容器

docker 的仓库

用 docker 共享我们的
工作

docker 的特点

docker 的安装

docker 的使用
实例

Docker 与其他虚拟化产品相比有以下几点差异：

- Docker 是基于 Linux 64bit 的, 无法在 windows/unix 或 32bit 的 linux 环境下使用
- Docker container 的 guest 系统只能是 linux base 的目前没有办法在 docker 中启动一个 window 程序
- 隔离性相比 KVM 之类的虚拟化方案还是有些欠缺, 所有 container 公用一部分的运行库
- 网络管理相对简单, 主要是基于 namespace 隔离

在 centos 上安装 docker

docker 技术分

享

冯海雄

目录

docker 是什么

docker 的基本
内容

docker 的安装

docker 的安装部署

docker 的使用
实例

docker 集群

- 目前只支持 centos6.5 及 centos7
- 内核要求 2.6.32-431 以及这个版本之上的内核
- 只支持 64 位的系统, docker 不能运行在 32 位的系统上
- centos6.5 上安装 docker
- `sudo yum -y install docker-io`
- 启动 docker 服务 `sudo service docker start`

在 ubuntu 上安装 docker

docker 技术分

享

冯海雄

目录

docker 是什么

docker 的基本
内容

docker 的安装

docker 的安装部署

docker 的使用
实例

docker 集群

- docker 支持的 ubuntu 系统有
- Ubuntu Trusty 14.04 (LTS)
- Ubuntu Precise 12.04 (LTS)
- Ubuntu Saucy 13.10
- 要求内核不低于 3.10 系统必须是 64 位系统 **
- `sudo apt-get install wget`
- `wget -qO- https://get.docker.com/ | sh`
- `sudo docker run hello-world`

使用容器生成 spark 客户端镜像

docker 技术分

享

冯海雄

目录

docker 是什么

docker 的基本
内容

docker 的安装

docker 的使用
实例

docker 集群

- 启动容器
- 在启动的容器中安装 spark 客户端
- 测试 spark 客户端运行正常
- 退出容器
- 将 spark 客户端容器 commit 成镜像

dockerfile 示例

docker 技术分

享

冯海雄

目录

docker 是什么

docker 的基本
内容

docker 的安装

docker 的使用
实例

docker 集群

- FROM ubuntu
- MAINTAINER fenghaixiong <fenghaixiong@nd.com.cn>
- RUN apt-get install -y mysql-server
- EXPOSE 3306
- USER mysql
- ENTRYPOINT /etc/init.d/mysql start

Docker Machine

docker 技术分

享

冯海雄

目录

docker 是什么

docker 的基本
内容

docker 的安装

docker 的使用
实例

docker 集群

Docker Machine 是一个基于现有虚拟机创建 docker 服务器的 docker 服务器管理工具, Docker Machine 还处于孵化期发展很不成熟。

docker machine 可以基于本地虚拟机如 virtualbox 构建 docker 服务器

docker machine 也可以使用云平台构建 docker 服务器

Docker Swarm

docker 技术分

享

冯海雄

目录

docker 是什么

docker 的基本
内容

docker 的安装

docker 的使用
实例

docker 集群

Swarm 是 docker 的分布式集群管理工具, 可以将多个

Docker 服务器虚拟成一个 docker 服务器基本使用方法如下:

- `docker pull swarm` 获取 swarm 镜像
- `docker run -rm swarm create` 创建集群
- 注册节点
 - `docker -H tcp://0.0.0.0:2375 -d`
 - `docker run -d swarm join -addr=<node_ip:2375>
token://<cluster_id>`
- `docker run -d -p <swarm_port>:2375 swarm manage
token://`
- 查看集群 `docker -H
tcp://<manager_ip:manager_port> info 、ps logs、run`
- 列出集群中的节点 `docker run -rm swarm list
token://<cluster_id>`

Docker compose

docker 技术分

享

冯海雄

目录

docker 是什么

docker 的基本
内容

docker 的安装

docker 的使用
实例

docker 集群

docker Compose 可以复杂的应用定义在一个脚本中, 使用
docker Compose 主要分为三步:

- 用 `dockerfile` 定义应用的环境
- 编辑 `docker-compose.yml` 文件将多个应用组合在一起
- `run docker-compose up` 执行 Docker Compose 脚本
- `docker-compose up -d` 可以将应用作为后台程序运行
- `docker-compose up stop` 可以停止正在运行的容器

docker-compose.yml

docker 技术分

享

冯海雄

目录

docker 是什么

docker 的基本
内容

docker 的安装

docker 的使用
实例

docker 集群

web:

build: .

command: python app.py

ports:

- "5000:5000"

volumes:

- ./code

links:

- redis

redis:

image: redis