Docker Compose 是 Docker 官方编排项目之一，负责快速的部署分布式应用。

[**#**](https://pig4cloud.com/#%E5%AE%89%E8%A3%85docker-%EF%BC%88centos7%EF%BC%89) **安装Docker （centos7）**

# 更新yum 源

yum update

#安装 Docker

yum -y install docker

#启动 Docker 后台服务

service docker start

#测试运行 hello-world,由于本地没有hello-world这个镜像，所以会下载一个hello-world的镜像，并在容器内运行。

docker run hello-world

[**#**](https://pig4cloud.com/#%E5%AE%89%E8%A3%85docker-compose) **安装docker-compose**

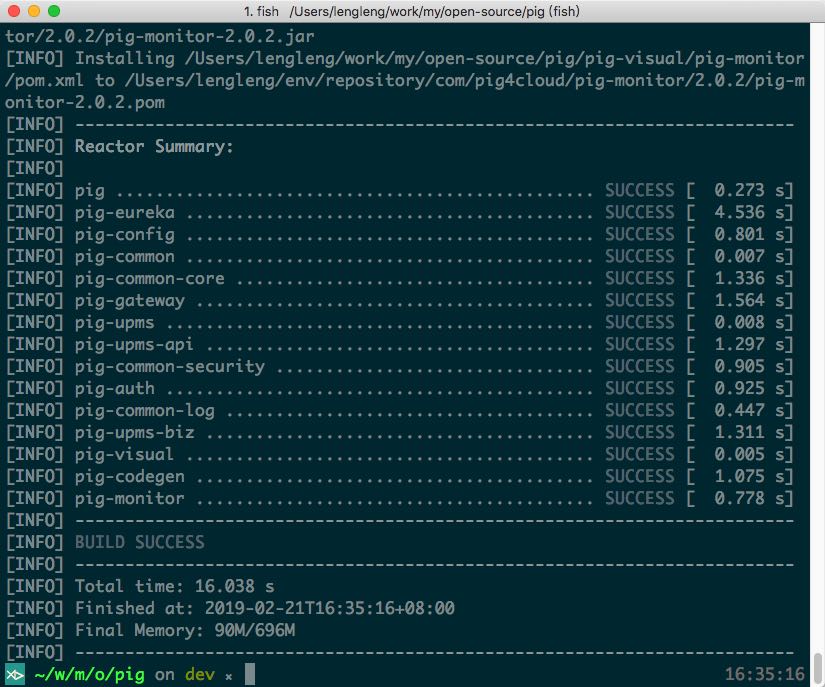
$ sudo curl -L https://github.com/docker/compose/releases/download/1.20.1/docker-compose-`uname -s`-`uname -m` > /usr/local/bin/docker-compose

$ sudo chmod +x /usr/local/bin/docker-compose

[**#**](https://pig4cloud.com/#pig-%E6%89%93%E5%8C%85) **pig 打包**

* pig 根目录

mvn clean install -Dmaven.test.skip=true



* 压缩pig 整个工程上传到docker 宿主机
* 执行 docker-compose 命令

# 构建镜像

docker-compose build

# 启动容器 （-d 后台启动，建议第一次不要加，方便看错误）

docker-compose up -d

[**#**](https://pig4cloud.com/#%E7%AD%89%E5%BE%853%E5%88%86%E9%92%9F) **等待3分钟**

访问Centos7 IP:8761 查看eureka状态，确定所有服务全部启动。

[**#**](https://pig4cloud.com/#%E6%80%BB%E7%BB%93) **总结**

1. 服务端已启动完毕，前端请参考下一章节《前端部署》
2. 不要和开发环境一样，修改容器hosts,docker-compose 会根据容器名称自动处理