

Wefe 项目以 Docker 容器的方式启动，需要依赖于 Docker 环境。

Docker 运行操作需要 root 用户权限。

## 主机环境

- CentOS 7 +。
- 防火墙关闭或开启对应的容器开放端口（具体端口在 `wefe.cfg` 文件中进行检查与修改）。

```
1 systemctl stop firewalld.service
2 # 重启 docker
3 systemctl restart docker
```

- 关闭 `setLinux`。

```
1 vim /etc/selinux/config
2 # 将其中 SELINUX 设置为 disable
```

## Docker 环境安装

```
1 ### 本机 Docker 卸载
2 yum remove docker \
3     docker-client \
4     docker-client-latest \
5     docker-common \
6     docker-latest \
7     docker-latest-logrotate \
8     docker-logrotate \
9     docker-engine
10 rpm -qa |grep docker
11 yum list installed | grep docker
12
13 yum remove -y containerd.io.x86_64 \
14     docker-ce.x86_64 docker-ce-cli.x86_64
15
16 ### Docker 安装
17 # 复制后注意检查粘贴后的代码是否正确
```

```
18 yum install -y \  
19 yum-utils device-mapper-persistent-data lvm2  
20  
21 yum-config-manager \  
22 --add-repo http://mirrors.aliyun.com\  
23 /docker-ce/linux/centos/docker-ce.repo  
24  
25 yum install -y https://mirrors.aliyun.com/  
26 docker-ce/linux/centos/7/x86_64/edge/  
27 Packages/containerd.io-1.2.13-3.1.e17.x86_64.rpm  
28  
29 yum install docker-ce docker-ce-cli containerd.io
```

## Docker-Compose 环境安装

---

```
1 # 下载对应版本的官方包  
2 wget  
  https://github.com/docker/compose/releases/download/1.2  
  7.4/docker-compose-Linux-x86_64  
3 mv docker-compose-Linux-x86_64 /usr/local/bin/docker-  
  compose  
4 # 添加执行权限  
5 chmod +x /usr/local/bin/docker-compose  
6 # 检测是否安装正确  
7 docker-compose -version
```

## 项目配置

---

```
1 ### spark 配置
2 # driver 进程使用的内存数, 推荐 2g 以上。参考: 100w 数据 (30
  特征), 参考配置 5g
3 DRIVER_MEMORY="15g"
4 # executors 数量, 生产推荐 12 以上
5 NUM_EXECUTORS="6"
6 # 每个 executor 进程使用的内存数, 推荐 2g 以上
7 EXECUTOR_MEMORY="2g"
8 # 每个 executor 分配的核心数, 建议2
9 EXECUTOR_CORES="2"
10 # 数据分片数, 参考配置: NUM_EXECUTORS*EXECUTOR_CORES 的整数
    倍, 100w 以下数据建议 64 片以下
11 NUM_SLICES="128"
```

## 项目启动

解压 jar 包到同一个文件夹下, 解压后的目录如下所示。

```
1 - board
2 - docker-start.sh
3 - flow
4 - gateway
5 - middleware
6 - sh
7 - wefe.cfg
8 - wefe-stop.sh
```

编辑 `wefe.cfg` 文件进行配置的修改。

运行命令 `sh docker-start.sh` 启动项目。

**注意, 配置中的端口不可与本机已占用的端口发生冲突, 若发生端口冲突导致启动失败, 则需要停止移除所有容器, 重新启动服务。**

```
1 ### 启动后在部署的服务器上运行
2 docker inspect $(docker ps -q) | \
3 grep "\"IPAddress\": \"1\" | \
4 awk '{print $2}' | tr '"' ' ' | \
5 sed s/[[:space:]]//g | sed "s/\./\./g" | \
6 uniq | tr '\n' ' ' | sed s/[[:space:]]//g
7 ### 将输出的一串 IP 地址复制粘贴到项目的 全局设置 --> 系统配置 -
-> 网关 IP 地址白名单中
```

注意，以上添加 IP 白名单操作在每一次容器重启之后都需要进行。

运行命令 `sh wefe-stop.sh` 停止项目。

## Docker 配置

```
1 ### /etc/docker/daemon.json
2 {
3     # 默认将 docker 产生的文件放在此目录下
4     "graph": "/data/wefe/docker-compose",
5     "live-restore": true
6 }
```

## 访问项目

- 访问 `http://[主机 IP]/board-website` 进入项目登陆界面。
- 第一个注册的用户为此联邦成员的超级管理员。
- 首次登录需要初始化联邦成员的信息：
  - Board Uri: `http://[主机 IP]/board-service`
- 在全局设置 --> 系统配置 --> 网关 IP 地址白名单中根据上面的操作进行白名单的更新。
- 在全局设置 --> member 信息 --> 填写 Gateway Uri: `[主机 IP]:50051` (若修改了端口配置, 则需要同步改动)。
- 右上角的帮助文档可帮助您了解基本的项目操作。