

自动化卓越中心 v1.0.0 部署文档

更新记录

2021.02.04 - 创建文档，编写后端镜像打包，启动步骤及示例。

项目描述

自动化需求的全生命周期管理平台，帮您轻松掌握每个需求的实现情况及追踪每个需求的状态转，帮助 COE 部门高效管理企业内部的所有自动化需求。

部署步骤

文件清单

文件名	描述	MD5 值
ac-api.tar.gz	后端服务导出后的镜像文件	
nginx.tar.gz	Nginx 服务导出后的镜像文件	
ac-web.tar.gz	前端服务源码文件	
config.yaml	后端服务配置文件，需根据实际需求进行修改	
docker-compose.yaml	docker-compose 配置文件	
init.sql	数据初始化文件	

导入镜像

使用以下命令将镜像文件进行导入

```
docker load < ac-api.tar.gz
docker load < nginx.tar.gz
```

命令执行完成后使用 docker images 命令查看镜像是否存在

```
docker images
```

将 config.yaml 文件拷贝至 /var/www/ac/ 目录并进行修改，以下配置为示例：

```
# APP
APP_ENV: # app 环境，可以填写 local, stage, production
SECRET_KEY: # 安全 key，字符串即可，长度不限
ADMIN: # 超级管理员账号名称，该账号需要在中控中存在
JSON_SORT_KEYS: False # 是否对接口返回数据保序，必须为 False
SERVER_ADDRESS: http://127.0.0.1:5000 # 后端服务地址
```

```
FRONTEND_SERVER_ADDRESS: http://127.0.0.1:8081 # 前端服务地址
USER_LOGIN_EXPIRED_TIME: !!int 86400 # 用户登录超时时间，默认24h

# MYSQL
SQLALCHEMY_DATABASE_URI: # 数据库地址，格式为 mysql+pymysql://用户名:密码@数据库地址:数据库端口/数据库名称

# RPA
RPA_USERNAME: # 同步中控数据账号，该账号需要能获取所有用户和组织得到数据
RPA_PASSWORD: # 同步中控数据密码，该账号需要能获取所有用户和组织得到数据
RPA_URL: # 中控接口地址

# SCHEDULER
SCHEDULER_JOB_RUN_INTERVAL: # 同步中控数据事件间隔，单位分钟

# SENTRY
SENTRY_URL: # sentry 告警地址，只有在 APP_ENV = stage 或 production 条件下才启用，可以不填写

# EMAIL
MAIL_USERNAME: # smtp 账号
MAIL_PASSWORD: # smtp 密码
SMTP_HOST: # smtp 地址
SMTP_PORT: # smtp 端口
```

启动服务

调整 docker-compose.yml 配置文件中的相关配置

```
version: "3"
services:
  backend:
    image: nexus-docker.test.cyclone.com/cyclone-web-python/ac:v1.0.0
    ports:
      - "9091:5000"
    container_name: ac-api
    restart: always
    volumes:
      - /var/log/ac:/app/ac/logs/
      - /var/www/ac/config.yaml:/app/ac/config.yaml
      - /var/www/ac/static:/app/ac/ac/static/
      - /etc/resolv.conf:/etc/resolv.conf # cyclone test environment
      - /etc/hosts:/etc/hosts # cyclone test environment needs to be added
    command: [ 'sh', 'docker-entrypoint.sh' ]

  frontend:
    image: nexus-docker.test.cyclone.com/cyclone-web-python/nginx:1.19.6
    ports:
      - "9092:80"
    container_name: ac-web
    restart: always
    volumes:
      - /var/www/ac/dist:/usr/share/nginx/html
```

在 docker-compose.yml 文件所在路径执行 docker-compose up -d 启动服务，并使用 docker ps 查看容器状态。

```
docker-compose up -d
docker ps
```

注意事项

1. 日志归档：ac 会将日志按照日期进行切割，并映射至宿主机中上述配置文件所配置的 /var/log/ac/ 目录。如若需要日志归档，则对映射出来的目录进行归档即可。
2. 上传文件：上传的文件会映射至宿主机中 /var/www/ac/static/ 目录。若有必要请及时对文件目录进行备份。
3. 服务健康探测：ac 提供了健康检查接口，url 地址为 /api/v1/health。

常见问题

1. 服务无法启动：可先执行 docker-compose up，查看错误日志。问题修复再使用 docker-compose up -d 启动项目。
2. 服务启动后接口错误：可参考映射出来的日志文件，查看具体错误信息。
3. 数据库中没有表结构，需要同步表结构到数据库中。

1. 进入容器
2. 执行以下命令，同步表结构到数据库中

```
flask db init
flask db migrate
flask db upgrade
```

4. 初始化数据：内置权限等初始化数据已放在 init.sql 文件中，创建表结构后需要导入初始化数据文件。