**阿凡搭企业数字化底座平台**

**部署手册**

深圳市非凡信息技术有限公司

本文档若未及时更新或对软件说明有不完整之处，则以实际交付的软件产品所具备的功能为准。本文档的版权和最终解释权归深圳市非凡信息技术有限公司所有。

**目录**

[1 部署架构 4](#_Toc9937)

[1.1 单服务器架构 4](#_Toc7908)

[1.1.1 部署架构图 4](#_Toc30627)

[1.1.2 服务器清单 5](#_Toc6002)

[1.2 多服务器架构 5](#_Toc13071)

[1.2.1 部署架构图 6](#_Toc17289)

[1.2.2 服务器清单 6](#_Toc10968)

[1.3 软件负载均衡架构 7](#_Toc8868)

[1.3.1 部署架构图 8](#_Toc2630)

[1.3.2 服务器清单 8](#_Toc7612)

[1.4 硬件负载均衡架构 9](#_Toc7674)

[1.4.1 部署架构图 10](#_Toc24655)

[1.4.2 服务器清单 10](#_Toc14161)

[2 部署方式 11](#_Toc25274)

[2.1 私有化本地部署 11](#_Toc28639)

[2.1.1 服务器目录结构（建议） 11](#_Toc30787)

[2.1.2 应用组件安装 12](#_Toc30157)

[2.1.3 平台部署 27](#_Toc1665)

[2.2 私有化容器部署 38](#_Toc4126)

[2.2.1 在线部署 38](#_Toc30987)

[2.2.2 离线部署 41](#_Toc16936)

[3 系统初始化 44](#_Toc23442)

[3.1 设置系统超级管理员 44](#_Toc28611)

[3.2 创建组织机构及用户 45](#_Toc5584)

[3.3 创建系统功能模块 48](#_Toc12282)

[3.4 测试应用转生产环境 50](#_Toc24566)

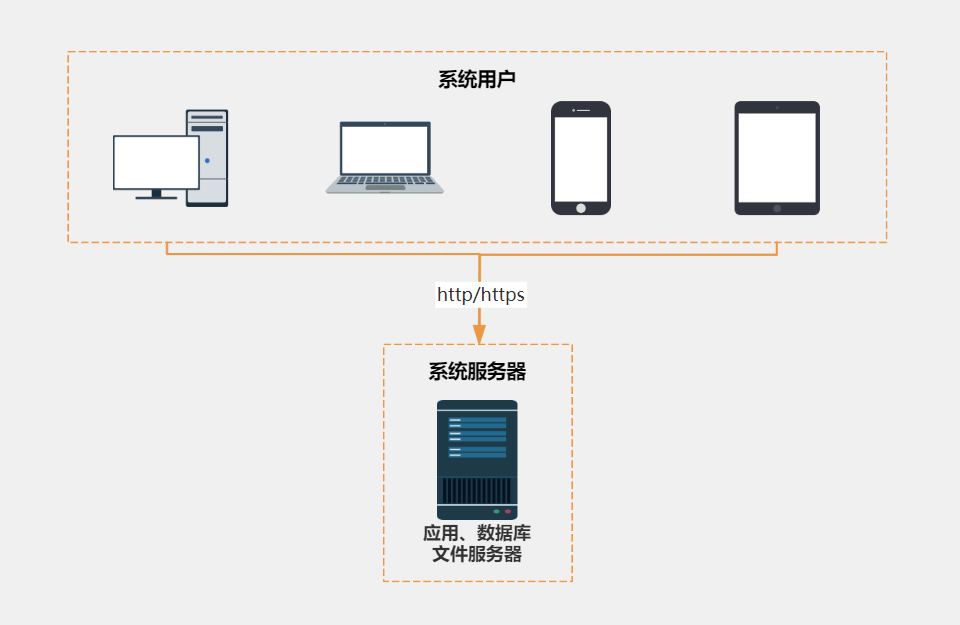
# 部署架构

## 单服务器架构

阿凡搭平台单服务器部署架构模式是将平台所有的应用程序、组件和服务都部署在同一个服务器上。服务器上需完成以下工作：

1. 安装阿凡搭平台部署所需的所有组件，包括数据库和文件存储服务；
2. 部署阿凡搭平台引擎服务；
3. 部署阿凡搭平台PC端应用前后端；
4. 部署阿凡搭平台移动端应用前后端；

### 部署架构图



### 服务器清单

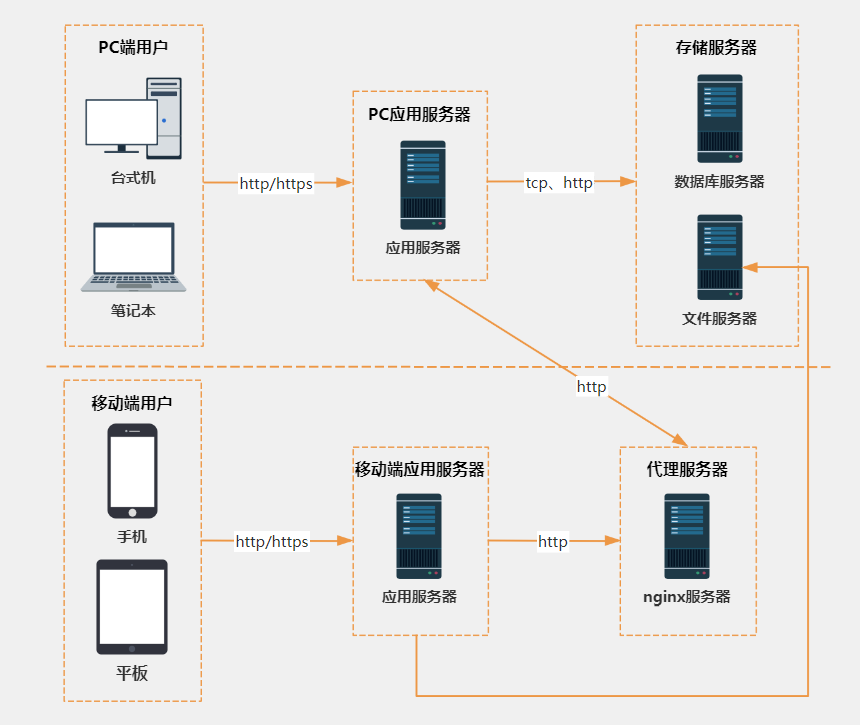
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **服务器名称** | **操作系统** | **建议配置** | | |
| **CPU** | **内存** | **存储** |
| 阿凡搭平台服务器 | CentOS7+/Redhat8+/银河麒麟v10 | 16C | 32G | 1T |

## 多服务器架构

阿凡搭平台多服务器部署架构模式是指将平台应用程序、数据及文件存储分不同的服务器来部署。服务器上需执行以下操作：

* PC端应用服务器上部署阿凡搭平台PC端应用前后端和引擎服务；
* 移动端应用服务器上需部署阿凡搭平台移动端应用前后端；
* 数据库服务器上安装关系数据库mysql并存储阿凡搭平台数据；
* 文件服务器上安装文件存储服务并存储阿凡搭平台的文件；

### 部署架构图



### 服务器清单

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **服务器名称** | **操作系统** | **建议配置** | | |
| **CPU** | **内存** | **存储** |
| PC应用服务器 | CentOS7+/Redhat8+/银河麒麟v10 | 8C | 16G | 200G |
| 数据库服务器 | CentOS7+/Redhat8+/银河麒麟v10 | 8C | 16G | 500G |
| 文件服务器 | CentOS7+/Redhat8+/银河麒麟v10 | 8C | 16G | 500G |
| 移动端应用服务器 | CentOS7+/Redhat8+/银河麒麟v10 | 8C | 16G | 200G |
| 代理服务器 | CentOS7+/Redhat8+/银河麒麟v10 | 4C | 8G | 100G |

说明：

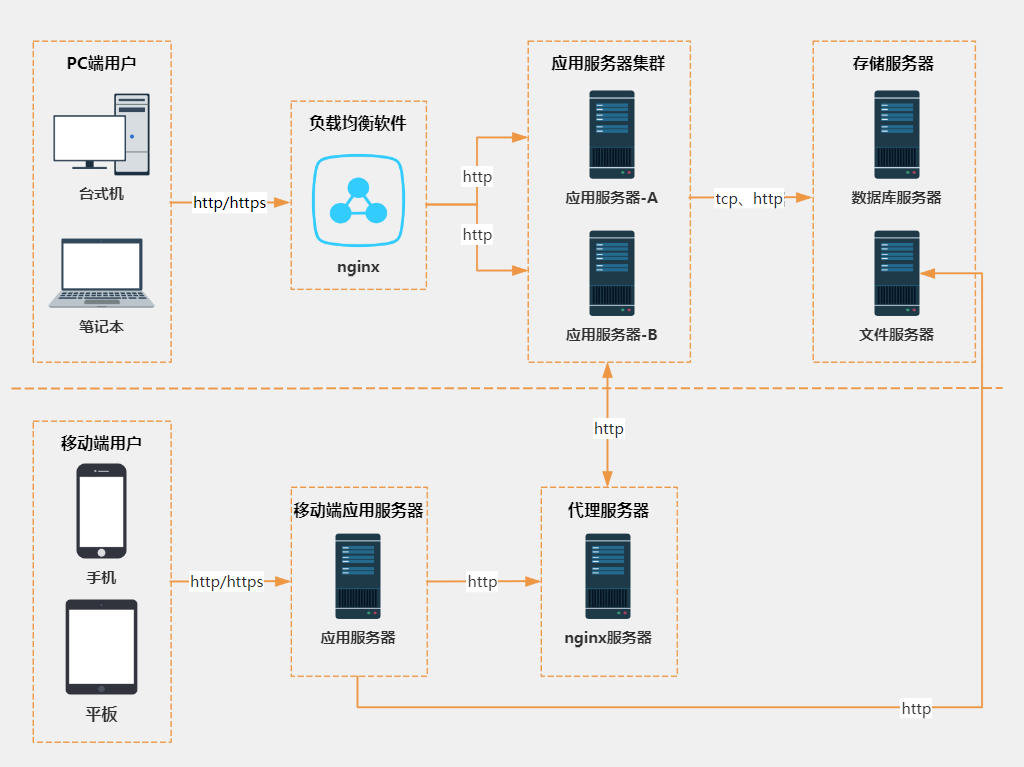
1. 文件服务器为非必须配备，根据实际情况配备，文件存储服务可安装在应用服务器或数据库服务器上。
2. 移动端应用服务器的配备是根据是否有移动端实施需求确定的，移动端应用服务器需开通外网访问策略，如果移动端接入的是微信公众号则需开通80端口；
3. 代理服务器为非必须配备，代理服务器的作用是：当企业有内外网络隔离的策略时，代理服务器可用来建立内外网络之间的通信，移动端与PC端之间的交互及存取数据。

## 软件负载均衡架构

阿凡搭平台软件负载均衡架构模式是指将平台应用程序、数据及文件存储分不同的服务器来部署，部署多套应用程序在不同应用服务器上，应用服务器支持弹性扩展，由nginx软件来实现负载均衡。服务器上需执行以下操作：

* PC端应用服务器上部署阿凡搭平台PC端应用前后端和引擎服务；
* 移动端应用服务器上需部署阿凡搭平台移动端应用前后端；
* 数据库服务器上安装关系数据库mysql并存储阿凡搭平台数据；
* 文件服务器上安装文件存储服务并存储阿凡搭平台的文件；

### 部署架构图



### 服务器清单

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **服务器名称** | **操作系统** | **建议配置** | | |
| **CPU** | **内存** | **存储** |
| 负载均衡服务器 | CentOS7+/Redhat8+/银河麒麟v10 | 8C | 16G | 200G |
| 应用服务器-A | CentOS7+/Redhat8+/银河麒麟v10 | 8C | 16G | 200G |
| 应用服务器-B | CentOS7+/Redhat8+/银河麒麟v10 | 8C | 16G | 200G |
| 数据库服务器 | CentOS7+/Redhat8+/银河麒麟v10 | 8C | 16G | 500G |
| 文件服务器 | CentOS7+/Redhat8+/银河麒麟v10 | 8C | 16G | 500G |
| 移动端应用服务器 | CentOS7+/Redhat8+/银河麒麟v10 | 8C | 16G | 200G |
| 代理服务器 | CentOS7+/Redhat8+/银河麒麟v10 | 4C | 8G | 100G |

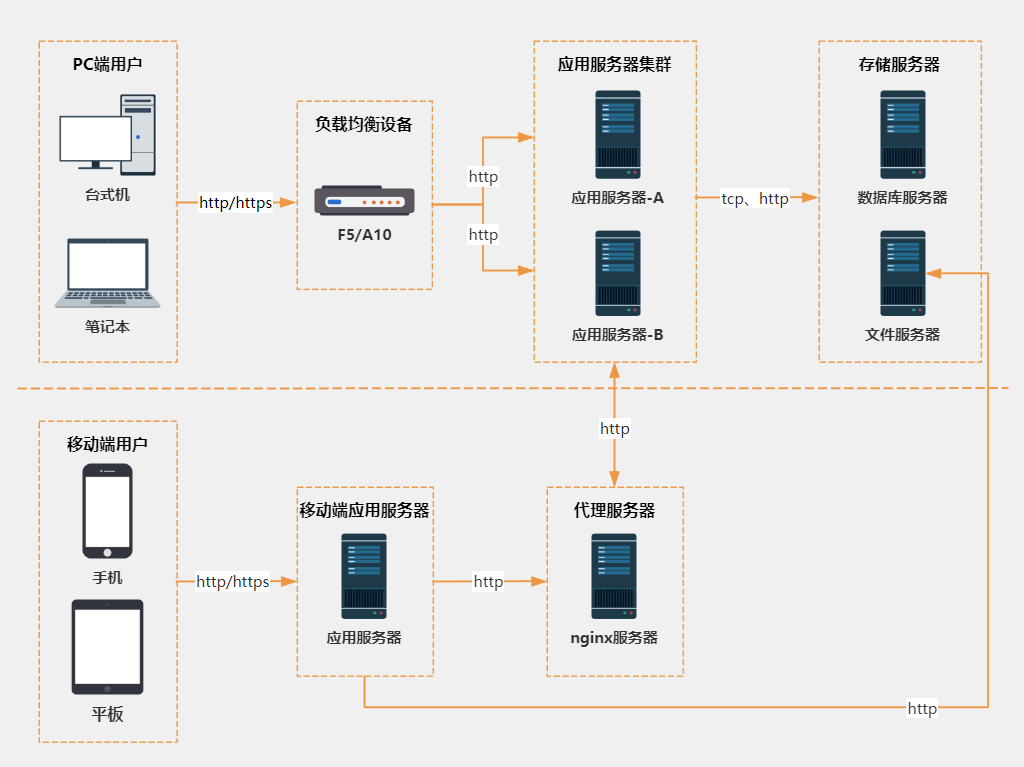
1. 负载均衡服务器为非必须配备，负载均衡服务器主要是用来安装负载均衡软件nginx，负载均衡软件可安装在应用服务器上；
2. 应用服务器数量至少两台，可支持弹性扩展；
3. 文件服务器为非必须配备，根据实际情况配备，文件存储服务可安装在应用服务器或数据库服务器上；
4. 移动端应用服务器的配备是根据是否有移动端实施需求确定的，移动端应用服务器需开通外网访问策略，如果移动端接入的是微信公众号则需开通80端口；
5. 代理服务器为非必须配备，代理服务器的作用是：当企业有内外网络隔离的策略时，代理服务器可用来建立内外网络之间的通信，移动端与PC端之间的交互及存取数据。

## 硬件负载均衡架构

阿凡搭平台硬件负载均衡架构模式是指将平台应用程序、数据及文件存储分不同的服务器来部署，部署多套应用程序在不同应用服务器上，应用服务器支持弹性扩展，由负载均衡设备F5或A10来实现负载均衡。服务器上需执行以下操作：

* PC端应用服务器上部署阿凡搭平台PC端应用前后端和引擎服务；
* 移动端应用服务器上需部署阿凡搭平台移动端应用前后端；
* 数据库服务器上安装关系数据库mysql并存储阿凡搭平台数据；
* 文件服务器上安装文件存储服务并存储阿凡搭平台的文件；

### 部署架构图



### 服务器清单

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **服务器名称** | **操作系统** | **建议配置** | | |
| **CPU** | **内存** | **存储** |
| 应用服务器-A | CentOS7+/Redhat8+/银河麒麟v10 | 8C | 16G | 200G |
| 应用服务器-B | CentOS7+/Redhat8+/银河麒麟v10 | 8C | 16G | 200G |
| 数据库服务器 | CentOS7+/Redhat8+/银河麒麟v10 | 8C | 16G | 500G |
| 文件服务器 | CentOS7+/Redhat8+/银河麒麟v10 | 8C | 16G | 500G |
| 移动端应用服务器 | CentOS7+/Redhat8+/银河麒麟v10 | 8C | 16G | 200G |
| 代理服务器 | CentOS7+/Redhat8+/银河麒麟v10 | 4C | 8G | 100G |

说明：

1. 应用服务器数量至少两台，可支持弹性扩展；
2. 文件服务器为非必须配备，根据实际情况配备，文件存储服务可安装在应用服务器或数据库服务器上；
3. 移动端应用服务器的配备是根据是否有移动端实施需求确定的，移动端应用服务器需开通外网访问策略，如果移动端接入的是微信公众号则需开通80端口；
4. 代理服务器为非必须配备，代理服务器的作用是：当企业有内外网络隔离的策略时，代理服务器可用来建立内外网络之间的通信，移动端与PC端之间的交互及存取数据。

# 部署方式

阿凡搭平台可基于两种方式部署：本地服务器直接部署和本地服务器容器化部署。

## 私有化本地部署

### 服务器目录结构（建议）

|  |  |  |  |
| --- | --- | --- | --- |
| **一级目录** | **二级目录** | **三级目录** | **四级目录** |
| home | ffapaas | web |  |
| back | jdk |
| tomcat |
| engine | jdk |
| tomcat |
| upload |  |
| mobile | web |  |
| back | jdk |
| tomcat |
| software |  |  |

### 应用组件安装

将部署阿凡搭平台所需的所有应用组件、数据库和文件存储服务安装包上传到服务器/home/software目录下。

#### 组件清单

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **资源类型** | **资源名称** | **版本** | **说明** | **资源包或下载地址** |
| 运行环境 | JDK | 1.8 | jdk-8u201-linux-x64.tar.gz | 链接：https://pan.baidu.com/s/1ckNXAjLU6L-meFHI7J6iDg?pwd=o28a  提取码：o28a |
| 中间件 | Tomcat | 8.5 | apache-tomcat-8.5.57.tar.gz |
| Web  服务器 | Nginx | 1.18 | nginx-1.18.0.tar.gz |
| 缓存存储数据库 | Redis | 6.26 | redis-6.2.6.tar.gz |
| 文件存储服务 | Minio | 6.0 | minio |
| 附件在线预览 | KkFileView | 4.0 | kkFileView-4.0.0.tar.gz |
| kkFileView-4.0.0.jar |
| LibreOffice\_7.1.4.2\_Linux\_x86-64\_rpm.zip |
| fonts.zip |
| font\_rpm.zip |
| 文件转换 | unoconv | 0.8.2 | unoconv-0.8.2.zip |
| 关系数据库 | MySQL | 8.0 | mysql-8.0.28-el7-x86\_64.tar.gz |

#### 安装JDK

1. 在software目录下解压JDK压缩包并重命名；

目录：cd /home/software

命令：tar -zxvf jdk-8u201-linux-x64.tar.gz

命令：mv jdk1.8.0\_201 jdk

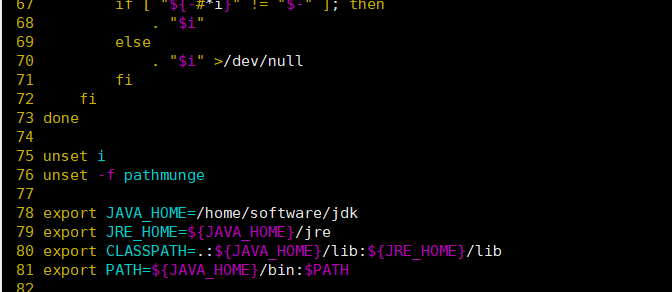
1. 配置JDK环境变量，在profile文件中文本最后添加以下信息:

export JAVA\_HOME=/home/software/jdk

export JRE\_HOME=${JAVA\_HOME}/jreexport CLASSPATH=.:${JAVA\_HOME}/lib:${JRE\_HOME}/lib

export PATH=${JAVA\_HOME}/bin:$PATH

命令：vim /etc/profile

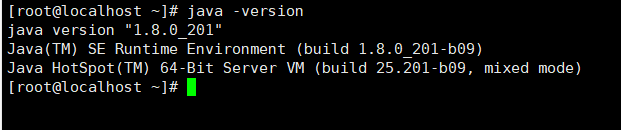


1. 保存后执行命令让profile文件立马生效

命令：source /etc/profile

1. 查看安装的版本

命令：java -version

显示版本号信息则表示安装成功。

#### 安装Nginx

1. 首先安装依赖包

命令：yum -y install gcc zlib zlib-devel pcre-devel openssl openssl-devel unzip

出现选择时，输入y，一直等待安装完成

1. 在software目录下解压nginx压缩包

目录：cd /home/software

命令：tar -zxvf nginx-1.18.0.tar.gz

1. 在nginx目录下创建日志目录

目录：cd /home/software/nginx-1.18.0

命令：mkdir logs

1. 进入nginx目录，执行安装

目录：cd /home/software/nginx-1.18.0

命令：./configure --prefix=/home/software/nginx-1.18.0 --with-http\_stub\_status\_module --with-http\_ssl\_module

1. 在nginx目录执行编译命令

目录：cd /home/software/nginx-1.18.0

命令：make

1. 在nginx目录执行安装命令

目录：cd /home/software/nginx-1.18.0

命令：make install

1. 在nginx目录sbin目录下执行启动命令

目录：cd /home/software/nginx-1.18.0/sbin

命令：./nginx -c /home/software/nginx-1.18.0/conf/nginx.conf

1. 浏览器访问IP:80，验证nginx安装和启动状态

出现 Welcome to nginx! 表示启动成功

修改nginx配置文件参考如下：

查看/修改配置命令：vim /home/software/nginx-1.18.0/conf/nginx.conf



重启nginx命令：/home/software/nginx-1.18.0/sbin/nginx -s reload -c /home/software/nginx-1.18.0/conf/nginx.conf

#### 安装Redis

1. 在softwware目录下解压Redis压缩包

目录：cd /home/software

命令：tar -zxvf redis-6.2.6.tar.gz

1. 修改redis相关配置，

目录：cd /home/software/redis-6.2.6

命令：vim redis.conf

1. 将redis.conf文件中daemonize的值修改为yes;
2. 将redis.conf文件中requirepass前面的注释去掉，并将redis密码值改为：123456

/+关键字 ，回车即可。

此为从文档当前位置向下查找关键字，按n键查找关键字下一个位置；

?+关键字，回车即可。

此为从文档当前位置向上查找关键字，按n键向上查找关键字；

1. 在redis目录下执行编译

目录：cd /home/software/redis-6.2.6

命令：make

1. 在redis目录下执行启动redis命令

目录：cd /home/software/redis-6.2.6

命令：. /src/redis-server /home/software/redis-6.2.6/redis.conf

默认情况下，redis只允许本机访问。

如果要设置redis允许远程连接，需要修改配置文件。

修改配置文件redis.conf：

修改第70行，将bind 127.0.0.1 注释

修改第90行，将protected-mode修改为no

#### 安装Minio

1. 创建minio文件夹，并在minio文件夹下上传minio文件

目录：cd /home/software

命令：mkdir minio

1. 创建文件存放目录和日志存放目录

目录：cd /home/software/minio

命令：mkdir minio\_data

命令：mkdir minio\_log

1. 创建minio启动脚本

目录：cd /home/software/minio

命令：vim minio\_start.sh

复制以下脚本代码作为minio\_start.sh的内容：

#!/bin/sh

echo "minio开始启动.........................................................................."

export MINIO\_ROOT\_USER=minioadmin

export MINIO\_ROOT\_PASSWORD=minioadmin

./minio server --console-address 0.0.0.0:9001 /home/software/minio/minio\_data > /home/software/minio/minio\_log/minio.log 2>&1 &

echo "minio启动成功.........................................................................."

1. 授权minio及minio\_start启动脚本

目录：cd /home/software/minio

命令：chmod +x minio

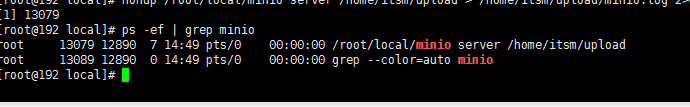
命令：chmod +x minio\_start.sh

1. 启动minio

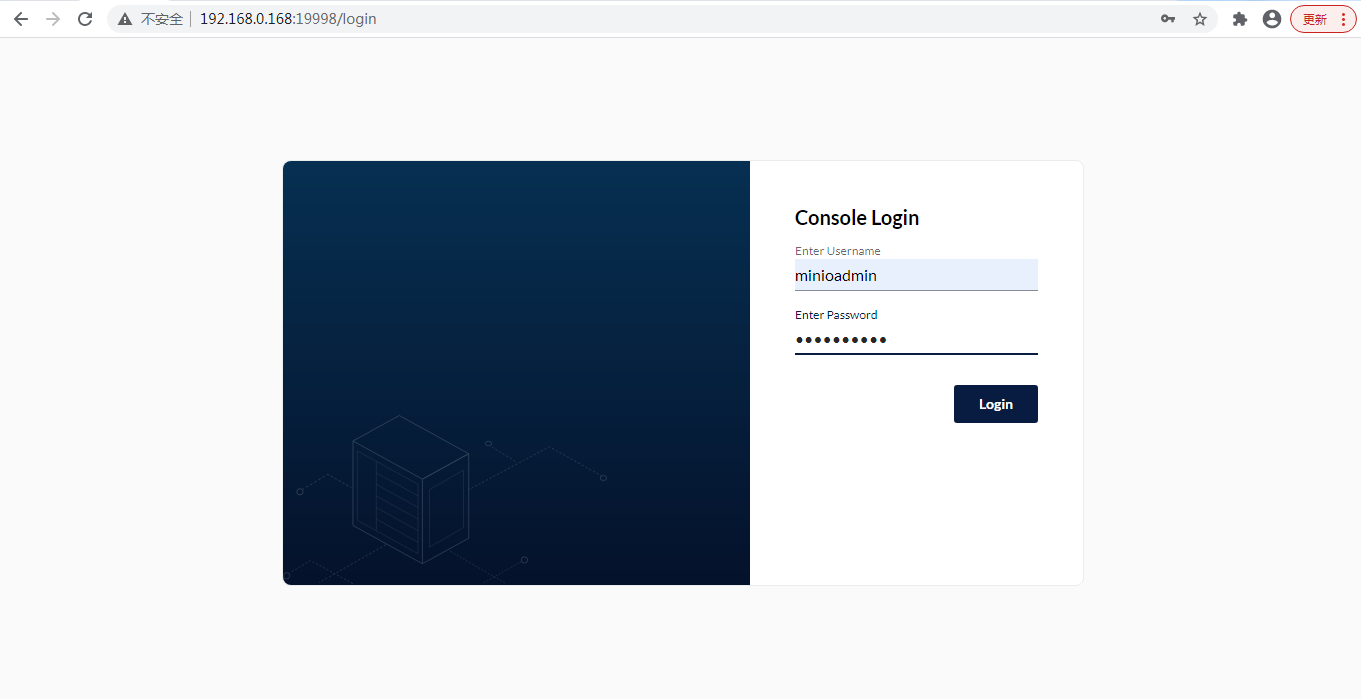
目录：cd /home/software/minio

命令：./minio\_start.sh

查看minio进程命令：ps -ef | grep minio



1. 浏览器访问IP:9000，验证minio安装和启动状态



如上图所示：minio初始登录用户和密码都为minioadmin

#### 安装KkFileView

1. 在software目录下解压KkFileView压缩包。

目录：cd /home/software

命令：tar -zxvf kkFileView-4.0.0.tar.gz

1. 在KkFileView解压包bin目录下执行安装命令。

目录：cd /home/software/kkFileView-4.0.0/bin

命令：./install.sh

1. 进入LibreOffice目录，执行安装命令。

命令：cd /tmp/LibreOffice\_7.1.4.2\_Linux\_x86-64\_rpm/RPMS

命令：yum localinstall \*.rpm

如果离线状态下，可以不用执行./install.sh命令，直接将LibreOffice\_7.1.4.2\_Linux\_x86-64\_rpm.zip包解压至tmp目录，然后按上述步骤直接执行yum localinstall \*.rpm进行安装

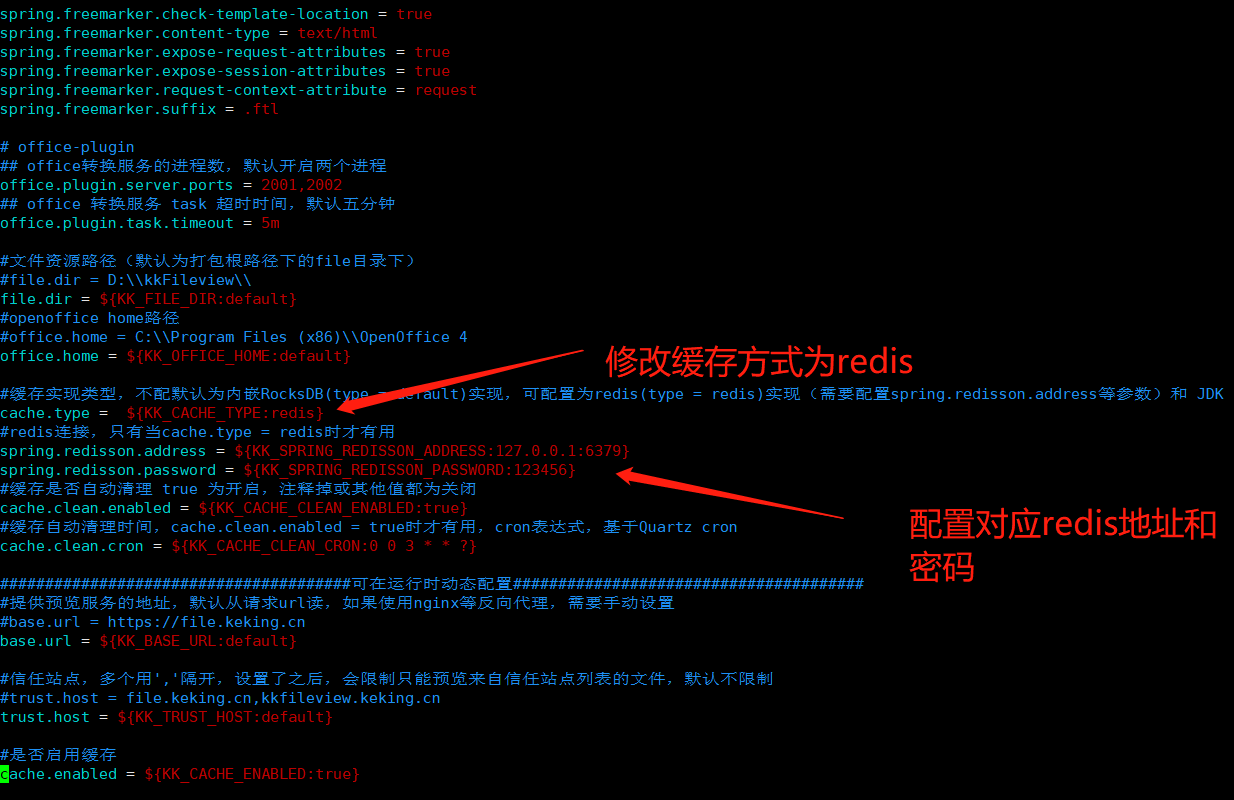
1. 配置kkFileView。

命令：cd /home/software/kkFileView-4.0.0/config

命令：vim application.properties

将cache.type的值改为redis；

将redis连接密码改为123456；



1. 替换jar包

目录：cd /home/software/kkFileView-4.0.0/bin

替换：将kkFileView-4.0.0.jar包上传至bin目录进行替换

1. 启动kkFileView。

目录：cd /home/software/kkFileView-4.0.0/bin

命令：./startup.sh

1. 验证kkFileView启动状态。

浏览器访问IP:8012



如果在线预览word文件出现乱码

1. 上传fonts.zip至 /usr/share/fonts 目录下并解压
2. 依次执行 mkfontscale 、mkfontdir 、fc-cache 使字体生效
3. 如果执行mkfontscale失败，使用 yum install -y mkfontscale 安装

如果执行mkfontscale失败，可以使用 yum install -y mkfontscale 安装（离线环境时，可上传依赖包font\_rpm.zip至/usr/share/fonts目录并解压，然后进入解压后目录，执行安装命令rpm -ivh ./\*.rpm --nodeps --force）

1. 授权字体访问权限，进入解压后的目录，执行命令：

chmod -R 755 \*.TTF

chmod -R 755 \*.ttf

chmod -R 755 \*.ttc

1. 刷新配置source /etc/profile
2. 重启kkFileView
3. 清空kkFileView缓存

连接redis，找到对应的kkFileView存储的key值进行删除

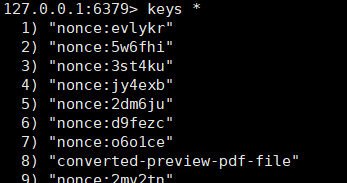
目录：cd /home/software/redis-6.2.6/src

命令：./redis-cli

命令：auth 填写redis密码进行登录



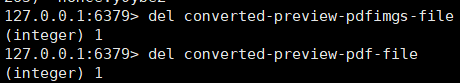
命令：keys \*



删除对应key

命令：del converted-preview-pdfimgs-file

命令：del converted-preview-pdf-file



#### 安装Unoconv

基于已经安装Libreoffice且部署好中文字体包的前提下，安装UNOVOCN，具体office安装可以参考kkFileView里office安装步骤。

1. 在software目录下解压unoconv压缩包

目录：cd /home/software

命令：unzip unoconv-0.8.2.zip

1. 进入unoconv目录，执行安装

目录：cd /home/software/unoconv-0.8.2

命令：make install

1. 验证，将md文件转换为pdf

命令：unoconv -f pdf CHANGELOG.md

#### 安装Mysql

1. 将software目录下mysql压缩包解压并重命名。

目录：cd /home/software

命令：tar -zxvf mysql-8.0.28-el7-x86\_64.tar.gz

命令：mv mysql-8.0.28-el7-x86\_64 mysql

1. 进入mysql目录下创建data目录。

目录：cd /home/software/mysql

命令：mkdir data

1. 创建mysql用户组、用户密码并授权

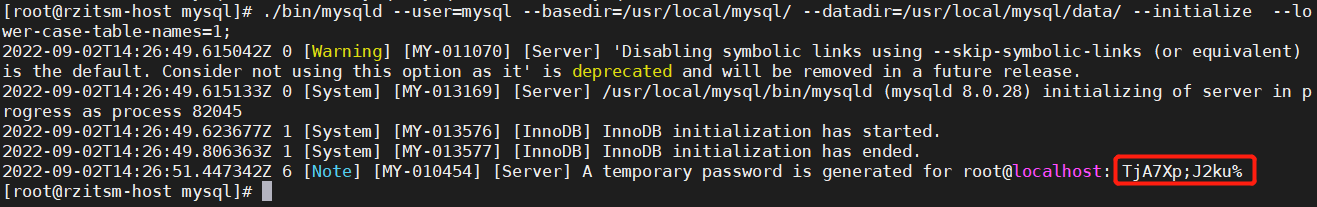
命令：groupadd mysql

命令：useradd -g mysql mysql

命令：chown -R mysql.mysql /home/software/mysql

1. 初始化数据库

命令：./bin/mysqld --user=mysql --basedir=/home/software/mysql/ --datadir=/home/software/mysql/data/ --initialize --lower-case-table-names=1



TjA7Xp;J2ku%为初始化密码。

1. 修改my.cnf文件，配置mysql连接相关值

命令：vim /etc/my.cnf

增加或修改配置项：port、basedir、datadir、socket、lower\_case\_table\_names、skip-grant-tables

[mysqld]

port=3306

basedir=/home/software/mysql

datadir=/home/software/mysql/data

socket=/tmp/mysql.sock

lower\_case\_table\_names=1

character-set-server=UTF8MB4

default-storage-engine=INNODB

default\_authentication\_plugin=mysql\_native\_password

group\_concat\_max\_len=4294967295

secure-file-priv=''

skip-grant-tables

#symbolic-links=0

# Settings user and group are ignored when systemd is used.

# If you need to run mysqld under a different user or group,

# customize your systemd unit file for mariadb according to the

# instructions in http://fedoraproject.org/wiki/Systemd

# 开启慢查询日志

slow\_query\_log = 1

# 指定慢查询日志文件路径

slow\_query\_log\_file = /home/software/mysql/slow-query.log

# 设置慢查询的阈值，单位为秒

long\_query\_time = 2

binlog\_format=row

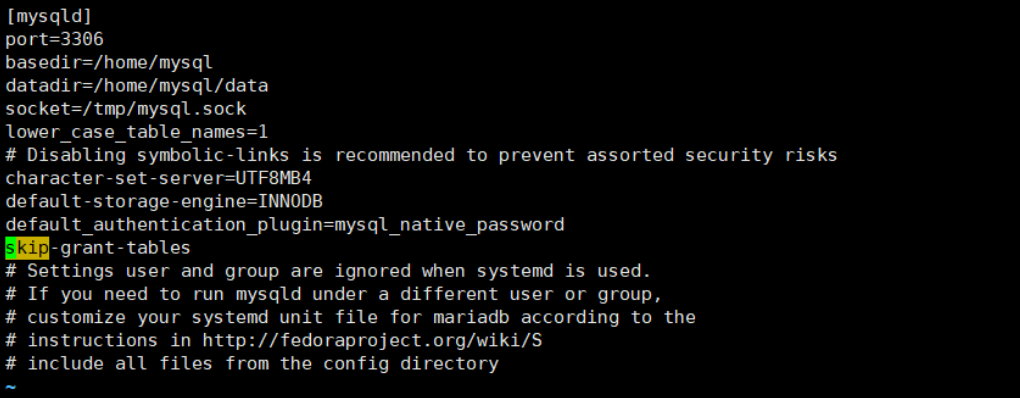
server\_id=1

log-bin=master

user=mysql

注意：skip-grant-tables这一行配置，需要先开放，修改完密码后再回到这个文件注释这一行配置。

如下图：



1. 添加mysqld服务到系统并授权

目录：cd /home/software/mysql

指定mysql的基本目录

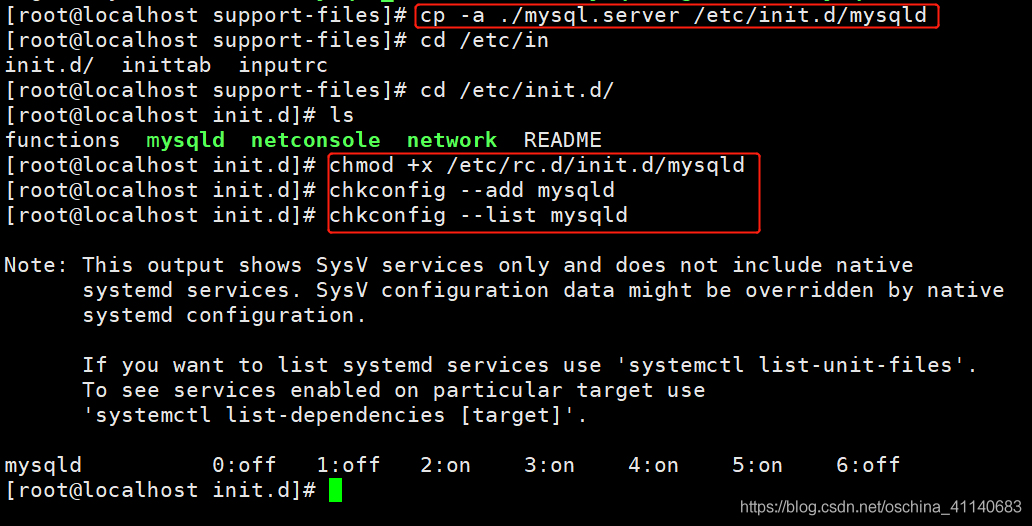
vim ./support-files/mysql.server

basedir=/home/software/mysql

datadir=/home/software/mysql/data

命令：cp -a ./support-files/mysql.server /etc/init.d/mysqld

命令：chmod +x /etc/init.d/mysqld  
 命令：chkconfig --add mysqld  
 检查服务是否生效  
 命令：chkconfig --list mysqld



1. 将mysql命令添加到服务

命令：ln -s /home/software/mysql/bin/mysql /usr/bin

如果存在则使用下面命令覆盖

命令：ln -sf /home/software/mysql/bin/mysql /usr/bin

1. 启动MySQL服务

命令：service mysql start

查看启动状态

命令：service mysql status

1. 登录mysql

命令：mysql -uroot -p

提示输入密码，由于当前/etc/my.cnf里配置了免密登录skip-grant-tables

则直接回车就可以进入mysql

进入mysql数据库，刷新权限

命令：use mysql;

命令：flush privileges;

1. 清空root密码

命令：update user set authentication\_string='' where user='root';

命令：flush privileges;

1. 修改root密码及密码验证方式

命令：alter user 'root'@'localhost' IDENTIFIED WITH mysql\_native\_password BY '123456';

命令：flush privileges;

执行命令后，123456就是root账号设置的新密码

1. 修改root账号允许远程访问（如果需要远程访问可以设置）

命令：update user set host='%' where user='root';

命令：flush privileges;

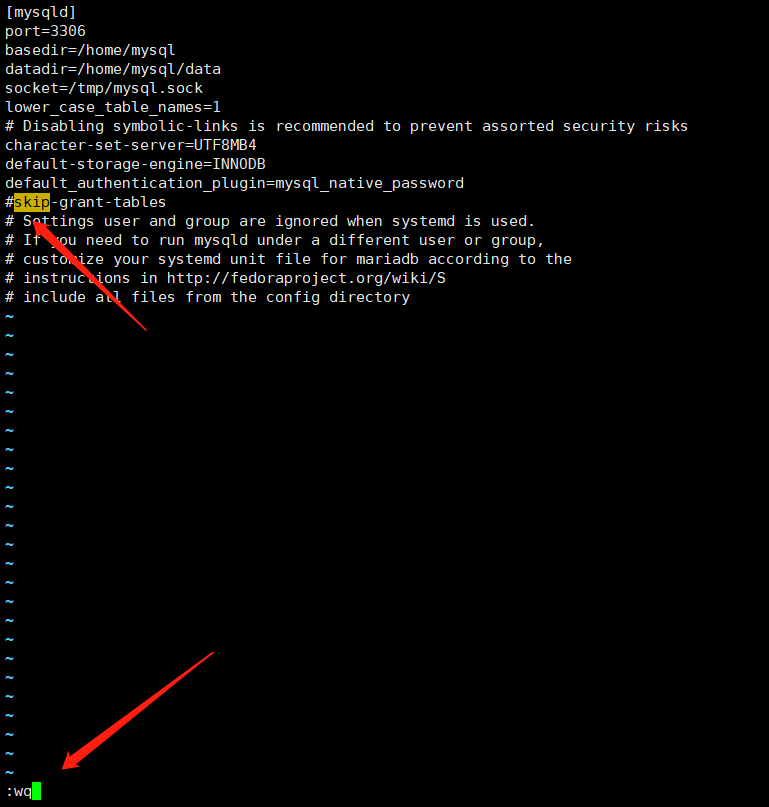
1. 退出mysql

命令：quit;

1. 取消免密登录

编辑my.cnf文件，将skip-grant-tables 注释然后保存文件退出

命令：vim /etc/my.cnf



1. 重启mysql服务

命令：service mysql restart

#### 安装Tomcat

将tomcat压缩包分别上传到engine和back目录下解压并重命名

目录：/home/ffapaas/engine

目录：/home/ffapaas/back

命令：tar -zxvf apache-tomcat-8.5.57.tar.gz

命令：mv apache-tomcat-8.5.57 tomcat

命令：rm -f apache-tomcat-8.5.57.tar.gz

### 平台部署

#### 部署文件清单

|  |  |  |
| --- | --- | --- |
| **类型** | **文件** | **说明** |
| 引擎部署文件 | config文件夹 | 配置文件夹，包含两个配置文件 |
| sjt\_agent.jar |  |
| setenv.sh |  |
| ff-apaas-engine.war | 引擎war包 |
| 后端部署文件 | config文件夹 | 配置文件夹，包含两个配置文件 |
| ffapaas.war | 后端war包 |
| 前端部署文件 | web.tar.gz | 前端部署压缩包 |
| 移动端部署文件 | mobileweb.tar.gz | 移动端部署压缩包 |
| 数据库文件 | ffapaas.sql | 初始化数据库备份文件 |
| 授权文件及配置参数文件 | XXXX | 通过访问左侧URL链接填写相关数据获取授权文件及配置参数文件 |

#### 初始化数据库

1. 将数据库备份文件ffapaas.sql文件上传到目录home/ffapaas下
2. 登录mysql创建ffapaas数据库并设置字符集；

命令：mysql -uroot -p

命令：create database ffapaas;

命令：show databases;

命令：use ffapaas;

命令：set names utf8mb4;

命令：set foreign\_key\_checks=0;

1. 导入数据库；

命令：source /home/ffapaas/ffapaas.sql;

命令：set foreign\_key\_checks=1;

#### 系统引擎部署

1. 将Tomcat压缩包上传到engine目录下解压并重命名（如已完成则忽略此步骤）

进入目录：cd /home/ffapaas/engine

解压：tar -zxvf apache-tomcat-8.5.57.tar.gz

重命名：mv apache-tomcat-8.5.57 tomcat

删除压缩包：rm -f apache-tomcat-8.5.57.tar.gz

1. Tomcat监听端口配置

进入目录：cd /home/ffapaas/engine/tomcat/conf/

编辑文件：vim server.xml

将 <Server port="8005" shutdown="SHUTDOWN"> 中的8005改为8006；

将<Connector port="8080" protocol="HTTP/1.1" connectionTimeout="20000" redirectPort="8443" />中的8080改为8181、8443改成8446

1. Tomcat配置指定JDK

修改bin目录下setclasspath.sh文件内容 ，指定JDK目录；

JDK安装目录/home/software/jdk

进入目录：cd /home/ffapaas/engine/tomcat/bin/

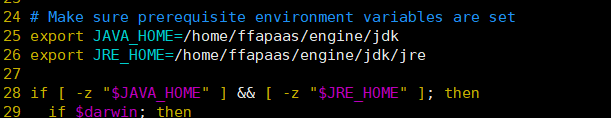
拷贝文件夹：cp /home/software/jdk/ -a /home/ffapaas/engine

编辑文件：vim setclasspath.sh

在文件中顶部添加如下内容：

export JAVA\_HOME=/home/ffapaas/engine/jdk

export JRE\_HOME=/home/ffapaas/engine/jdk/jre



1. 上传相关文件及参数配置
2. 将授权文件和sjt\_agent.jar文件上传到/home/ffapaas/engine/目录下；
3. 将配置文件夹config上传至tomcat目录conf文件夹下，config中包含文件：application.yml和application-prod.yml文件；

修改application.yml文件内容：

* minioEndpoint：minio服务地址及端口；
* accessKey：启动minio服务时的MINIO\_ACCESS\_KEY的值，默认minioadmin；
* secretKey：启动minio服务时设置的MINIO\_SECRET\_KEY值，默认minioadmin；
* fileFullPath：授权文件完整路径加文件名及文件后缀；
* platformPrivateKey：授权文件私钥；
* platformUrl：平台后台接口地址；
* dbType: 数据库类型；

修改application-prod.yml文件内容：

* driverClassName：数据库驱动
* master下url：数据库配置连接地址及相关参数；
* master下username：数据库登录账号；
* master下password：数据库登录密码（加密后的字符串）；
* privateKey：密码解密私钥
* redis下host：redis地址；
* redis下port：redis地址端口；
* redis下password：redis登录密码；

1. 将setenv.sh文件上传到/home/ffapaas/engine/tomcat/bin/目录下；

setenv.sh文件中内容设置环境变量如：

export CATALINA\_OPTS="$CATALINA\_OPTS -javaagent:/home/ffapaas/engine/sjt\_agent.jar"；

1. 部署引擎包

将引擎包文件ff-apaas-engine.war上传/home/ffapaas/engine/tomcat/webapps/目录下；

1. 验证

启动tomcat

目录：/home/ffapaas/engine/tomcat/bin

命令：./startup.sh

浏览器访问http://IP:8181/ff-apaas-engine/doc.html，显示如下图表示部署成功！



#### 应用后端部署

1. 将Tomcat压缩包上传到back目录下解压并重命名（如已完成则忽略此步骤）

进入目录：cd /home/ffapaas/back

解压：tar -zxvf apache-tomcat-8.5.57.tar.gz

重命名：mv apache-tomcat-8.5.57 tomcat

删除压缩包：rm -f apache-tomcat-8.5.57.tar.gz

1. Tomcat配置指定JDK

修改bin目录下setclasspath.sh文件内容 ，指定JDK目录；

JDK安装目录/home/software/jdk

进入目录：cd /home/ffapaas/back/tomcat/bin/

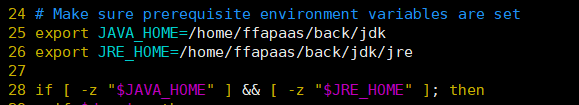
拷贝文件夹：cp /home/software/jdk/ -a /home/ffapaas/back

编辑文件：vim setclasspath.sh

在文件中顶部添加如下内容：

export JAVA\_HOME=/home/ffapaas/back/jdk

export JRE\_HOME=/home/ffapaas/back/jdk/jre



1. 上传相关文件及配置相关参数

将配置文件夹config上传至tomcat目录conf文件夹下，config中包含文件：application.yml和application-prod.yml文件；

修改application.yml文件内容：

* profile: 上传文件临时路径 默认/home/ffapaas/upload
* minioEndpoint：minio服务地址及端口；
* accessKey：启动minio服务时的MINIO\_ACCESS\_KEY的值，默认minioadmin；
* secretKey：启动minio服务时设置的MINIO\_SECRET\_KEY值，默认minioadmin；
* engineUrl：引擎后台接口地址；
* mobileBackEnd：移动端后台地址(暂时不变，后续配置完移动端在调整)
* dbType：数据库类型；
* appKey：签名验证编号；
* appSecret: 签名验证秘钥；
* signSource: 签名来源；

修改application-prod.yml文件内容：

* driverClassName：数据库驱动
* master下url：数据库配置连接地址及相关参数；
* master下username：数据库登录账号；
* master下password：数据库登录密码（加密后的字符串）；
* privateKey：密码解密私钥
* redis下host：redis地址；
* redis下port：redis地址端口；
* redis下password：redis登录密码；

1. 部署后台包

将后台包文件ffapaas.war上传到/home/ffapaas/back/tomcat/webapps/目录下；

1. 验证

启动tomcat

目录：/home/ffapaas/back/tomcat/bin

命令：./startup.sh

浏览器访问http://IP:8080/ffapaas/doc.html，显示如下图表示部署成功！



#### 应用前端部署

1. 上传web部署压缩包到ffapaas目录下解压；

进入目录：cd /home/ffapaas

解压：tar -zxvf web.tar.gz

如果压缩包是web.zip，则执行unzip web.zip进行解压。

1. 修改前端配置参数

进入目录：cd /home/ffapaas/web

编辑文件：vim config.js

* window.BASEURL：后台API接口地址；
* window.ENGINEBASEURL：流程引擎接口地址；
* window.PLATFORMURL：管理后台地址；
* window.FILEVIEW：在线预览地址；
* window.APPKEY：签名验证编号；
* window.APPSECRET: 签名验证秘钥；
* window.SIGNSOURCE: 签名来源；

1. 重新启动nginx

进入目录：cd /home/software/nginx-1.18.0/sbin/

重启nginx：./nginx -s reload -c /home/software/nginx-1.18.0/conf/nginx.conf

#### 移动端应用部署（微信公众号）

##### 移动后端部署

1. 登录PC端网页进入微信公众平台管理中心->服务连接->公众号填写相关公众号配置值，如下图：



1. 在ffapaas目录下创建mobile目录，在mobile目录下创建mobile\_back目录
2. 将Tomcat压缩包上传到mobile\_back目录下解压并重命名

进入目录：cd /home/ffapaas/mobile/mobile\_back

解压：tar -zxvf apache-tomcat-8.5.57.tar.gz

重命名：mv apache-tomcat-8.5.57 tomcat

删除压缩包：rm -f apache-tomcat-8.5.57.tar.gz

1. Tomcat监听端口配置

进入目录：cd /home/ffapaas/mobile/mobile\_back/tomcat/conf/

编辑文件：vim server.xml

将<Server port="8005" shutdown="SHUTDOWN">中的8005改为6005；

将<Connector port="8080" protocol="HTTP/1.1" connectionTimeout="20000" redirectPort="8443" />中的8080改为6161、8443改成6443

1. Tomcat配置指定JDK

修改bin目录下setclasspath.sh文件内容 ，指定JDK目录；

JDK安装目录/home/software/jdk

进入目录：cd /home/ffapaas/mobile/mobile\_back/tomcat/bin/

拷贝文件夹：cp /home/software/jdk/ -a /home/ffapaas/mobile/mobile\_back

编辑文件：vim setclasspath.sh

在文件中顶部添加如下内容：

export JAVA\_HOME=/home/ffapaas/mobile/mobile\_back/jdk

export JRE\_HOME=/home/ffapaas/mobile/mobile\_back/jdk /jre

1. 上传相关文件

将配置文件夹config\_mobile上传至tomcat目录conf文件夹下，config中包含文件：application.yml和application-prod.yml文件；

修改application.yml文件内容：

* sourceArray: 支持的移动端部署类型
* backUrl：后台接口地址

修改application-prod.yml文件内容：

* redis下host：redis地址；
* redis下port：redis地址端口；
* redis下password：redis登录密码；

1. 部署后台包

将移动后台包文件ff-apaas-mobile.war上传到

/home/ffapaas/mobile/mobile\_back/tomcat/webapps/目录下

1. 验证

启动tomcat

目录：/home/ffapaas/mobile/mobile\_back /tomcat/bin

命令：./startup.sh

浏览器访问<http://IP:6161/ff-apaas-mobile/doc.html>

1. 修改ffapaas后台

进入目录：cd /home/ffapaas/back/tomcat/conf/config

编辑文件：vim application.yml

修改application.yml文件内容：

* mobileBackEnd：移动端后台地址

进入目录：cd /home/ffapaas/back/tomcat/bin

关闭tomcat: ./shutdown.sh

启动tomcat: ./startup.sh

##### 移动前端部署

1. 上传web部署压缩包到mobile目录下解压；

进入目录：cd /home/ffapaas/mobile

解压：tar -zxvf mobileweb.tar.gz

如果压缩包是mobileweb.zip，则执行unzip mobileweb.zip进行解压。

1. 修改前端配置

进入目录：cd /home/ffapaas/mobile/mobile\_web/static/config

编辑文件：vim url.config.js

* BASEURL：后台API接口地址；
* ENGINEBASEURL：流程引擎接口地址；
* FILEVIEW：在线预览地址；
* PLATFORM：移动端类型,默认值 MP-WEIXIN；
* BASEURLMOBILE: 移动后台接口地址；
* APPKEY：签名验证编号；
* APPSECRET: 签名验证秘钥；
* SIGNSOURCE: 签名来源；
* Invitetoregister：邀请注册地址

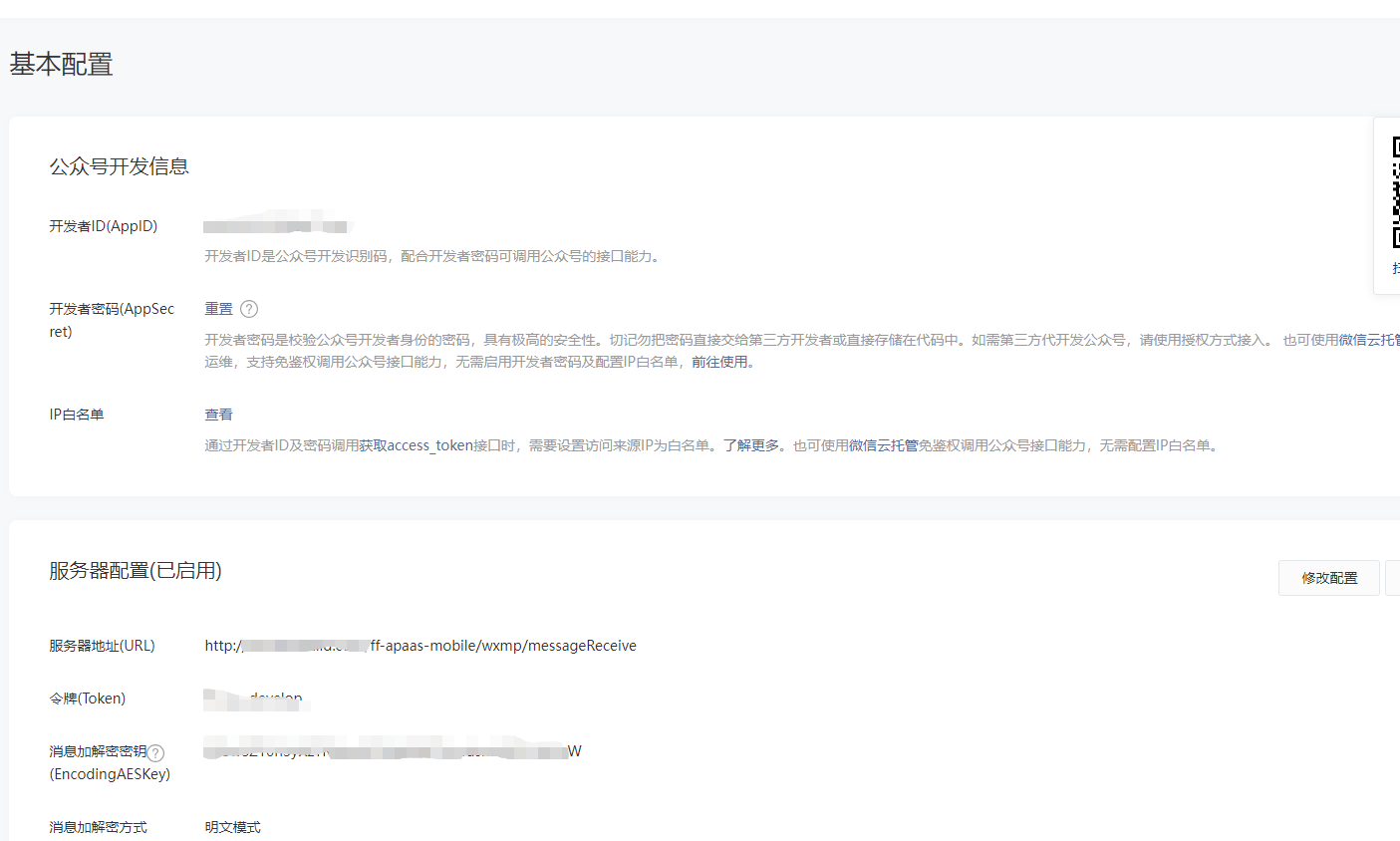
1. 配置并重新启动nginx

进入目录：cd /home/software/nginx-1.18.0/conf 修改nginx.conf增加移动端代理

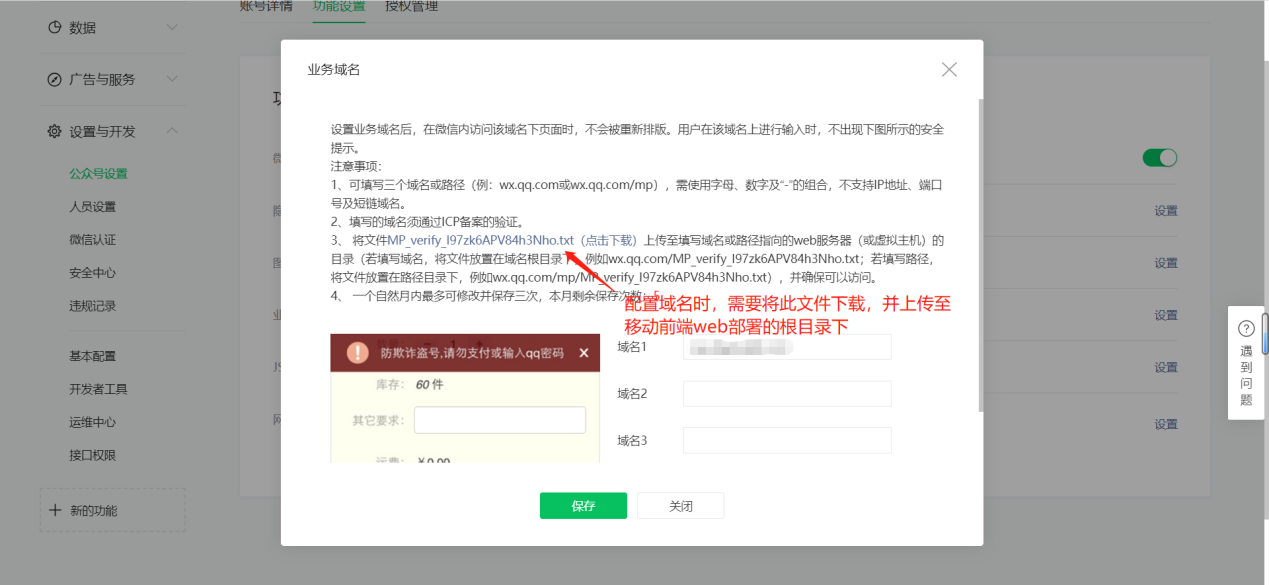
进入目录：cd /home/software/nginx-1.18.0/sbin/

重启nginx：./nginx -s reload -c /home/software/nginx-1.18.0/conf/nginx.conf

1. 进入微信公众平台修改相关配置，及配置访问白名单









## 私有化容器部署

### 在线部署

#### 环境资源清单

|  |  |  |  |
| --- | --- | --- | --- |
| **资源类型** | **资源名称** | **镜像/版本** | **说明** |
| 运行环境 | JDK | 1.8 |  |
| 容器引擎 | Docker | 24.0.2（docker-ce-24.0.2-1.el7.x86\_64.tar.gz） |  |
| 容器编排服务 | Docker Compose | 2.21.0 |  |
| 中间件 | Tomcat | tomcat-8.5.47.tar、ff-tomcat-fonts:8.5.47.tar |  |
| Web服务器 | Nginx | nginx:1.20.2.tar |  |
| 缓存存储数据库 | Redis | redis:6.2.6.tar |  |
| 文件存储服务 | Minio | minio-RELEASE.2022-03-03T21-21-16Z.tar |  |
| 附件在线预览 | KkFileView | kkfileview-v4.0.0.tar |  |
| 关系数据库 | MySQL | mysql:8.0.28.tar |  |

#### Docker安装

1. 安装yum-utils包；

命令：yum install -y yum-utils

1. 设置存储库

命令：yum-config-manager --add-repo https://mirrors.aliyun.com/docker-ce/linux/centos/docker-ce.repo

1. 安装Docker引擎

命令：yum install docker-ce-24.0.2 docker-ce-cli-24.0.2 containerd.io -y

1. 启动docker服务

命令：systemctl start docker

1. 复制docker-compose

将docker-compose复制到/usr/local/bin目录下

命令：cp /usr/libexec/docker/cli-plugins/docker-compose /usr/local/bin/

#### 平台部署

##### 文件清单

|  |  |  |  |
| --- | --- | --- | --- |
| **部署类型** | **服务器** | **文件** | **文件说明** |
| 单服务器 | 部署服务器 | ffapaas.zip | 前后端及引擎部署包 |
| install.sh | 启动文件 |
| 多服务器 | 应用服务器（支持多台） | ffapaas-webapp.zip | 前后端及引擎部署包 |
| install-webapp.sh | 应用启动文件 |
| 数据库服务器 | ffapaas-mysql.zip | 数据库文件包 |
| install-mysql.sh | 数据库启动文件 |

##### 部署步骤

1. 上传部署包和启动文件

将ffapaas.zip/ffapaas-webapp.zip/ffapaas-mysql.zip和 install.sh/install-webapp.sh/install-mysql.sh启动文件上传到/opt目录下;

1. 授权并启动

目录：/opt

命令：chmod +x install.sh

命令：./install.sh

是否手动输入当前服务器IP地址 ?(y/n):

选择n；

如果install.sh文件有在windows系统编辑过，需执行格式转换，然后再执行启动

命令：yum install -y dos2unix

命令：dos2unix install.sh

命令：./install.sh

启动文件执行过程中，会自动下载镜像，需等待镜像全部下载完成。

1. 查看所有镜像

命令：docker images

##### 验证

浏览器直接访问web应用部署服务器IP地址；

### 离线部署

#### 环境资源清单

|  |  |  |  |
| --- | --- | --- | --- |
| **资源类型** | **资源名称** | **镜像/版本** | **说明** |
| 运行环境 | JDK | 1.8 |  |
| 容器引擎 | Docker | 24.0.2（docker-ce-24.0.2-1.el7.x86\_64.tar.gz） |  |
| 容器编排服务 | Docker Compose | 2.21.0 |  |
| 中间件 | Tomcat | tomcat-8.5.47.tar、ff-tomcat-fonts:8.5.47.tar |  |
| Web服务器 | Nginx | nginx:1.20.2.tar |  |
| 缓存存储数据库 | Redis | redis:6.2.6.tar |  |
| 文件存储服务 | Minio | minio-RELEASE.2022-03-03T21-21-16Z.tar |  |
| 附件在线预览 | KkFileView | kkfileview-v4.0.0.tar |  |
| 关系数据库 | MySQL | mysql:8.0.28.tar |  |

#### Docker安装

1. 安装docker及相关依赖；

将安装文件docker-ce-24.0.2-1.el7.x86\_64.tar.gz上传到服务器Home目录下；

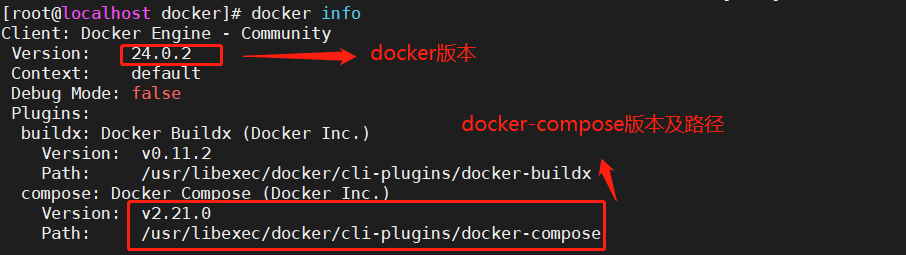
执行解压命令：tar -zxvf docker-ce-24.0.2-1.el7.x86\_64.tar.gz

成功解压后，进入解压目录：cd /home/docker

执行安装命令：rpm -ivh \*.rpm --force --nodeps

1. 启动docker

查看安装的docker信息，命令：docker info



将docker-compose复制到/usr/local/bin目录下

命令：cp /usr/libexec/docker/cli-plugins/docker-compose /usr/local/bin/

启动docker，命令：systemctl start docker

#### 镜像加载

1. 将镜像文件上传到服务器home/docker-images目录下；
2. 在home/docker-images目录下执行加载镜像命令；

进入目录命令：cd /home/docker-images

加载redis镜像命令：docker load -i redis:6.2.6.tar

加载minio镜像命令：docker load -i minio-RELEASE.2022-03-03T21-21-16Z.tar

加载kkfileview镜像命令：docker load -i kkfileview-v4.0.0.tar

加载tomcat镜像命令：

docker load -i tomcat-8.5.47.tar

docker load -i ff-tomcat-fonts:8.5.47.tar

加载nginx镜像命令：docker load -i nginx:1.20.2.tar

加载mysql镜像命令：docker load -i mysql:8.0.28.tar

#### 平台部署

##### 文件清单

|  |  |  |  |
| --- | --- | --- | --- |
| **部署类型** | **服务器** | **文件** | **文件说明** |
| 单服务器 | 部署服务器 | ffapaas.zip | 前后端及引擎部署包 |
| install.sh | 启动文件 |
| 多服务器 | 应用服务器（支持多台） | ffapaas-webapp.zip | 前后端及引擎部署包 |
| install-webapp.sh | 应用启动文件 |
| 数据库服务器 | ffapaas-mysql.zip | 数据库文件包 |
| install-mysql.sh | 数据库启动文件 |

##### 部署步骤

1. 上传部署包和启动文件

将ffapaas.zip/ffapaas-webapp.zip/ffapaas-mysql.zip和 install.sh/install-webapp.sh/install-mysql.sh启动文件上传到/opt目录下;

1. 授权并启动

目录：/opt

命令：chmod +x install.sh

命令：./install.sh

是否手动输入当前服务器IP地址 ?(y/n):

选择n；

如果install.sh文件有在windows系统编辑过，需执行格式转换，然后再执行启动

命令：yum install -y dos2unix

命令：dos2unix install.sh

命令：./install.sh

启动文件执行过程中，会自动下载镜像，需等待镜像全部下载完成。

1. 查看所有镜像

命令：docker images

##### 验证

浏览器直接访问web应用部署服务器IP地址；

# 系统初始化

系统部署完成后，建议按照以下几步对系统进行初始化工作。

## 设置系统超级管理员

系统部署完成后，通过URL地址访问系统登录界面，点击如下图所示“账号重置”按钮即可初始化超级管理员账号。（账号重置完成后，该账号重置按钮会自动消失）



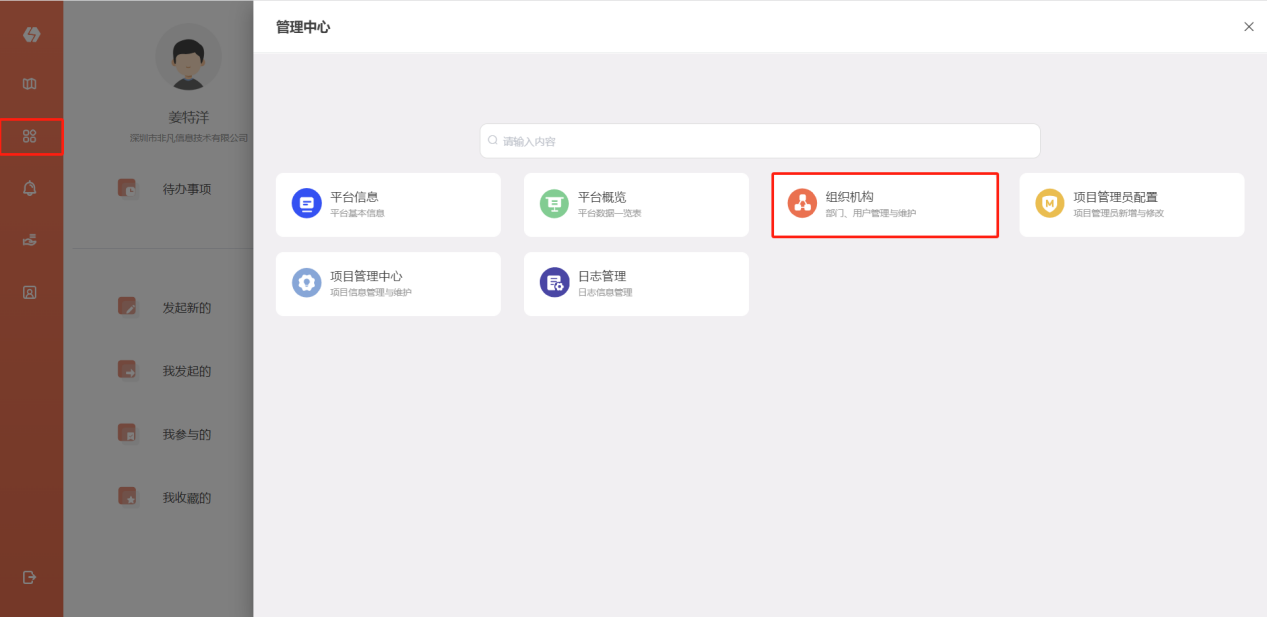
## 创建组织机构及用户

当我们在创建系统功能模块时，常常会关联到组织架构及用户的相关数据（例如权限分配、业务流程审批、表单数据填写），所以为了更贴切真实的业务场景，我们建议在创建系统功能模块之前，先完成组织机构及用户的创建。

阿凡搭平台支持三种方式（单个录入、excel批量导入、同步其他系统数据）来完成组织机构及用户的创建。

**第一种：部门及用户单个录入**

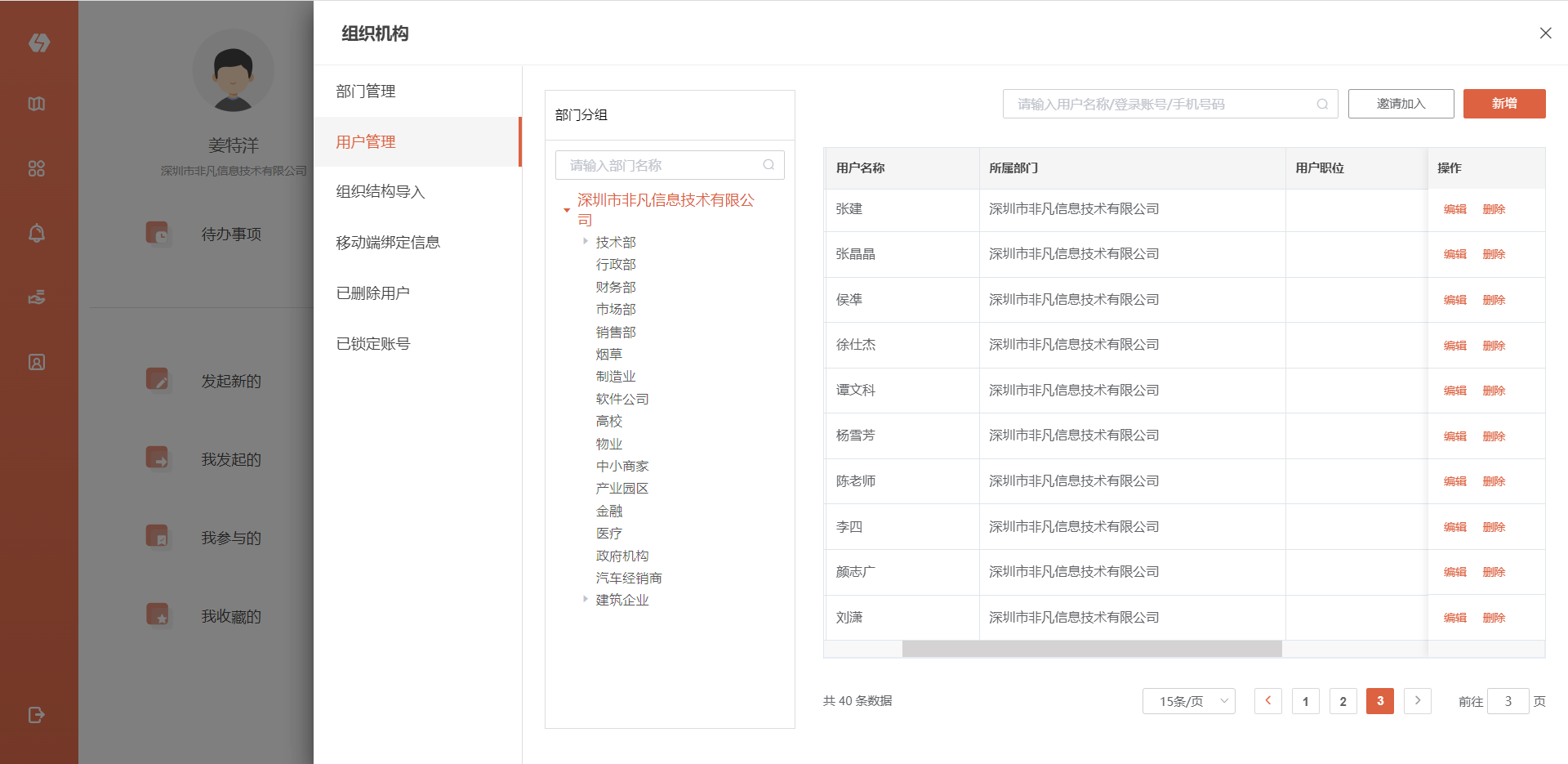
1. 单个录入：用超级管理员账号登录到系统，在最左侧点击“管理中心”菜单，然后点击管理中心面板中的”组织机构“。（操作如下图）



1. 在弹出的页面中，通过部门管理及用户管理对企业部门及企业用户进行创建与维护。（操作如下图）



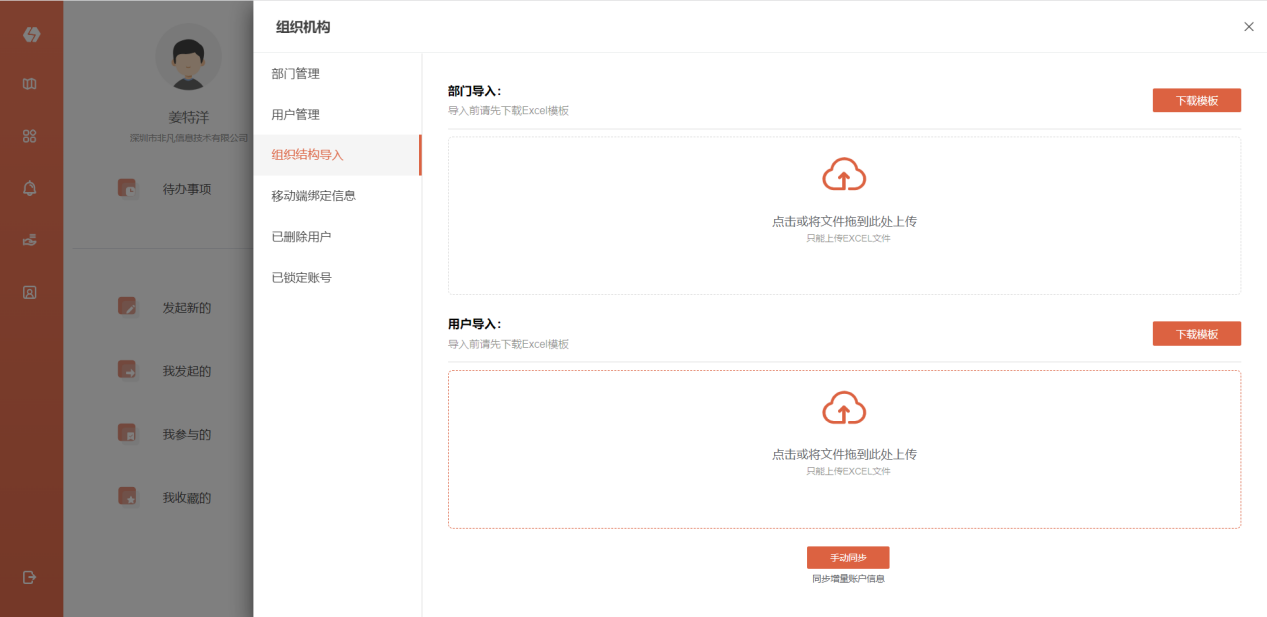
（部门维护）



（用户维护）

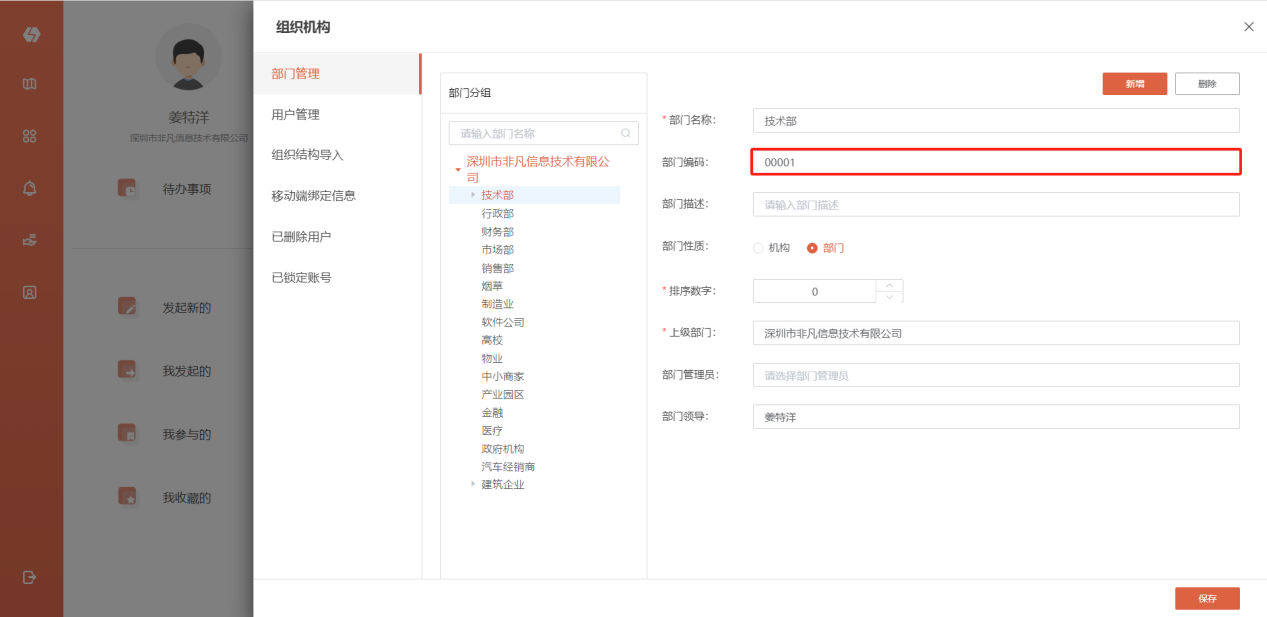
**第二种：excel批量导入组织结构**

前面步骤参考上述步骤，选择”组织机构导入“菜单，下载相关模板，在模板中填充相关数据，然后按照提示进行文件上传，点击导入，组织结构则导入成功。（操作如下图所示）



（组织机构导入）

**备注：用户导入时需填写部门编码，查看部门编码可通过点击部门管理，然后在部门树形结构中点击具体的部门名称，在右边属性栏中可看到对应部门的部门编码。若无部门编码可自定义一个（需确保部门编码的唯一性）。**



（查看部门编码）

**第三种：同步其他系统数据**

同步其他系统来完成组织机构及用户的创建一般需要根据每个企业的实际环境做少量二次开发才能完成，因为不同的企业主数据来源（OA 、HR等）存在差异。

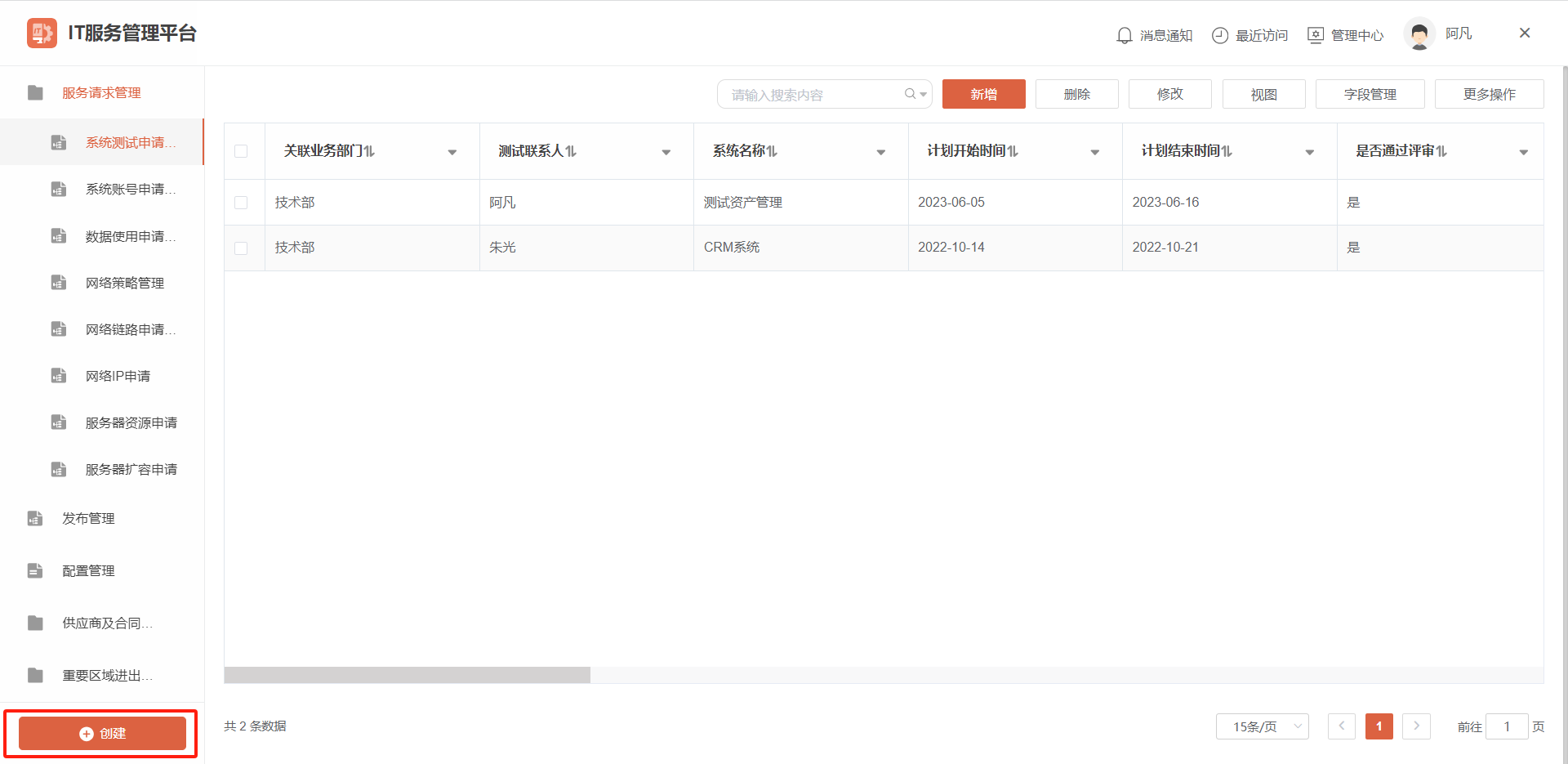
## 创建系统功能模块

在阿凡搭平台中，通过”创建应用“的方式来完成一个新功能模块的搭建，例如：客户信息管理、合同管理、费用报销管理、请假管理等等。但每一个功能模块必须归属于某个项目，不能是独立的个体。所以我们在**创建新功能模块之前必须先创建一个项目**。

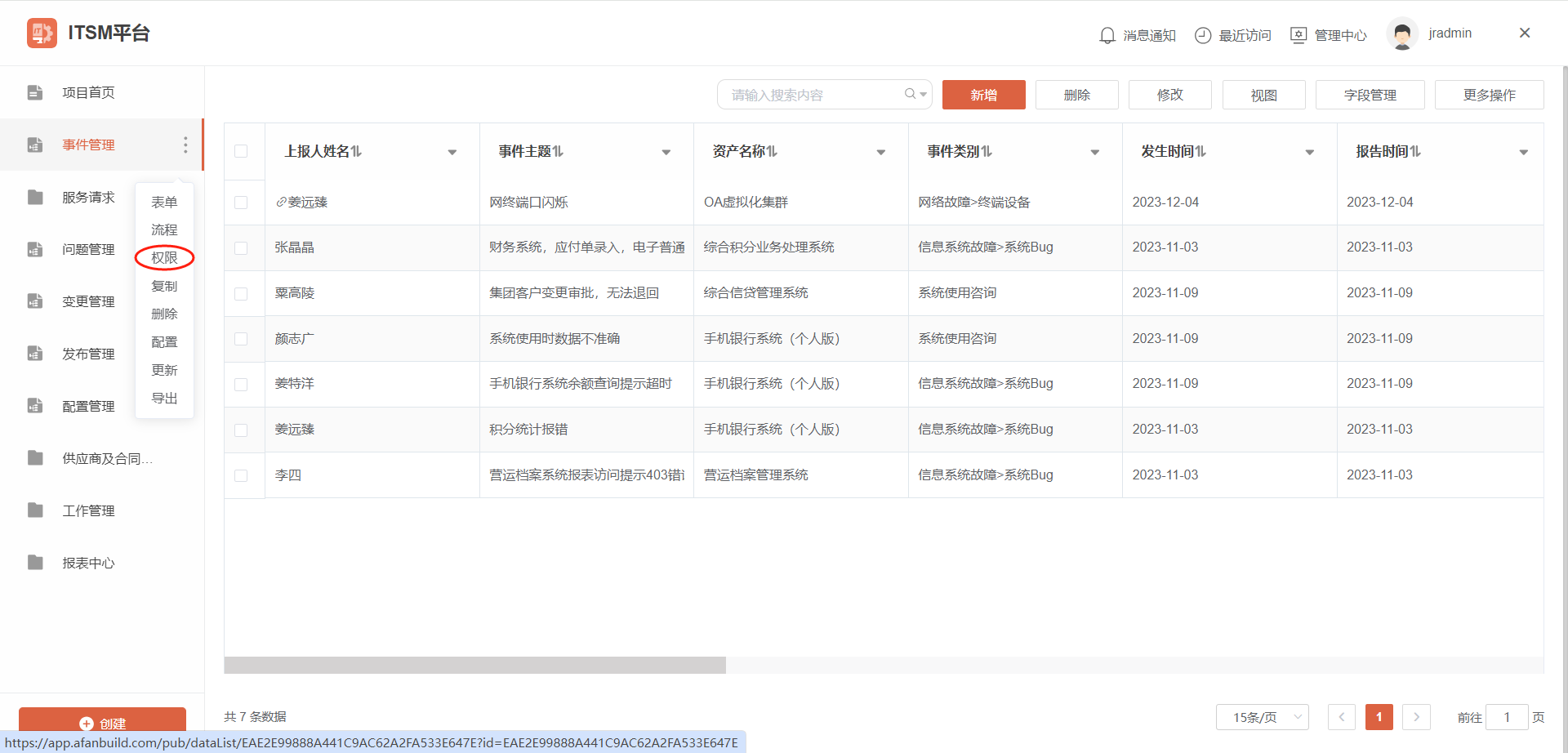


（项目创建示意图）

项目创建完成后，点击该项目即可进入项目空间，在项目空间的左侧底部有一个“创建”按钮，点击该按钮即可按照提示完成功能模块的新增。功能模块新增完成后通过权限分配即可授权他人访问该功能模块了。（详细介绍可参考阿凡搭管理员操作手册）



（功能模块创建按钮）



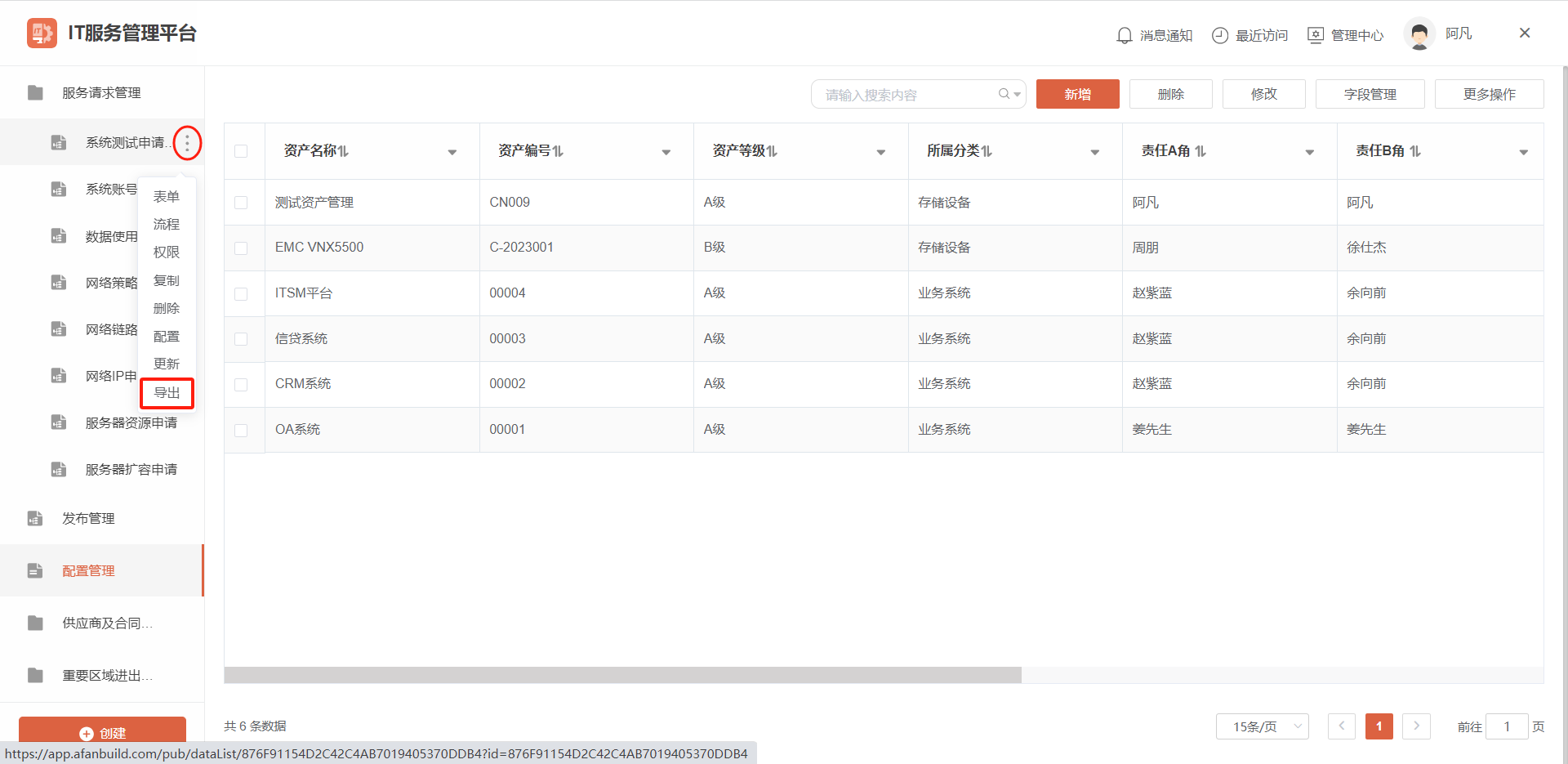
（功能模块权限分配按钮）

## 测试应用转生产环境

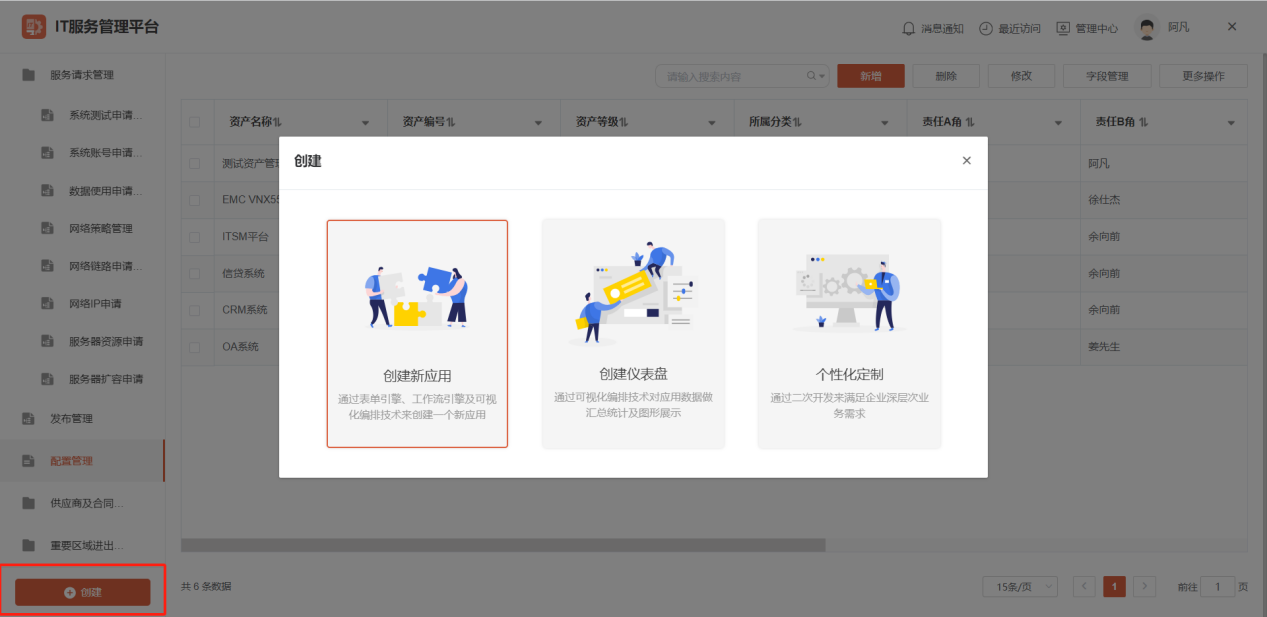
在阿凡搭平台中支持将测试环境的应用导入至生产环境，降低企业应用发布复杂度。

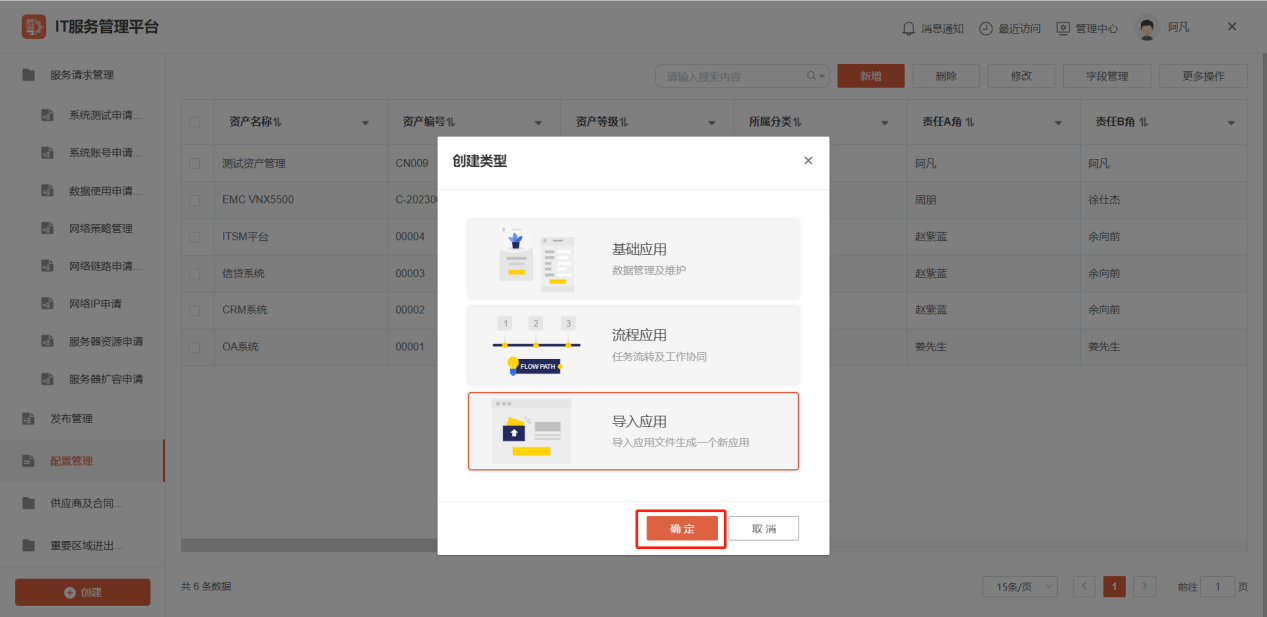
导入应用操作步骤如下：

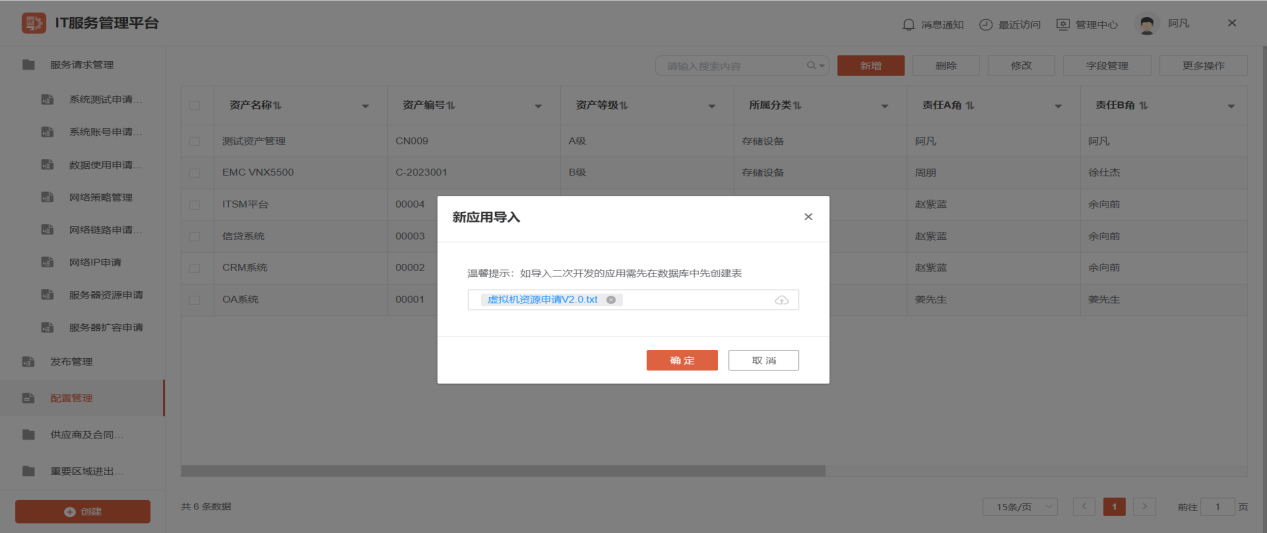
1. 将鼠标移动至应用菜单，点击右侧“三个点”，在下拉菜单中点击“导出”。



1. 点击底部“创建”按钮，然后点击”创建新应用”，然后选中“导入应用”选项并点击“确定”按钮，在弹出框中将刚才导出的文件上传，然后点击“确定”即导入完成。







（完成）。