

# 安装部署手册

## 业务中台之认证授权服务

- 修订历史

版本	作者	日期	备注
v1	刘洪青	2020-06-10	初稿

### 安装部署手册

#### 安装准备

[MySQL 初始配置及相关基础命令](#)

[Harbor 准备及相关说明](#)

[Rancher 准备及相关说明](#)

[域名准备](#)

#### 开始安装

[数据库创建](#)

[rancher 容器部署](#)

[数据配置](#)

## 安装准备

### MySQL 初始配置及相关基础命令

数据文件目录：/var/lib/mysql

- 安装完成后，调整 mysql 服务的配置参数

查看当前配置：show variables;

最大连接数 max\_connections 操作日志的保留时长 binlog\_expire\_logs\_seconds

参考命令：

```
set global max_connections = 1000;
set persist max_connections = 1000;

// 7天 86400 * 7
// 1天 86400
set global binlog_expire_logs_seconds = 86400 * 7;
set persist binlog_expire_logs_seconds = 86400 * 7;
```

#### 时区设置

确保MySQL的时区设置为 GMT+8

- 创建数据库帐号

参考命令：

```
create user 'user'@'%' identified with mysql_native_password by 'your_password';
```

- 创建 database

参考命令：

```
create database `user` DEFAULT CHARSET utf8 COLLATE utf8_general_ci;
```

- 授予权限

将 database 的权限授予对应的帐号

参考命令：

```
grant all privileges on `user`.* to 'user'@'%' with grant option;
```

- 授予 SUPER 权限 由于 部分帐号 需要创建 触发器，故，需要 SUPER 权限 涉及帐号有 user、user\_authz、cas\_server

参考命令：

```
grant SUPER on *.* to 'user'@'%';  
grant SUPER on *.* to 'user_authz'@'%';  
grant SUPER on *.* to 'cas_server'@'%';  
  
grant SUPER on *.* to 'tmp_data'@'%';
```

- 备份与还原

参考命令： 备份：

```
mysqldump -u root -p cas_server > cas_server.sql  
mysqldump -u root -p token_server > token_server.sql  
mysqldump -u root -p user > user.sql  
mysqldump -u root -p user_authz > user_authz.sql  
mysqldump -u root -p agent_service > agent_service.sql
```

还原：

```
mysql -u root -p cas_server < cas_server.sql  
mysql -u root -p token_server < token_server.sql  
mysql -u root -p user < user.sql  
mysql -u root -p user_authz < user_authz.sql  
mysql -u root -p agent_service < agent_service.sql
```

## Harbor 准备及相关说明

- 创建 devops 帐号  
用于 rancher 部署时拉取镜像  
用户管理 下 创建用户 如 devops
- 镜像同步  
从 <https://harbor.supwisdom.com> 中同步镜像  
仓库管理 下 新建目标

```
supwisdom    https://harbor.supwisdom.com    rancher.devops / PWMgP85qiLFC
```

同步管理 下 新建规则

admin-portal	admin-portal/*
authx-service	authx-service/*
thirdparty-agent-service	thirdparty-agent-service/*
user-data-service	goa/*
user-authorization-service	user-authorization-service/*
cas-server	cas-server/*
token-server	token-server/*
jobs-server	jobs-server/*
personal-security-center	personal-security-center/*

同步规则，创建完成后，进行镜像同步

选择某个同步规则，点击 同步，等待任务完成

- 授予 devops 帐号 对各个项目的 访客 权限  
项目 下，点击 项目名称，进入到 成员，添加用户，查找用户 devops，选择角色 访客，确定，添加即可

## Rancher 准备及相关说明

- 创建项目  
进入 全局 - 集群（具体名称视项目安装而定） - 项目/命名空间，添加项目  
输入 项目名称，保存
- 创建命名空间  
进入 全局 - 集群（具体名称视项目安装而定） - 项目/命名空间  
在新建的项目中，添加命名空间  
输入 名称，保存
- 导入YAML

进入 全局 - 集群（具体名称视项目安装而定） - 项目（某个项目）

进入 资源 - 工作负载

## 域名准备

- 确定域名

首先明确是否使用泛域名，如：`*.paas.xxx.edu.cn`，或 直接使用学校域名 `xxx.edu.cn`

本产品安装需要的域名如下：

<code>cas.paas.xxx.edu.cn</code>	认证（视具体情况，可调整）
<code>token.paas.xxx.edu.cn</code>	认证（APP适用）
<code>personal-security-center.paas.xxx.edu.cn</code>	个人安全中心后端API
<code>security-center.paas.xxx.edu.cn</code>	安全中心前端UI（帐号激活、忘记密码）

如果使用 学校域名，则去除 `.paas` 即可，同时申请开通相关域名

## 开始安装

### 数据库创建

- 数据库帐号

以下是 各服务对应的数据库帐号

服务	数据库帐号
用户服务 <code>user-data-service</code>	<code>user</code>
授权服务 <code>user-authorization-service</code>	<code>user_authz</code>
认证服务 <code>cas-server</code>	<code>cas_server</code>
认证服务（APP适用） <code>token-server</code>	<code>token_server</code>
-	-
第三方代理服务 <code>thridparty-agent-service</code>	<code>agent_service</code>
-	-
v4认证迁移数据	<code>tmp_data</code>

命令： 请修改命令中的 `your_password` 为实际的数据库帐号的密码

```
create user 'user'@'%' identified with mysql_native_password by 'your_password';
create user 'user_authz'@'%' identified with mysql_native_password by
'your_password';
create user 'cas_server'@'%' identified with mysql_native_password by
'your_password';
create user 'token_server'@'%' identified with mysql_native_password by
'your_password';

create user 'agent_service'@'%' identified with mysql_native_password by
'your_password';

create user 'tmp_data'@'%' identified with mysql_native_password by
'your_password';
```

- 数据库

以下是 各服务对应的数据库

服务	数据库
用户服务 user-data-service	user
授权服务 user-authorization-service	user_authz
认证服务 cas-server	cas_server
认证服务（APP适用） token-server	token_server
-	-
第三方代理服务 thridparty-agent-service	agent_service
-	-
v4认证迁移数据	tmp_data

命令：

```
create database `user` DEFAULT CHARSET utf8 COLLATE utf8_general_ci;
create database `user_authz` DEFAULT CHARSET utf8 COLLATE utf8_general_ci;
create database `cas_server` DEFAULT CHARSET utf8 COLLATE utf8_general_ci;
create database `token_server` DEFAULT CHARSET utf8 COLLATE utf8_general_ci;

create database `agent_service` DEFAULT CHARSET utf8 COLLATE utf8_general_ci;

create database `tmp_data` DEFAULT CHARSET utf8 COLLATE utf8_general_ci;
```

- 数据库权限授予

将 database 的权限授予对应的帐号

命令：

```
grant all privileges on `user`.* to 'user'@'%' with grant option;
grant all privileges on `user_authz`.* to 'user_authz'@'%' with grant option;
grant all privileges on `cas_server`.* to 'cas_server'@'%' with grant option;
grant all privileges on `token_server`.* to 'token_server'@'%' with grant option;

grant all privileges on `agent_service`.* to 'agent_service'@'%' with grant
option;

grant all privileges on `tmp_data`.* to 'tmp_data'@'%' with grant option;
```

- SUPER 权限授予

由于 部分帐号 需要创建 触发器，故，需要 SUPER 权限 涉及帐号有 user、user\_authz、cas\_server

命令：

```
grant SUPER on *.* to 'user'@'%;
grant SUPER on *.* to 'user_authz'@'%;
grant SUPER on *.* to 'cas_server'@'%;

grant SUPER on *.* to 'tmp_data'@'%;
```

- 用户数据的交换帐号

#### 待部署完成后操作

如果，存在数据交换 须将组织机构数据、帐号数据 同步到用户服务的数据库的 则，需要创建一个交换用的数据库帐号（user\_trans），并为该帐号授予 表 user.TMP\_ORGANIZATION\_ORIGIN、user.TMP\_ACCOUNT\_ORIGIN 的读写操作的权限

命令：

```
create user 'user_trans'@'%' identified with mysql_native_password by
'your_password';

grant select on `user`.`TMP_ORGANIZATION_ORIGIN` to 'user_trans'@'%;
grant insert on `user`.`TMP_ORGANIZATION_ORIGIN` to 'user_trans'@'%;
grant update on `user`.`TMP_ORGANIZATION_ORIGIN` to 'user_trans'@'%;
grant delete on `user`.`TMP_ORGANIZATION_ORIGIN` to 'user_trans'@'%;

grant select on `user`.`TMP_ACCOUNT_ORIGIN` to 'user_trans'@'%;
grant insert on `user`.`TMP_ACCOUNT_ORIGIN` to 'user_trans'@'%;
grant update on `user`.`TMP_ACCOUNT_ORIGIN` to 'user_trans'@'%;
grant delete on `user`.`TMP_ACCOUNT_ORIGIN` to 'user_trans'@'%;

grant select on `user`.`TMP_ORGANIZATION_TRANS` to 'user_trans'@'%;
grant insert on `user`.`TMP_ORGANIZATION_TRANS` to 'user_trans'@'%;
grant update on `user`.`TMP_ORGANIZATION_TRANS` to 'user_trans'@'%;
grant delete on `user`.`TMP_ORGANIZATION_TRANS` to 'user_trans'@'%;
```

```
grant select on `user`.`TMP_ACCOUNT_TRANS` to 'user_trans'@'%';
grant insert on `user`.`TMP_ACCOUNT_TRANS` to 'user_trans'@'%';
grant update on `user`.`TMP_ACCOUNT_TRANS` to 'user_trans'@'%';
grant delete on `user`.`TMP_ACCOUNT_TRANS` to 'user_trans'@'%';
```

## rancher 容器部署

- 修改 yaml 中的相关配置

具体参考 yaml 文件中的说明

0.infras

基础设施，目前包含 MySQL数据库的Web管理端、SpringBoot服务的管理端

0.0.0.infras-base.yaml	请修改 harbor-registry 的帐号密码
0.0.1.infras-mysql.yaml 访问域名	请修改 MySQL数据库 的地址、IP, mysql-adminer
0.0.2.infras-sba.yaml	请修改 docker 镜像地址

1.authx-service

业务中台 之 认证授权服务

参考 yaml 中的说明，修改相关配置

在各个服务的安装脚本目录下，修改以下文件（若存在）中的配置

0.*-base.yaml	请修改 harbor-registry 的帐号密码
4.x.*.yaml, 5.*-datax-job.yaml	请修改 docker 镜像地址
1.*-env.yaml, 5.*-datax-job.yaml	请修改 数据库密码
2.*-ingresses.yaml	请修改 访问域名

0.0.trans-service-v4

此为 认证v4 的数据迁移服务（可选）

将 认证v4 的数据导入到 tmp\_data 下

数据迁移后，还需要手动编写脚本，将数据迁移至 用户服务、授权服务 的数据库中

0.authx-service

此为 公共基础服务

如：MySQL 服务地址（Endpoints）、文件存储服务

#### 1.authx-service-mysql.yaml

请修改 mysql 的服务地址 IP

#### 2.authx-service-minio.yaml

请修改 minio 的 `MINIO\_ACCESS\_KEY`、`MINIO\_SECRET\_KEY`

根据情况修改 pvc 的 storageClassName

#### 9.poa-api-docs\_install.yaml

用于将 认证授权服务的 poa 接口文档，导入到 poa-sa 中，\*\*请在 poa 安装完成后处理\*\*

请修改 poa 的服务地址 `POA\_SERVER\_URL`

#### 1.thirdparty-agent-service

此为 第三方服务的代理服务

file-minio

修改 minio 的 `FILE\_MINIO\_ACCESSKEY`、`FILE\_MINIO\_SECRETKEY`

mail-smtp

获取 学校的 smtp 服务地址，邮箱帐号，用于发送邮件

sms-aliyun

如果 学校使用 阿里云的短信服务，提供 `ACCESS\_KEY\_ID`、`ACCESS\_SECRET`；  
否则，提供相关的短信平台，进行定制开发

#### 2.user-data-service

此为 用户服务

goa

如果 须将用户数据的变更下发到 Openldap 等第三方业务中，则须配置 `JOBS\_RABBITMQ\_\*`  
为开启 (ENABLED=true)

#### 3.user-authorization-service

此为 授权服务

#### 4.cas-server



此为 认证服务

cas-server-site-webapp

生成公私钥证书，参考 certs/jwt/readme.md 生成公私钥pem，修改相关配置  
`CASSERVER\_JWT\_PRIVATE\_KEY\_PEM\_PKCS8`、`CASSERVER\_JWT\_PUBLIC\_KEY\_PEM`

修改 认证服务的外网访问地址 `CAS\_SERVER\_NAME`

修改 CAT TGC 的安全，若 使用 https，则须修改 `CAS\_TGC\_SECURE: "true"`

修改 安全中心（帐号激活、找回密码）的链接地址  
`CASSERVERSITE\_FORGOT\_PASSWORD\_URL`、`CASSERVERSITE\_ACTIVE\_ACCOUNT\_URL`

联合登录（QQ、微信、企业微信、支付宝等）配置 `CASSERVER\_FEDERATION\_\*`

动态密码认证 相关配置

1. 短信模板（动态密码） `CASSERVERSITE\_PASSWORDLESS\_SMS\_TEXT\_TEMPLATE`
2. 短信接口地址 `TPAS\_AGENT\_SERVICE\_SMS\_SENDER\_PATH`

如果 须与 超级APP 对接，须修改 Token 验签公钥地址  
`SUPERAPP\_TOKEN\_SIGNING\_KEY\_URL`

如果 须开启图片验证码，修改 `CASSERVERSITE\_CAPTCHA\_ENABLED: "true"`

## 5.token-server

此为 认证服务（适用于APP，可选）

token-server

生成公私钥证书（与cas-server保持一致），参考 certs/jwt/readme.md 生成公私钥pem，修改相关配置 `TOKEN\_SERVER\_SECURITY\_JWT\_PRIVATE\_KEY\_PEM\_PKCS8`、  
`TOKEN\_SERVER\_SECURITY\_JWT\_PUBLIC\_KEY\_PEM`

修改 认证服务的外网访问地址 `TOKEN\_SERVER\_PREFIX`

修改 认证服务 Id-Token 的签发者标识 `TOKEN\_SERVER\_SECURITY\_JWT\_ISS`

动态密码认证 相关配置（与cas-server保持一致）

1. 短信模板（动态密码） `TOKEN\_SERVER\_PASSWORDLESS\_SMS\_TEXT\_TEMPLATE`
2. 短信接口地址 `TPAS\_AGENT\_SERVICE\_SMS\_SENDER\_PATH`

人脸认证，须配置人脸服务，目前支持 新开普人脸服务、百度人脸服务，根据情况获取相关配置参数

APP 登录信息 个推，使用了消息服务的接口，该接口由 POA 提供，故须

1. 注册 POA client，获取 `clientId`、`clientSecret`，申请 Scope  
`messagecenter:v1:sendMessage`

## 2. 获取 消息服务的 `appId`

### 6.personal-security-center

此为 个人安全中心 后端API, 安全中心 前端UI

提供个人帐号相关的操作的接口, 以及 帐号激活、密码找回 等功能

TODO: 修改 bff、zuul 配置

TODO: 修改 security-center-ui 配置

### 9.jobs-server

此为 任务调度服务

基于 定时任务、触发任务 等, 完成 用户数据的同步

如:

- \* 源头数据进入到临时表后, 写入用户的正式表
- \* 用户数据更新后, 通过消息队列, 增量更新 Openldap 数据

## ● 添加项目、命名空间

### 项目

infras	# 基础设施 (可选, 方便实施工作)
authx-service	# 认证授权服务
admin-platform	# 管理平台

### 命名空间

在项目 infras 下创建 命名空间:

base

在项目 authx-service 下创建 命名空间:

trans-service (认证v4的数据迁移服务, 可选)

authx-service

thirdparty-agent-service

user-data-service

```
user-authorization-service

cas-server

token-server

personal-security-center

jobs-server
```

- 导入YAML

在项目 infras 中，将 0.infras 下的 yaml 按编号依次导入

```
0.0.0.infras-base.yaml

0.0.1.infras-mysql.yaml          mysql web管理

0.0.2.infras-sba.yaml
```

在项目 authx-service 中，将 1.authx-service 下的 yaml 按编号依次导入

务必确保 **4.0.\*-installer.yaml** 执行成功

## 数据配置

数据脚本初始化

先修改 脚本中的域名（如果存在）

- 可选，1.authx-service/10.0.tmp.sql  
若通过交换同步组织机构、帐号数据的，须执行该数据库脚本
- 可选，1.authx-service/10.1.init-flow.sql  
若部署了 流程平台 的产品  
可默认创建几个管理员帐号，以及初始授权