# 应用环境介绍

## 应用路径

/www/cgi/tickets\_server

## 运行环境

Python 3.6、MongoDB

cryptography==2.7

openpyxl==2.6.2

aiohttp==3.5.4

APScheduler==3.6.0

motor==2.0.0

PyYAML==5.1.1

## 守护程序

Supervisor

# 应用参数配置

## 守护程序配置

/etc/supervisor/conf.d/ticket\_server.conf

command=/root/.pyenv/shims/python3.7 /www/cgi/tickets\_server/tickets\_polls/main.py

directory=/www/cgi/tickets\_server

autostart=true

## 数据库参数配置

默认参数，只允许本地访问，无密码

# 数据备份与恢复

## 数据备份

通过公司内服可以正常联网的服务器，定时访问远程服务器，备份数据库并取回本地。

mongodump -d ticket --archive=/tmp/ticket\_db\_xxx.dump

scp /tmp/ticket\_db\_xxx.dump ticket\_db\_xxx.dump

## 数据恢复

上传备份数据到远程服务器，解压数据备份文件，恢复数据库。

scp ticket\_db\_xxx.dump /tmp/ticket\_db\_xxx.dump

mongorestore -d ticket --archive=/tmp/ticket\_db\_xxx.dump

# 异常状况处理方法

## 应用宕机检查项目

1. 通过ssh连接服务器，检查服务器是否可用，
2. 检查守护程序、应用程序、数据库程序是否正常运行
3. 通过守护程序命令supervisorctl status 检查应用运行状态
4. 检查运行日志查找宕机原因，日志路径 /var/log/supervisor/ticket\_server\_stderr\_xxx.log

## 服务器无法连接

通过服务器供应商提供的管理页面启动服务器，启动成功后一段时间内所有服务应正常启动。

## 守护服务未启动

使用 system supervisord start 拉起supervisord 服务后，system supervisord status查看其是否启动成功，如未启动根据日志排查具体问题。

## 数据库无法连接

使用 systemctl mongod start启动mongodb服务，查询是否成功启动，如启动失败应根据日志进行排查。