

系统时区配置:

### 1. 查看时区

```
[root@guoxiaogang ~]# more /etc/sysconfig/clock
ZONE="Asia/Shanghai"
UTC=true
ARC=false
[root@guoxiaogang ~]#
```

### 2. 修改时区

```
[root@guoxiaogang ~]# ls /usr/share/zoneinfo/
Africa      Brazil      Egypt      GB-Eire     HST         Japan      MST7MDT     posix       Singapore   WET
America     Canada     Eire       GMT         Iceland     Kwajalein  Navajo      posixrules  Turkey      W-SU
Antarctica  CET        EST        GMT0        Indian      leapseconds  NZ          PRC         tzdata.zi   zone1970.tab
Arctic      Chile      EST5EDT    GMT-0        Iran        Libya      NZ-CHAT     PST8PDT     UCT         zone.tab
Asia        CST6CDT    Etc        GMT+0        iso3166.tab  MET         Pacific     right       Universal   Zulu
Atlantic    Cuba       Europe     Greenwich   Israel      Mexico     Poland      ROC         US
Australia   EET        GB         Hongkong    Jamaica     MST         Portugal    ROK         UTC

[root@guoxiaogang ~]# cp /usr/share/zoneinfo/Asia/Shanghai /etc/localtime
```

### 3 安装 mysql

下载 MySQL 数据脚本（如下有安装 MySQL 服务可以忽略）

```
[hadoop@guoxiaogang data]$ ll
total 56
drwxrwxr-x 2 hadoop hadoop 4096 Mar 17 00:31 azk
-rw-r--r-- 1 hadoop hadoop 1928 Sep 1 2017 azkaban-sql-script-2.5.0.tar.gz
drwxrwxr-x 4 hadoop hadoop 4096 Oct 7 03:23 dfs
```

下载然后解压; tar -zxvf azkaban-sql-script-2.5.0.tar.gz

```
[hadoop@guoxiaogang azkaban-2.5.0]$ ls
create.active_executing_flows.sql  create.project_files.sql          create.schedules.sql
create.active_sla.sql              create.project_flows.sql          create.triggers.sql
create-all-sql-2.5.0.sql           create.project_permissions.sql    database.properties
create.execution_flows.sql         create.project_properties.sql     update-all-sql-2.1.sql
create.execution_jobs.sql          create.projects.sql              update-all-sql-2.2.sql
create.execution_logs.sql          create.project_versions.sql       update.execution_logs.2.1.sql
create.project_events.sql          create.properties.sql            update.project_properties.2.1.sql
```

创建用户、分配权限并执行脚本

mysql -u root -pxxxx

创建数据库: CREATE DATABASE azkaban;

创建好数据库然后退出。

然后将 sql-script 文件中的 create-all-sql-2.5.0.sql 中的数据表创建在刚创建好的 azkaban 数据库当中。

使用命令: mysql -u root -pxxxx -Dazkaban < /绝对路径/create-all-sql-2.5.0.sql

登录 mysql 数据库。mysql -u root -pxxx 查看存在的表。

```
total 88
-rwxr-xr-x 1 hadoop hadoop 129 Apr 21 2014 create.active_executing_flows.sql
-rwxr-xr-x 1 hadoop hadoop 216 Apr 21 2014 create.active_sla.sql
-rwxr-xr-x 1 hadoop hadoop 4694 Apr 21 2014 create-all-sql-2.5.0.sql
-rwxr-xr-x 1 hadoop hadoop 610 Apr 21 2014 create.execution_flows.sql
-rwxr-xr-x 1 hadoop hadoop 519 Apr 21 2014 create.execution_jobs.sql
-rwxr-xr-x 1 hadoop hadoop 358 Apr 21 2014 create.execution_logs.sql
```

注意: 这里我们使用的 root 用户创建的数据库。在生产上我们是单独的创建一个用户, 方便管理使用;

还需注意，在 azkaban2.5 以后不需要自己在引入 mysql 的驱动包。在 lib 目录下面存在驱动包。

创建用户名: CREATE USER 'azkaban'@'%' IDENTIFIED BY 'azkaban'

分配权限: GRANT SELECT,INSERT,UPDATE,DELETE,CREATE,INDEX ON `azkaban`.\* TO 'azkaban'@'%' WITH GRANT OPTION;

flush privileges;

quit;

通过 azkaban 账号登录 mysql: mysql -uazkaban -pazkaban

Use azkaban;

source ~/azkaban/azkaban-2.5.0/create-all-sql-2.5.0.sql

#### 4 Azkaban Web Server 安装

##### 1 下载安装 azkaban-web-server-2.5.0.tar.gz

解压 tar 包。目录如下

```
[hadoop@guoxiaogang azkaban-web-2.5.0]$ ls
azkaban.version bin conf currentpid extlib lib plugins web
[hadoop@guoxiaogang azkaban-web-2.5.0]$
```

↑ ↑ ↑ ↑ ↑  
执行脚本 配置文件 可扩展的jar包 插件 web端配置

##### 2 生成 SSL 证书

因为 azkaban 是加密传输，因此要配置 ssl 服务。方便网路传输。

关于怎么使用 Java keytool 生成 keystore 文件 在这里可以只简单的生成 keystore 文件，并将生成的 keystore 文件拷贝至 ~/azkaban-web-2.5.0/web 文件下。

```
keytool -keystore keystore -alias jetty -genkey -keyalg RSA
```

```
[hadoop@guoxiaogang web]$ keytool -keystore keystore -alias jetty -genkey -keyalg RSA
Enter keystore password: 
Re-enter new password: 
What is your first and last name? 
[Unknown]
What is the name of your organizational unit? 
[Unknown]
What is the name of your organization? 
[Unknown]
```

↑ ↑  
文件名称 别名

必须是6为以上

这些地方可以不填

#### Jetty Parameters

Parameter	Description	Default
jetty.maxThreads	Max request threads	25
jetty.ssl.port	The ssl port	8443
jetty.keystore	The keystore file	
jetty.password	The jetty password	
jetty.keypassword	The keypassword	
jetty.truststore	The trust store	
jetty.trustpassword	The trust password	

描述不清楚，建议密码设置一样。避免麻烦

### 3 修改 ./conf/azkaban.properties

修改首页样式及时区信息

```
azkaban.name=Job Tasks
```

```
azkaban.label=mysteel workflow
```

```
azkaban.color=#FF3601
```

```
azkaban.default.servlet.path=/index
```

```
web.resource.dir=web/
```

```
default.timezone.id=Asia/Shanghai
```

```
azkaban.properties azkaban-users.xml
[hadoop@guoxiaogang conf]$ cat azkaban.properties
#Azkaban Personalization Settings
azkaban.name=Test 名字自定义
azkaban.label=My Local Azkaban
azkaban.color=#FF3601
azkaban.default.servlet.path=/index 绝对路劲加能够加载到web页面信息
web.resource.dir=/home/hadoop/app/azkaban-web-2.5.0/web/
default.timezone.id=Asia/Shanghai 最好绝对路劲，防止找不到文件
```

修改 mysql 连接,可自行修改

```
database.type=mysql
```

```
mysql.port=3306
```

```
mysql.host=192.168.200.184
```

```
mysql.database=azkaban
```

```
mysql.user=azkaban
```

```
mysql.password=azkaban
```

```
mysql.numconnections=100
```

```
database.type=mysql 数据库类型mysql，这个是阿兹卡班预留给后面支持
mysql.port=3306 的数据库设置的配置。
mysql.host=192.168.118.136
mysql.database=azkaban
mysql.user=root 这个地方是刚才创建的数据库，mysql使用的是用户是root，生产上设置单独的用户
mysql.password=root
mysql.numconnections=100
```

修改 Jetty 服务器属性，包括 keystore 的相关配置

jetty.maxThreads=25

jetty.ssl.port=8443

jetty.port=8081

jetty.keystore=keystore 最好设置为绝对路劲，防止加载不到

jetty.password=mysteel

jetty.keypassword= mysteel

jetty.truststore=keystore 最好设置为绝对路劲，防止加载不到

jetty.trustpassword= mysteel

```
# Azkaban Jetty server properties.
jetty.maxThreads=25      最大线程数
jetty.ssl.port=8443      ssl加密传输的端口号，默认是8443，也是web登录的端口号
jetty.port=8081
jetty.keystore=/home/hadoop/app/azkaban-web-2.5.0/web/keystore 这个就是刚才设置的ssl生成的文件路劲
jetty.password=guoxiaogang
jetty.keypassword=guoxiaogang
jetty.truststore=/home/hadoop/app/azkaban-web-2.5.0/web/keystore
jetty.trustpassword=guoxiaogang 这三个密码就是创建ssl加密文件设置的密码，为了方便记忆一定要设置一样
```

修改邮件设置（可选）

mail.sender=xxxxxxx@mysteel.com

mail.host=smtp.126.com

mail.user= xxxxxx@mysteel.com

mail.password=xxx（注：发送邮箱的密码）

## 5.用户及权限配置

进入 azkaban web 服务器 conf 目录，修改 azkaban-users.xml ， 增加管理员用户：

< azkaban-users>

< user username="azkaban" password="azkaban" roles="admin" groups="azkaban" />

< user username="metrics" password="metrics" roles="metrics"/>

< user username="admin" password="admin" roles="admin,metrics" />

< role name="admin" permissions="ADMIN" />

< role name="metrics" permissions="METRICS"/>< /azkaban-users>

```

azkaban.properties azkaban-users.xml
[hadoop@guoxiaogang conf]$ cat azkaban-users.xml
<azkaban-users>
  <user username="azkaban" password="azkaban" roles="admin" groups="azkaban" />
  <user username="metrics" password="metrics" roles="metrics" />
  <role name="admin" permissions="ADMIN" />
  <role name="metrics" permissions="METRICS" />
</azkaban-users>
[hadoop@guoxiaogang conf]$

```

这两个是在阿兹卡班页面登录的用户名称和相对应的密码设置，可以设置多用户登录页面

## Azkaban Executor Server 安装

### 1. 下载安装包

azkaban-executor-server-2.5.0.tar.gz

解压到相对应的文件目录下。

```
tar -zxvf azkaban-executor-server-2.5.0.tar.gz -C ~/azkaban/
```

### 2. 修改配置文件 (conf/azkaban.properties)

修改时区信息

default.timezone.id=Asia/Shanghai

修改 mysql 连接, 可自行修改

database.type=mysql

mysql.port=3306

mysql.host=192.168.200.184

mysql.database=azkaban

mysql.user=azkaban

mysql.password=azkaban

mysql.numconnections=100

```

database.type=mysql
mysql.port=3306
mysql.host=guoxiaogang
mysql.database=azkaban
mysql.user=root
mysql.password=root
mysql.numconnections=100

```

这个地方和web server的连接的数据库必须一致，因为两者都是通过数据库进行交互的，像定时任务，还有一些工作流的执行，都是通过mysql。所以设置一致

配置端口，默认就好

executor.maxThreads=50

executor.port=12321

```
executor.flow.threads=30
```

```
# Azkaban Executor settings
executor.maxThreads=50
executor.port=12321
executor.flow.threads=30
```

这里默认设置的线程数是最大50个，工作流线程数最大是30  
另外，这个端口必须设置，他是和web端进行交互的唯一入口。在web端conf文件中必须也要设置这个端口。默认12321

在 web server 的 azkaban.properties 配置 executor 的连接，特别注意，如果两个 server 不是在一台主机上，web server 需要配置 executor.host

```
# Azkaban Executor settings
```

```
executor.port=12321
```

```
executor.host=ip/hostname
```

```
jetty.trustpassword=guoxiaogang
```

```
# Azkaban Executor settings
executor.port=12321
```

这个是web端的设置，必须指定如果和executor不在一台机器上，必须指定主机名

```
# mail settings
mail sender=
```

启动验证

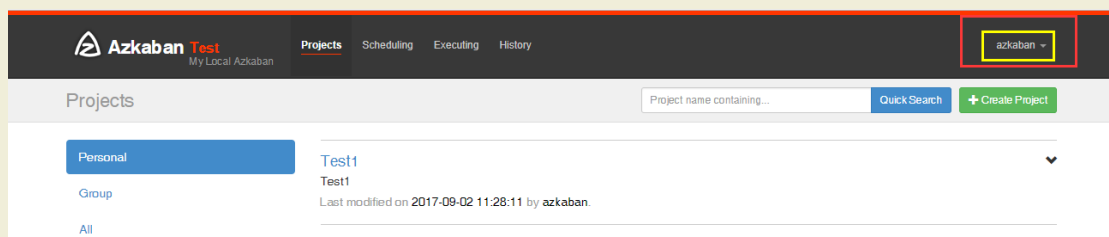
启动 web server

~/azkaban/azkaban-web-2.5.0/bin/azkaban-web-start.sh (azkaban-web-shutdown.sh 停止)

访问 <https://ip:8443>

登录用户名称和密码是上面 xml 文件中的用户名称和密码，也可自行配置。

```
ons":{"failureAction":"FINISH_CURRENTLY_RUNNING","queueLevel":0,"pipelineExecId":null,"concurrentOption":"ignore","notifyOnFirstFailure":false,"notifyOnLastFailure":false,"pipelineLevel":null,"successEmailsOverride":false,"failureEmails":[],"disabled":[],"flowParameters":{},"successEmails":[],"mailCreator":"default","failureEmailsOverride":false,"type":"ExecuteFlowAction","projectId":"hadoop","projectId":"14","flowName":"bar"},"type":"ExecuteFlowAction"},"expireCondition":{"nextCheckTime":"1522307460000","expression":"BasicTimeChecker_2.eval()","checkers":[{"checkerJson":{"nextCheckTime":"1522307460000","period":"1d","skipPastChecks":true,"timezone":"Asia/Shanghai","firstCheckTime":"1522307460000","isRecurring":true,"id":"BasicTimeChecker_2","type":"BasicTimeChecker"},"type":"BasicTimeChecker"}],"status":"READY","info":{}}
2018/04/01 12:54:42.065 +0800 INFO [JdbcTriggerLoader] [Azkaban] Loaded 1 triggers.
2018/04/01 12:54:42.065 +0800 INFO [log] [Azkaban] jetty-6.1.26
2018/04/01 12:54:42.078 +0800 INFO [TriggerManager] [Azkaban] Checking trigger 2
2018/04/01 12:54:42.078 +0800 INFO [Condition] [Azkaban] Testing condition BasicTimeChecker_1.eval()
2018/04/01 12:54:42.118 +0800 INFO [Condition] [Azkaban] Testing condition BasicTimeChecker_2.eval()
2018/04/01 12:54:42.999 +0800 INFO [log] [Azkaban] Started SslSocketConnector@0.0.0.0:8443
2018/04/01 12:54:42.999 +0800 INFO [AzkabanWebServer] [Azkaban] Server running on ssl port 8443. 启动成功
```



出现界面安装成功

启动 executor server

~/azkaban/azkaban-executor-2.5.0/bin/azkaban-exec-start.sh(azkaban-executor-shutdown.sh 停止

```
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
2018/04/01 12:56:51.712 +0800 INFO [log] [Azkaban] Logging to org.slf4j.impl.Log4jLoggerAdapter(org.mortbay.log) via org.mortbay.log.Slf4jLog
2018/04/01 12:56:51.989 +0800 INFO [FlowRunnerManager] [Azkaban] Execution dir retention set to 86400000 ms
2018/04/01 12:56:52.101 +0800 INFO [FlowRunnerManager] [Azkaban] Cleaning recently finished
2018/04/01 12:56:52.102 +0800 INFO [FlowRunnerManager] [Azkaban] Cleaning old projects
2018/04/01 12:56:52.102 +0800 INFO [FlowRunnerManager] [Azkaban] Cleaning old execution dirs
2018/04/01 12:56:52.111 +0800 INFO [AzkabanExecutorServer] [Azkaban] Registering MBeans...
2018/04/01 12:56:52.142 +0800 INFO [AzkabanExecutorServer] [Azkaban] Bean azkaban.jmx.JmxJettyServer registered.
2018/04/01 12:56:52.157 +0800 INFO [AzkabanExecutorServer] [Azkaban] Bean azkaban.jmx.JmxFlowRunnerManager registered.
2018/04/01 12:56:52.157 +0800 INFO [log] [Azkaban] jetty-6-1.26
2018/04/01 12:56:52.316 +0800 INFO [log] [Azkaban] Started SocketConnector@0.0.0.0:12321 executor启动成功
2018/04/01 12:56:52.316 +0800 INFO [AzkabanExecutorServer] [Azkaban] Azkaban Executor Server started on port 12321
```

至此 azkaban 安装完成。