

XXL-JOB DEPLOY

Linux



2017-10-19

浙江元亨通信技术股份有限公司

http://uhope.com

**解释权**

文档摘要

|  |  |
| --- | --- |
| 文件名 | xxl-job-deploy.docx |
| 最新版本 | 1.0.0 |
| 文档创建者 | 蒋家亮 |
| 文档维护者 |  |
| 文档评审者 |  |
| 定稿日期 |  |
| 文档目标 |  |

版本管理

|  |  |  |  |
| --- | --- | --- | --- |
| **版本** | **描述** | **日期** | **作者** |
| 1.0.0 | 创建 | 2017-10-19 | 蒋家亮 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目 录

[1 Windows环境打包 4](#_Toc496282751)

[1.1 xxl-job-admin打war包 4](#_Toc496282752)

[1.1.1 cmd命令 4](#_Toc496282753)

[1.1.2 maven命令 4](#_Toc496282754)

[1.2 xxl-job-executor打jar包 6](#_Toc496282755)

[1.2.1 cmd命令 6](#_Toc496282756)

[1.2.2 maven命令 7](#_Toc496282757)

[2 Linux服务器部署 8](#_Toc496282758)

[2.1 xxl-job-admin部署 9](#_Toc496282759)

[2.1.1 解压war包 9](#_Toc496282760)

[2.1.1.1 切换指定目录 9](#_Toc496282761)

[2.1.1.2 解压war包 9](#_Toc496282762)

[2.1.2 修改配置信息 10](#_Toc496282763)

[2.1.2.1 切换到“xxl-job-admin/WEB-INF/classes/spring”目录下 10](#_Toc496282764)

[2.1.2.2 编辑applicationcontext-xxl-job-admin.xml配置文件 10](#_Toc496282765)

[2.1.2.3 查看“applicationcontext-xxl-job-admin.xml”配置文件中的 11](#_Toc496282766)

[2.1.2.3.1 第一步 11](#_Toc496282767)

[2.1.2.3.2 第二步 11](#_Toc496282768)

[2.1.2.3.3 第三步 12](#_Toc496282769)

[2.1.2.3.4 第四步 13](#_Toc496282770)

[2.1.2.3.5 第五步 13](#_Toc496282771)

[2.1.3 启动Tomcat 13](#_Toc496282772)

[2.1.3.1 第一步：检查Tomcat是否已启动 13](#_Toc496282773)

[2.1.3.2 第二步：杀掉Tomcat进程 13](#_Toc496282774)

[2.1.3.3 第三步：启动Tomcat 13](#_Toc496282775)

[2.1.3.4 第四步：查看日志 14](#_Toc496282776)

[2.1.3.5 第五步：浏览器访问 14](#_Toc496282777)

[2.1.4 运行异常 15](#_Toc496282778)

[2.1.4.1 异常信息 15](#_Toc496282779)

[2.1.4.1.1 异常原因 15](#_Toc496282780)

[2.1.4.1.2 解决办法 16](#_Toc496282781)

[2.2 xxl-job-executor部署 17](#_Toc496282782)

[2.2.1.1 切换指定目录 17](#_Toc496282783)

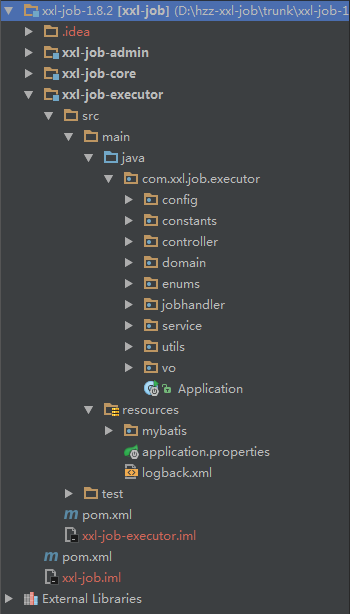
[2.2.1.2 运行jar包 18](#_Toc496282784)

[2.2.1.2.1 执行jar包运行命令 18](#_Toc496282785)

[2.2.1.2.2 查看jar包进程信息命令 18](#_Toc496282786)

[2.2.1.2.3 关闭jar包运行 19](#_Toc496282787)

**项目结构**



环境：

* Windows 7 64bit
* JDK 1.7.0\_79
* Tomcat 7.0.56
* Maven 3.2.5
* MySQL 5.6+ / MongoDB
* XXL-JOB 1.8.2
* Spring Boot 1.5.6
* IntelliJ IDEA 15.0.2
* xxl-job-admin（调度器）
* xxl-job-core（公共依赖包）
* xxl-job-executor（执行器–Spring Boot）

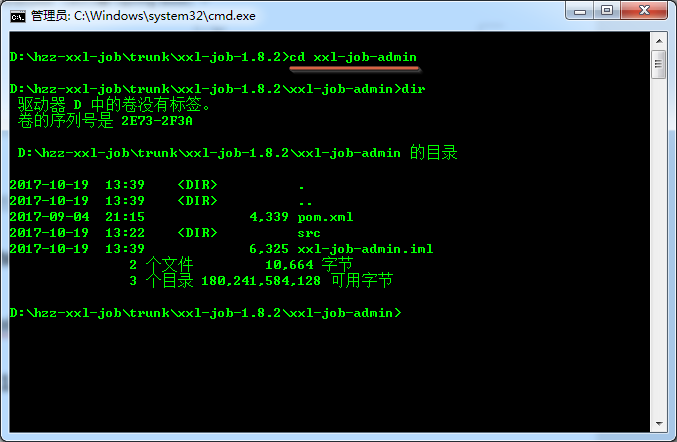
# Windows环境打包

## xxl-job-admin打war包

### cmd命令

切换到xxl-job-admin项目根目录

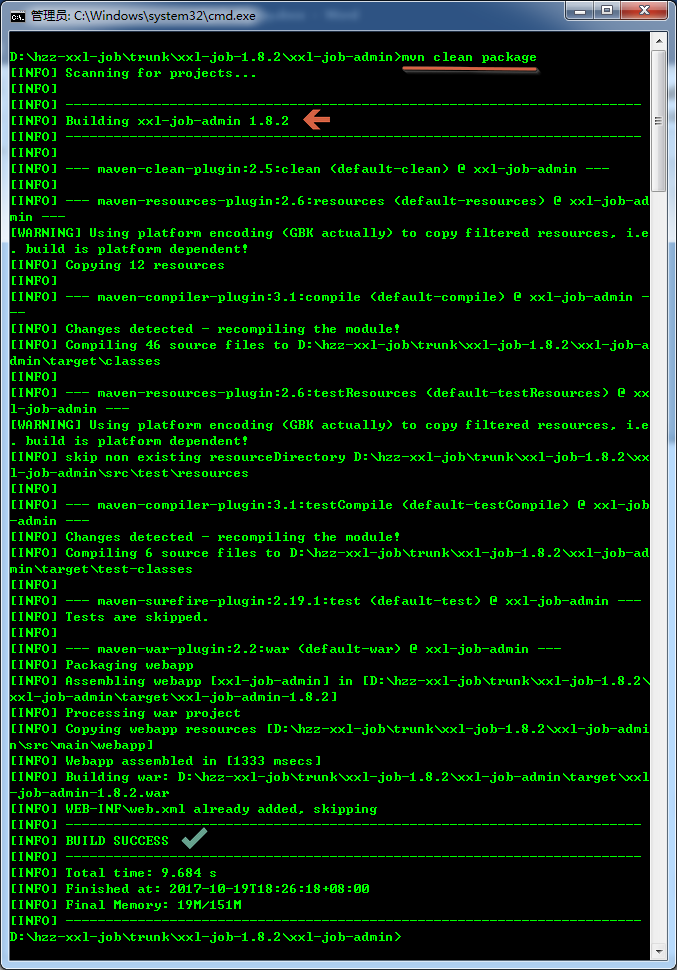
Cmd命令：cd xxl-job-admin（切换到xxl-job-admin工程文件夹下）



### maven命令

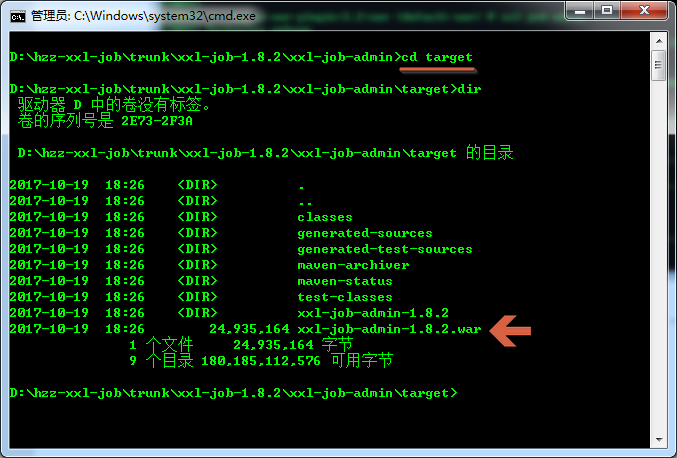
maven命令打包

Maven命令：mvn clean package（clean：清除target目录，package：打包，先清再打包）



BUILD SUCCESS：表示构建成功

切换到xxl-job-admin/target目录下就会发现“xxl-job-admin-1.8.2.war”已经打好。

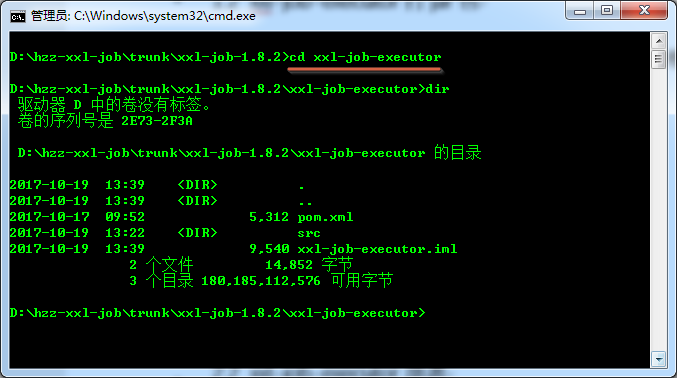


## xxl-job-executor打jar包

### cmd命令

切换到xxl-job-executor项目根目录

Cmd命令：cd xxl-job-executor（切换到xxl-job- executor工程文件夹下）

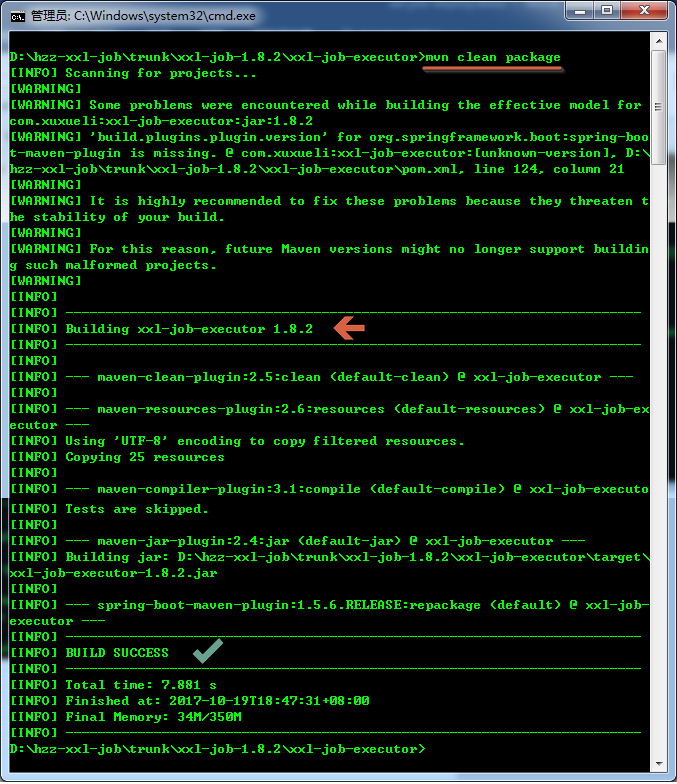


### maven命令

maven命令打包

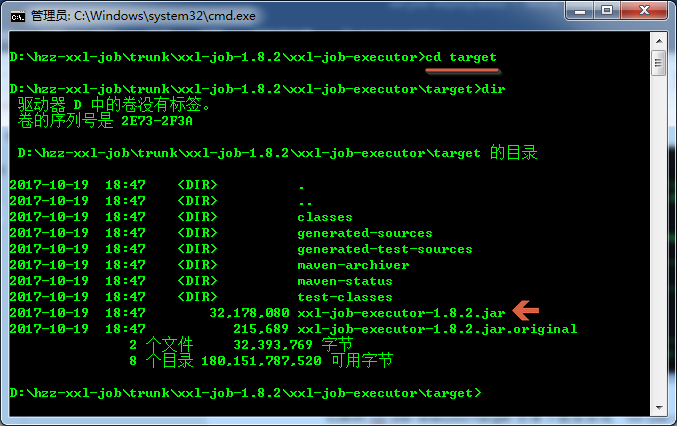
Maven命令：mvn clean package（clean：清除target目录，package：打包，先清再打包）

注：中间部分打印信息已截取掉

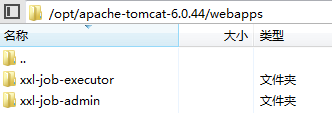


BUILD SUCCESS：表示构建成功

切换到xxl-job-executor/target目录下就会发现“xxl-job-executor-1.8.2.jar”已经打好。

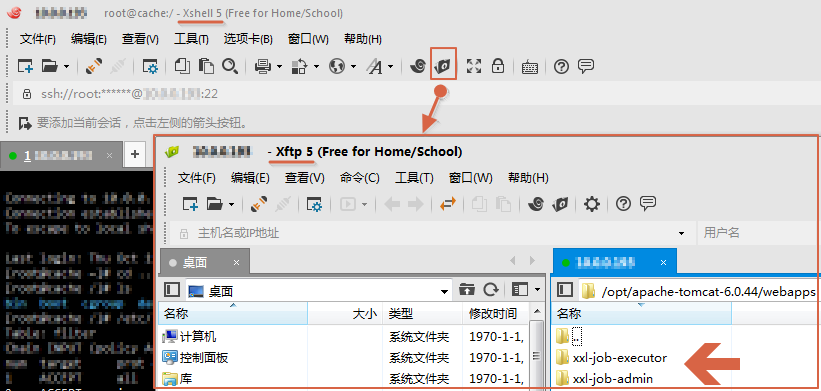


# Linux服务器部署



环境：

* CentOS release 6.8 (Final)
* JDK 1.7.0\_79
* Tomcat 6.0.44
* MySQL 5.6+
* XXL-JOB 1.8.2
* Spring Boot 1.5.6
* XShell 5（通过SSH客户端连接Linux）
* Xftp 5（XShell利用Xftp工具进行文件传输）
* 注：通过Xftp把本地工程上传至Linux服务器上



## xxl-job-admin部署

### 解压war包

#### 切换指定目录

首先，在“/opt/apache-tomcat-6.0.44/webapps”文件夹下，通过Xftp手动创建文件夹名：

xxl-job-admin（方便解压war包）

把“xxl-job-admin-1.8.2.war”上传到

“/opt/apache-tomcat-6.0.44/webapps/xxl-job-admin”Linux目录下；

用Linux命令切换到指定目录

Linux命令：cd opt/apache-tomcat-6.0.44/webapps/xxl-job-admin/



#### 解压war包

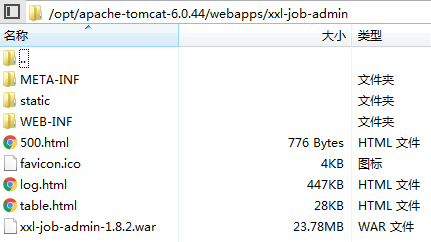
解压“xxl-job-admin-1.8.2.war”包

Linux命令：jar -xvf xxl-job-admin-1.8.2.war（解压到当前目录）



|  |  |
| --- | --- |
| tar -xvf file.tar | 解压\*.tar包 |
| tar -xzvf file.tar.gz | 解压\*.tar.gz包 |
| tar -xjvf file.tar.bz2 | 解压\*.tar.bz2包 |
| tar –xZvf file.tar.Z | 解压\*.tar.Z包 |
| unrar e file.rar | 解压\*.rar包 |
| unzip file.zip | 解压\*.zip包 |
| jar -xvf file.war | 解压\*.war包 |

解压出来目录结构为：



### 修改配置信息

#### 切换到“xxl-job-admin/WEB-INF/classes/spring”目录下

Linux命令：cd xxl-job-admin/WEB-INF/classes/spring/



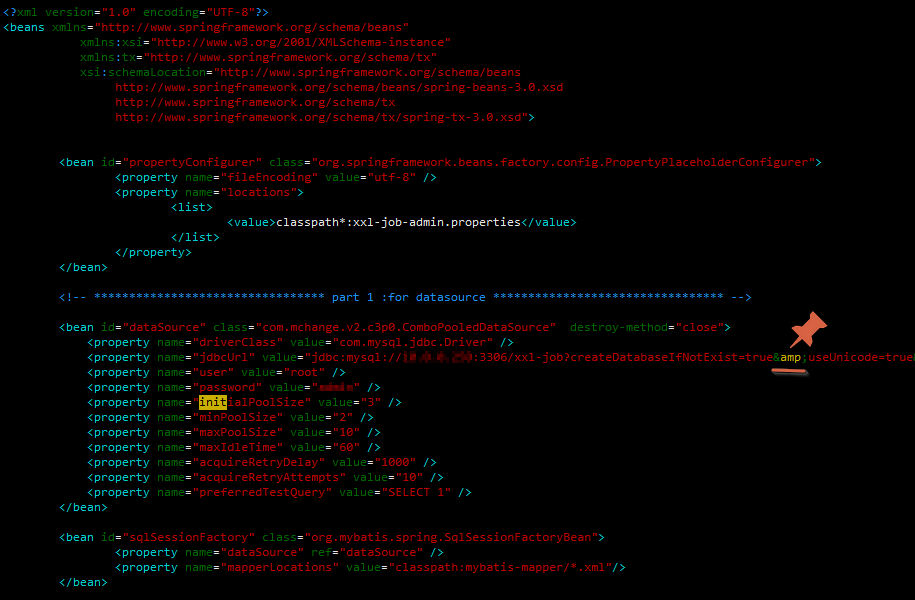
#### 编辑applicationcontext-xxl-job-admin.xml配置文件

Linux命令：vim applicationcontext-xxl-job-admin.xml

vim = 编辑模式

vim可以当作vi的升级版本，它可以用多种颜色的方式来显示一些特殊的信息。



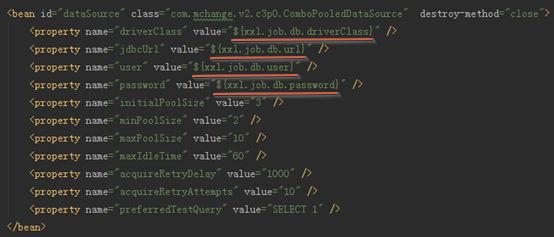


* 注意点：

#### 查看“applicationcontext-xxl-job-admin.xml”配置文件中的

* driverClass
* jdbcUrl
* user
* password
* accessToken

检查这5个property标签name的 value值是不是“${\*\*.\*\*}”，如下图：





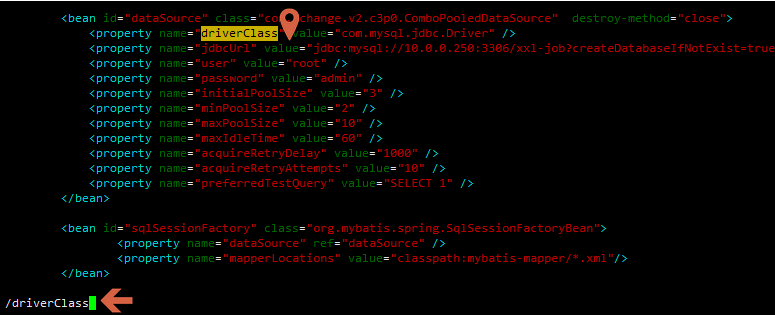
如果value的值是“${\*\*.\*\*}”，接下来就需要通过Linux命令修改：

##### 第一步

用“vim applicationcontext-xxl-job-admin.xml”命令打开文件并进入编辑模式

##### 第二步

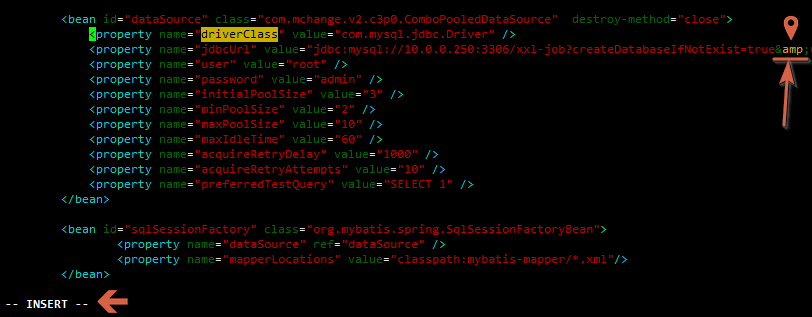
可以通过“/driverClass”命令进入关键字定位搜索



##### 第三步

按“i”命令进入编辑状态（可以更改内容）

按“ i ”之后，左下方就会出现“INSERT”或“插入”的字样，这样就表示已经进入编辑状态，可以更改内容了。



* + 注意点：jdbcUrl的value值有一个地方需要注意，就是

&amp;（applicationcontext-xxl-job-admin.xml）= &（xxl-job-admin.properties）

在xml文件中有以下几类字符要进行转义替换：

|  |  |  |
| --- | --- | --- |
| &lt; | < | 小于号 |
| &gt; | > | 大于号 |
| &amp; | & | 和 |
| &apos; | ' | 单引号 |
| &quot; | " | 双引号 |

不进行转义的话，Tomcat启动就会报异常：

tomcat The reference to entity "useUnicode" must end with the ';' delimiter

译文：（tomcat对实体“使用unicode”的引用必须以“;“分隔符）

##### 第四步

退出编辑状态，按“Esc”键或者按“Ctrl+C”组合键

（INSERT”或“插入”的字样消失就表示已退出编辑状态）

##### 第五步

保存退出，输入“ **:**wq ”（冒号+wq）命令，再按回车键

“ **:**wq ”保存并退出，返回到shell下

### 启动Tomcat

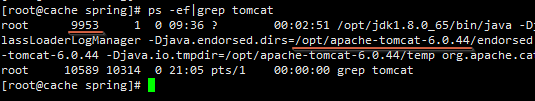
#### 第一步：检查Tomcat是否已启动

启动Tomcat容器之前，需要先检查一下Tomcat是否已经启动

Linux命令：ps -ef|grep tomcat（检查Tomcat进程号）



如果出现如下图所示，就表示Tomcat已启动过：



9953：Tomcat进程号

/opt/apache-tomcat-6.0.44：Tomcat路径位置

#### 第二步：杀掉Tomcat进程

Linux命令：kill -15 9953（Tomcat进程号）



kill -9和kill -15的区别？

SIGNKILL（9）的效果是立即杀死进程. 该信号不能被阻塞, 处理和忽略。

SIGNTERM（15）的效果是正常退出进程，退出前可以被阻塞或回调处理。

所以，推荐大家使用标准的kill进程方式，即 kill -15 pid

杀完Tomcat进程，再去查看Tomcat进程，就是这样子了，如下图：



（这样就表示Tomcat未启动）

#### 第三步：启动Tomcat

切换到Tomcat的bin目录下

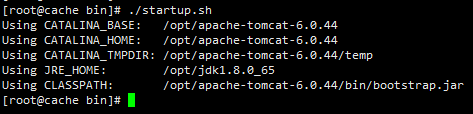
Linux命令：cd /opt/apache-tomcat-6.0.44/bin



然后再执行“./startup.sh”脚本

Linux命令：./startup.sh





#### 第四步：查看日志

切换到“logs”日志目录下

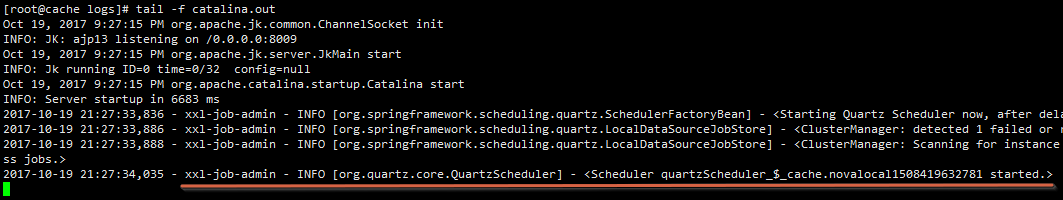
Linux命令：cd ../logs



查看日志内容

Linux命令：tail -f catalina.out



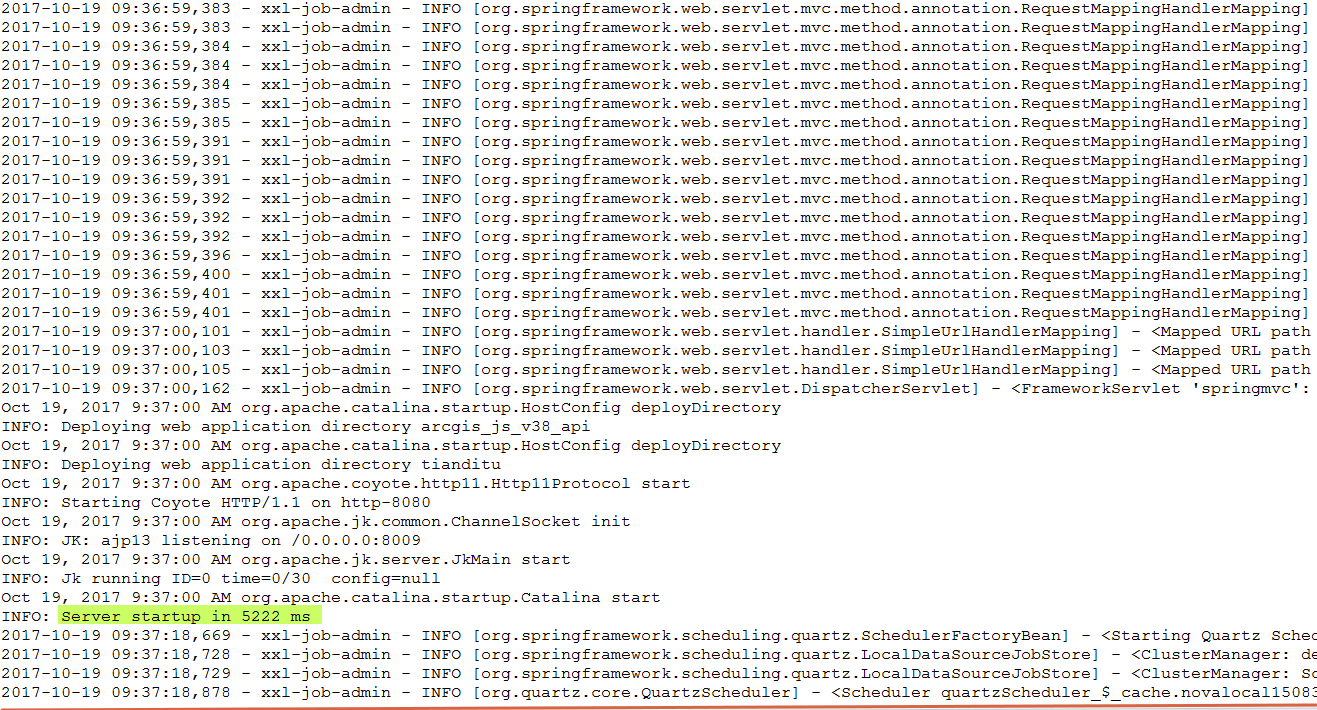


退出日志查看：Ctrl+C



#### 第五步：浏览器访问

如果运行时未报任何异常信息，则表示xxl-job-admin在Linux环境上已经部署成功。



浏览器URL访问：http://IP:8080/xxl-job-admin/toLogin



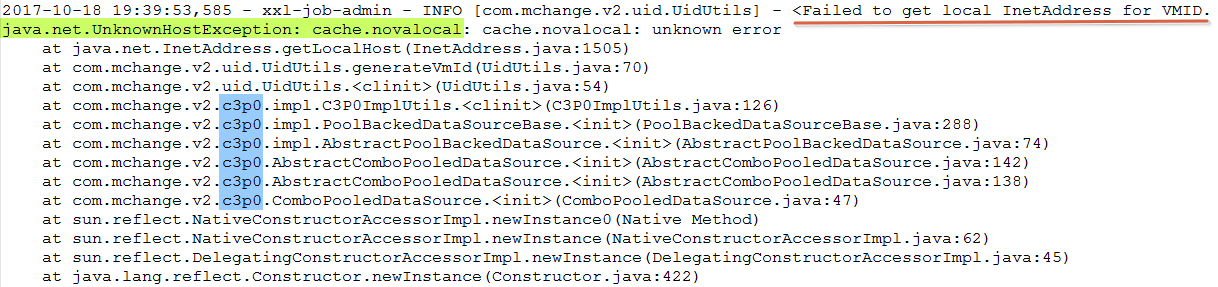
到此，xxl-job-admin工程已成功部署完毕！

### 运行异常

#### 异常信息

Failed to get local InetAddress for VMID

Java项目中用到了C3P0数据源，程序在Windows下运行没有任何问题，但是往服务器(Linux环境)上部署，运行的时候就会出现该异常，异常信息如下：



##### 异常原因

先看看“UnknownHostException: cache.novalocal”中cache.novalocal是怎么得来的吧，

Linux命令：cat /etc/sysconfig/network

内容如下：



HOSTNAME=cache.novalocal，是因为系统没有找到主机名cache.novalocal对应的IP，只需修改Linux的hosts文件即可。

##### 解决办法

* 方法一：修改Linux hosts文件（▲**推荐**）

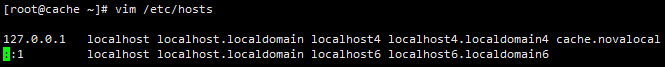
首先，执行 cat /etc/hosts 命令，如下：



然后再执行 vim /etc/hosts 命令

在第一行最后加上“空格+cache.novalocal”保存即可。

修改完之后的hosts文件如下：

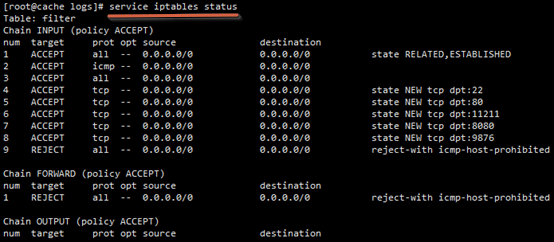


Linux查找域名时通常先查找文件/etc/hosts，找不到时再向DNS服务器请求。

* ~~方法二：关闭Linux防火墙~~

（**亲测过，防火墙关闭还是会报同样的异常信息，写出来就是让大家了解如何用Linux命令查看防火墙状态，以及开启或关闭**）

Linux命令：service iptables status （查看防火墙当前状态）



执行命令后，出现上图信息，就表示防火墙是开启状态。

* + Linux防火墙（iptables）重启系统生效

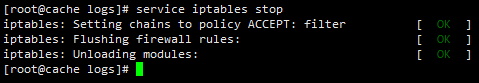
|  |  |
| --- | --- |
| chkconfig iptables on | 开启 |
| chkconfig iptables off | 关闭 |

* + Linux防火墙（iptables）即时生效，重启后失效（▲**推荐**）

|  |  |
| --- | --- |
| service iptables start | 开启 |
| service iptables stop | 关闭 |

关闭Linux防火墙

Linux命令：service iptables stop



关闭防火墙之后，再查看防火墙的状态，如下图：



iptables: Firewall is not running.（译文：iptables:防火墙没有运行。）

这样的话，就表示防火墙已经是关闭状态了。

开启Linux防火墙

Linux命令：service iptables start

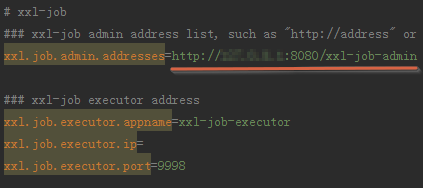


## xxl-job-executor部署

xxl-job- executor-1.8.2.jar包部署之前，

先确认application.properties配置文件中的xxl-job-admin的指向地址是否正确。





#### 切换指定目录

首先，在“/opt/apache-tomcat-6.0.44/webapps”文件夹下，通过Xftp手动创建文件夹名：

xxl-job- executor（方便管理）

把“xxl-job-executor-1.8.2.jar”上传到

“/opt/apache-tomcat-6.0.44/webapps/xxl-job- executor”Linux目录下；

用Linux命令切换到指定目录

Linux命令：cd opt/apache-tomcat-6.0.44/webapps/xxl-job- executor/

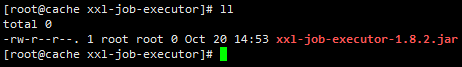


查看该目录

Linux命令：ls（列出文件和目录）



Linux命令：ll（ 列出文件、目录，包括隐藏文件的详细信息）



#### 运行jar包

##### 执行jar包运行命令

Linux命令：nohup java -jar xxl-job-executor-1.8.2.jar >xxl-job-executor.log &

* nohup：不挂断运行命令，当帐户退出或终端关闭时，程序仍然运行
* java -jar xxl-job-executor-1.8.2.jar：运行jar包命令
* >xxl-job-executor.log：把日志内容输出到指定的文件中，文件不存在则自动创建
* &：代表在后台运行



这时在用ls命令查看，xxl-job-executor.log文件就会被自动创建出来。



##### 查看jar包进程信息命令

（识别jar包是否已运行）

Linux命令：ps aux|grep xxl-job-executor-1.8.2.jar（查看jar包进程号）



如果有“java -jar xxl-job-executor-1.8.2.jar”这条信息，就表示jar包已运行

##### 关闭jar包运行

杀掉jar包进程号

Linux命令：kill -15 10113（jar包进程号）



杀完jar包进程号，再去查看jar包进程信息，就会出现下图内容：



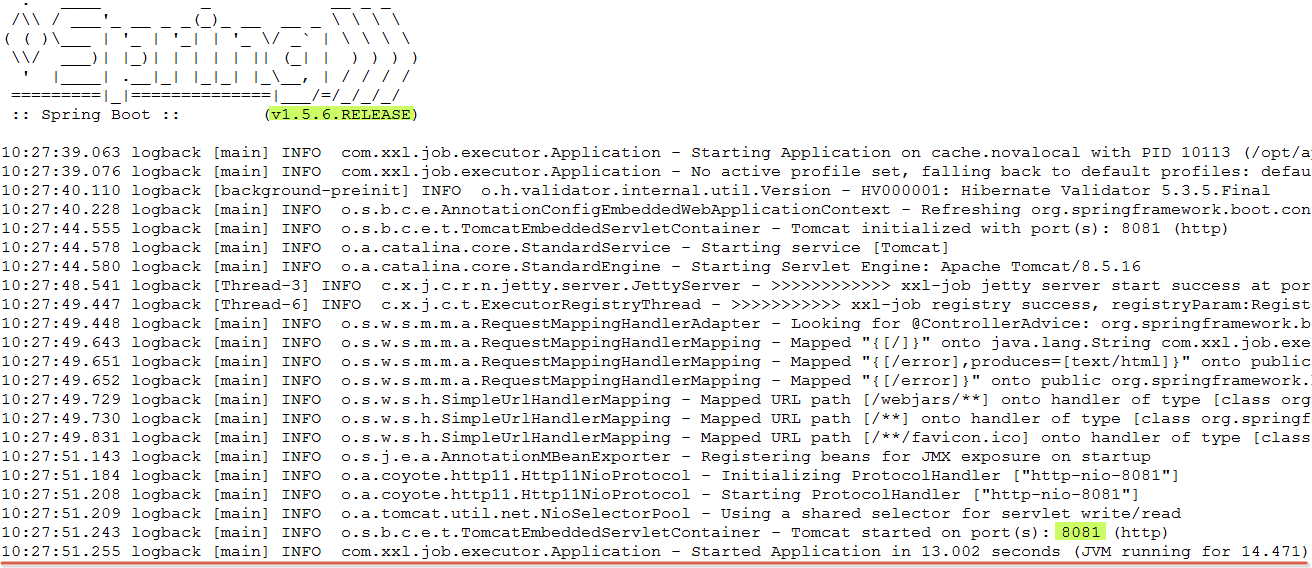
“java -jar xxl-job-executor-1.8.2.jar”这条信息就会没有了，这时就表示jar包未运行。

如何确定xxl-job-executor-1.8.2.jar包是否已运行成功，需要到xxl-job-executor.log文件查看日志内容，日志内容无异常信息，则表示jar包已运行成功，如下图所示：



注：为了方便内容展示，部分内容已删除。

无报错信息、有打印最后两行日志信息，则表示jar包在Linux环境上已运行成功！



在“任务调度中心”界面的执行器管理查看执行器是否已注册成功





“OnLine机器”这一列，有值就表示注册成功！