# HX\_ActiveMQ安装手册\_2012-10-30\_v1.doc

## 检查系统环境

查看操作系统版本

uname –a 或者

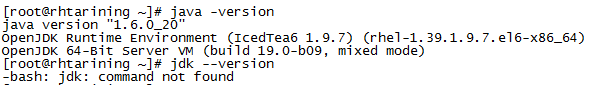


查看LINUX操作系统是多少位的



如果是64位的，下载相关软件的时候也下载64位的

查看是否安装jdk以及安装的jdk版本



如果系统已经安装openJDK，那么最好再下载一个sun的jdk,使用sun的jdk更稳定。

## 2.下载并安装linux平台下的jdk1.6版本

1) 下载地址：

http://www.oracle.com/technetwork/java/javase/downloads/jdk6u37-downloads-1859587.html

下载的安装包是jdk-6u37-linux-x64.bin，也可以下载以rpm.bin结尾的安装包，后面有注意事项解释两者的区别。

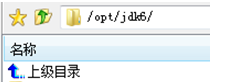
2) 通过文件上传工具falshfxp 或者ssh把下载好的jdk安装包上传到新建的/opt/jdk6目录下

1.新建目录命令：

mkdir /opt/jdk6

也可通过文件上传工具falshfxp手动创建文件夹，文件上传工具falshfxp可从网上搜索下载。

2.上传，上传成功后可以看到





3) 修改权限、执行安装

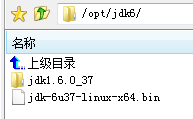
默认下载的安装包没有执行权限，通过下面的命令修改权限后，就可以执行安装了

切换目录：cd /opt/jdk6

修改文件权限： chmod 755 jdk-6u37-linux-x64.bin

执行： ./jdk-6u37-linux-x64.bin

安装后会显示出下图中的jdk1.6.0\_37文件夹。



1. 添加环境变量

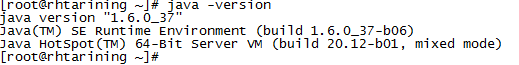
用文本编辑器打开/etc/profile，在profile文件末尾加入：  
#set java environment

export JAVA\_HOME=/opt/jdk6/jdk1.6.0\_37

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

export CLASSPATH=.:$JAVA\_HOME/lib/dt.jar:$JAVA\_HOME/lib/tools.jar  
修改完后，重新登录。

1. 测试是否安装成功



到此，64位的jdk安装成功。

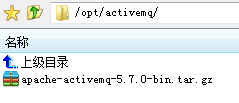
## 3.下载activemq并配置环境

1） 进入官网下载页面

<http://www.apache.org/dyn/closer.cgi?path=%2Factivemq%2Fapache-activemq%2F5.7.0%2Fapache-activemq-5.7.0-bin.tar.gz>

下载最新linux版本的Activemq

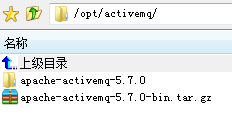
2) 通过文件上传工具falshfxp 或者ssh把下载好的activemq安装包上传到/opt/activemq目录下，如果没有该目录，则首先建立该目录，输入命令mkdir /opt/activemq



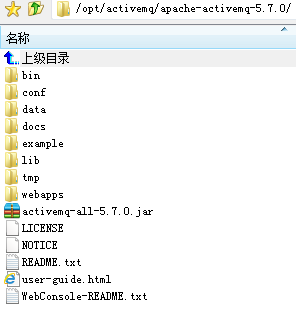
3） 进入activemq目录，解压缩安装包 tar –xvzf apache-…



解压缩后出现



目录结构如下：



4） 设置启动activemq的默认配置文件

如果是安装后的第一次启动，需要做以下设置。

./activemq setup /etc/default/activemq

截图：





5) 启动activemq

./activemq start





也可以使用命令

./activemq-admin start

这种方式是在前台启动的，关闭窗口，或者ctrl+c 就会结束activemq，后面第9点有介绍后台启动的方式。测试的时候直接使用./activemq start 就行了。

6) 查看61616端口是否正常启动

命令：netstat -an | grep 61616



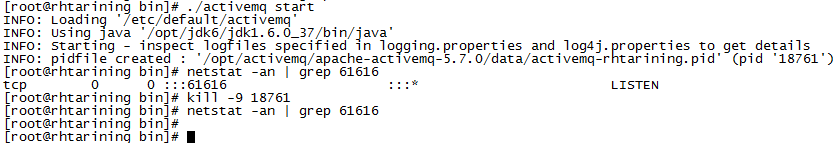
如果显示上面的信息，则进一步访问activemq的管理页面：<http://localhost:8161/admin/>，如果有页面显示，则说明启动成功了。注意标红处改成安装了activemq的主机ip地址。

7) 停止activemq

或者 杀死启动成功后生成的进程id，该进程id在每一次启动成功后都会生成一个。

图中的pid就是进程id.

命令： kill -9 18761

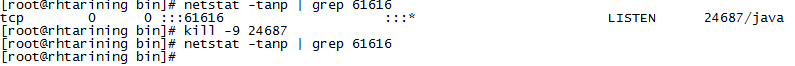


或者 也可以通过端口号查看进程id

输入命令： netstat -tanp | grep 61616

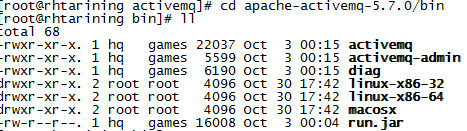
图中24687就是进程id,

只要使用命令 kill -9 24687 杀死进程,就关闭了activemq



8）查看activemq的状态

首先进入bin目录，然后输入命令: ./activemq status



C:\Users\Administrator\Documents\Fetion\temp\019edd9bbe999fba0edba66b58867055.png

如果命令结果最下面一行显示activemq not running ：没有启动成功

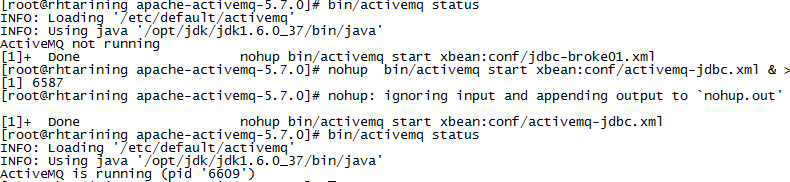


activemq is running :启动成功。

9） 以后台方式启动activemq,并指定日志文件

命令： nohup bin/activemq start xbean:conf/activemq-jdbc.xml & >/tmp/smlog

标红处根据自己写的 需要启动的activemq配置文件而定。



## 4.配置activemq的持久化

1. 配置文件可以参照安装文件的conf目录下的activemq-jdbc.xml来配置，最好拷贝conf下的配置文件，然后修改。如果使用oracle数据库持久化，则修改如下配置：

<!—使用oracle作为持久化数据库 - ->

<persistenceAdapter>

<jdbcPersistenceAdapter dataDirectory="${activemq.data}" dataSource="#oracle-ds" useDatabaseLock=”false”/>

</persistenceAdapter>

<!—配置oracle 数据源 –>

<bean id="oracle-ds" class="org.apache.commons.dbcp.BasicDataSource" destroy-method="close">

<property name="driverClassName" value="oracle.jdbc.driver.OracleDriver"/>

<!—此处的ip地址是安装oracle的主机ip地址，haf是数据库实例名-->

<property name="url" value="jdbc:oracle:thin:@10.248.60.224:1521:haf"/>

<!—此处的haf是访问数据库的用户名，passw0rd是访问数据库的密码-->

<property name="username" value="haf"/>

<property name="password" value="passw0rd"/>

<property name="maxActive" value="200"/>

<property name="poolPreparedStatements" value="true"/>

</bean>

1. 需要在安装文件的lib目录下，加入相应数据库（这里是oracle）的jar包。



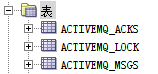
1. 如果在主机上10.248.60.224上还没有haf数据库实例,则创建一个名为haf的数据库，并设置访问此数据库的用户名为haf,密码为passw0rd。如果已经有的话，则上面的配置中标红的参数须与已经存在的数据库实例一致。
2. 重新启动activemq

首先切换到/opt/activemq/apache-activemq-5.7.0/bin下面，然后输入命令./activemq start





1. 使用数据库客户端，打开数据库实例，查看所有的表，如果发现系统自动创建了以下三张表，则表示持久化配置成功。



注意：activemq的配置文件的格式编写不能出错，出错了会导致不能启动成功。

## 5.Activemq集群配置

根据官方文档的介绍，activemq的master/slave集群有三种方式：[Pure Master Slave](http://activemq.apache.org/pure-master-slave.html) , [Shared File System Master Slave](http://activemq.apache.org/shared-file-system-master-slave.html) , [JDBC Master Slave](http://activemq.apache.org/jdbc-master-slave.html) 。

第一种需要手动重启宕掉的master,并且只支持一个slave。第二种是共享文件系统的主从集群，即使用文件系统进行持久化。第三种是使用jdbc，即数据库进行持久化。

下面使用的是第三种集群方式：

**1） 概述：**

1. 配置上，不存在Master和Slave，所有Broder的配置基本是一样的

2. 多个共享数据源的Broker构成JDBC Master Slave

3. 给每个Broker取个名字

4. 首先抢到资源（数据库锁）的Broker成为Masetr

5. 其他Broker保持预备状态，定期尝试抢占资源，运行其的Shell中清楚的显示了这一点

6. 一旦Master崩溃，其他Broker尝试抢占资源，最终只有一台抢到，它立刻成为Master

7. 之前的Master即使重启成功，也只能作为Slave等待

**2） MQ配置**（这里是一台机器配置）

1. 将 activemq安装文件下的/conf/activemq.xml 复制3份，分别改名为：activemq-broker01.xml、activemq-broker02.xml、activemq-broker03.xml

2. 首先修改jdbc\_broker01.xml

1) <broker brokerName=" activemq-broker01" …

2) <!--配置持久适配器-->

<persistenceAdapter>

<jdbcPersistenceAdapter dataDirectory="${activemq.base}/data"

dataSource="# oracle-ds "/>

</persistenceAdapter>

3) <transportConnector name="openwire" uri="tcp://0.0.0.0:61616"/>

4) <!--配置数据源：注意是在broker标记之外-->

··…

</broker>

<bean id="oracle-ds" class="org.apache.commons.dbcp.BasicDataSource" destroy-method="close">

<property name="driverClassName" value="oracle.jdbc.driver.OracleDriver"/>

<!—此处的ip地址是安装oracle的主机ip地址，haf是数据库实例名-->

<property name="url" value="jdbc:oracle:thin:@10.248.60.224:1521:haf"/>

<!—此处的haf是访问数据库的用户名，passw0rd是访问数据库的密码-->

<property name="username" value="haf"/>

<property name="password" value="passw0rd"/>

<property name="maxActive" value="200"/>

<property name="poolPreparedStatements" value="true"/>

</bean>

3. 同样修改 activemq-broker02.xml ，与 activemq-broker01.xml 不同之处

1) <broker brokerName=" activemq-broker02" …

2) <transportConnector name="openwire" uri="tcp://0.0.0.0:61617"/>

4. 同样修改 activemq-broker03.xml ，与 activemq-broker01.xml 不同之处

1) <broker brokerName=" activemq-broker03" …

2) <transportConnector name="openwire" uri="tcp://0.0.0.0:61618"/>

3） **JAVA测试：队列**

1. 生产者



2. 消费者





3. java代码测试：

新建一个maven 工程

1.自己建一个包，把Producer.java Consumer.java Listener.java 拷贝进去。 producer:生产者 consumer:消费者。



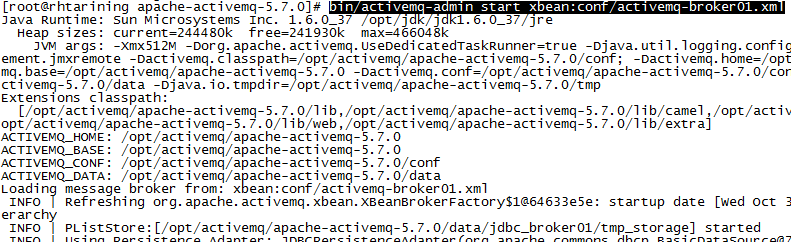
2.把log4j.properties拷到resource源码包里。 上面的三个文件是放在 java 源码包里的

3.在pom文件中加入相应jar包依赖。

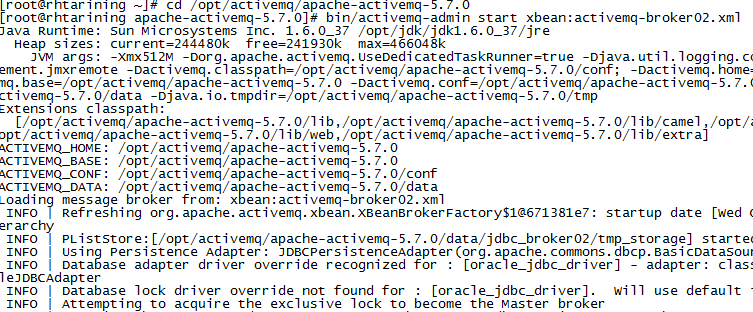
**4) 启动activemq**

做测试之前，activemq的配置要配正确，并且根据这三个配置文件启动activemq，也就是启动了三个不同的activemq

1.使用SecureCRT建立一个连接，切换目录，输入命令 bin/activemq-admin start xbean:conf/activemq-broker01.xml， 如下图



2. 使用SecureCRT再建立一个到linux的连接，切换目录，输入命令 bin/activemq-admin start xbean:conf/activemq-broker02.xml，如下图：



3. 使用SecureCRT再建立一个到linux的连接，切换目录，输入命令 bin/activemq-admin start xbean:conf/activemq-broker03.xml，如下图：



**5） 测试步骤**

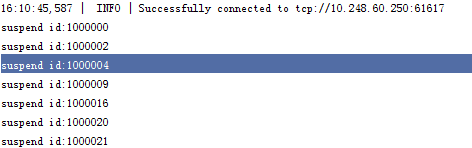
**1. 先启动生产者，发送几条消息**

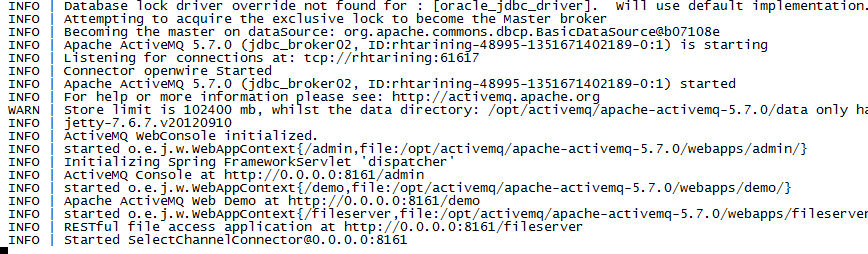
**2. 启动消费者，可看到接收到的消息**

**3. 关闭消费者**

**4. 生产者继续发送几条消息-消息A**

**5. 停止broker01（可看到生产者端显示连接到broker02（tcp://0.0.0.0:61617）了，同时运行broker02的Shell也显示其成为了Master） ，如下图：**



****

**6. 生产者继续发送几条消息-消息B**

**7. 启动消费者**

**8. 消费者接收了消息A和消息B，可见broker02接替了broker01的工作，而且储存了之前生产者经过broker01发送的消息**

**9. 关闭消费者**

**10.生产者继续发送几条消息-消息C**

**11.停止broker02（可看到生产者端显示连接到broker03（tcp://0.0.0.0:61618）了，同时运行broker03的Shell也显示其成为了Master）**

**12.生产者继续发送几条消息-消息D**

**13.启动消费者**

**14.消费者接收了消息C和消息D，可见broker03接替了broker02的工作，而且储存了之前生产者经过broker02发送的消息**

**15.再次启动broker01，生产者或消费者均未显示连接到broker01(tcp://0.0.0.0:61616)，表明broker01此时只是个Slave**

## 6.注意事项

1) 下载软件包时需注意：linux中 rpm.bin文件 与.bin文件的区别？

rpm.bin 你执行后会生成rpm打包的安装文件 你就可以使用命令 rpm -ivh xxx.rpm 来安装了

只能在redhat系的发行版上使用 (RHEL,FC,centOs)

.bin 这种的 ，你执行后就是jdk的安装文件了,适合所有的linux发行版。用bin文件和用windows的exe一样，rpm就像rar不管愿意与否，都解在同一个地方，好的是只用点确定。  
有些Linux软件不公开源代码，只发布二进制可执行程序，这类程序一般会以bin来标记；  
另一类软件是二进制安装程序和源代码一起发布，二进制程序文件以bin.rpm标记（以RedHat包为例），源代码以source.rpm标记。   
BIN是可以直接运行的二进制文件,RPM是红帽公司用的软件包管理器,一般来说,双击也可以直接安装,不过只限于支持RPM的发行版,如红帽等.  
UBUNTU就不可以,他支持DEB包   
jdk-1\_6\_0\_16-linux-i586-rpm.bin 运行后会解压出来一个rpm包，可用rpm -iUh命令安装。   
jdk-1\_6\_0\_16-linux-i586.bin 运行后会直接把所有文件解压到当前目录的jdk子目录下。

2）上面的命令截图中，

之后的字符是命令，其他的是命令的执行信息。 这里的root是当前用户，rhtarining是主机名，bin是当前的目录，查看当前目录可使用pwd命令。

3） 命令./activemq start 是以默认的配置文件启动的，即conf目录下的activemq.xml

命令./activemq start xbean:conf/activemq-jdbc.xml start 可以指定根据conf目录下的activemq-jdbc.xml来启动activemq.