版权声明：本文为博主原创文章，未经博主允许不得转载。

第一次遇到Tomcat在Linux服务器启动卡住的情况，情况很简单，tomcat启动以后卡在INFO: Deploying web application directory ......这句话，具体会卡多久就没测试了。google、baidu都没找到解决方法。

幸亏UCloud的技术支持人员给出了解决方案。

找到jdk1.x.x\_xx/jre/lib/security/[**Java**](http://lib.csdn.net/base/17).security文件，在文件中找到securerandom.source这个设置项，将其改为：

securerandom.source=file:/dev/./urandom

这时候根据修改内容就可以查到因为此原因不仅可以造成tomcat卡住，也会造成weblogic启动缓慢，

linux或者部分unix系统提供随机数设备是/dev/random 和/dev/urandom ，两个有区别，urandom安全性没有random高，但random需要时间间隔生成随机数。jdk默认调用random。  
  
  
再后来，终于在weblogic的官方文档中 Monitoring and Troubleshooting 找到了 Avoiding JVM Delays Caused By Random Number Generation 这样一个标题。摘录如下：  
  
The library used for random number generation in Sun's JVM relies on /dev/random by default for UNIX platforms. This can potentially block the Oracle WebLogic Communication Services process because on some operating systems /dev/random waits for a certain amount of "noise" to be generated on the host machine before returning a result. Although /dev/random is more secure, Oracle recommends using /dev/urandom if the default JVM configuration delays Oracle WebLogic Communication Services startup.  
  
To determine if your operating system exhibits this behavior, try displaying a portion of the file from a shell prompt:  
  
head -n 1 /dev/random  
Open the $JAVA\_HOME/jre/lib/security/java.security file in a text editor.  
  
Change the line:  
  
securerandom.source=file:/dev/random  
to read:  
  
securerandom.source=file:/dev/urandom  
Save your change and exit the text editor.  
其中说到：可通过 head -n 1 /devrandom 查看是否你的系统会出现伪随机数提供等待。OK就这个，试了一下，果然，在服务器第一次启动后，这个可以快速提供一个值，但当再次调用时发生等待。  
  
  
解决办法：  
  
  
永久：oracle 说修改 $JAVA\_HOME/jre/lib/security/java.security 文件，替换securerandom.source=file:/dev/random 为 securerandom.source=file:/dev/urandom。对所有使用JVM的应用生效。（这个永久的方法，这里面有个问题，就是设置时候实际应该设置为**securerandom.source=file:/dev/./urandom**，否则不生效）  
  
  
DOMAIN临时：修改startWeblogic.sh文件，JAVA\_OPTIONS="${SAVE\_JAVA\_OPTIONS} -Djava.security.egd=file:/dev/./urandom"  
  
  
后继的SecureRandom 测试学习  
  
  
编写JAVA类如下，运行测试，第一次正常，第二次等待，重启服务器后第一次又正常。启动加入参数 -Djava.security.egd=file:/dev/./urandom 正常