



链滴

CentOS7 下安装 Tomcat8 并且解决启动速度巨慢的问题

作者: [mzlion](#)

原文链接: <https://ld246.com/article/1479778148182>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

<h2 id="CentOS7下安装Tomcat8并且解决启动速度巨慢的问题">CentOS7 下安装 Tomcat8 并且决启动速度巨慢的问题</h2>

<h2 id="-0x0-介绍">[0x0]介绍</h2>

<p>Tomcat 的安装本身并没有什么复杂性，但是在启动的时候巨慢，运气不好卡个 3-5 分钟，所以篇文章主要是为了记录解决巨慢的问题。</p>

<h2 id="-0x1-环境介绍">[0x1]环境介绍</h2>

CentOS7.0_x86-64

JDK1.8

<h2 id="-0x2-下载">[0x2]下载</h2>

<p>Tomcat 的升级非常快，现在维持更新的有 7.x, 8.0.x, 8.5.x 和 9.x, 每个版本对 JDK 的要求不一样，需要自己根据自己当时 JDK 环境选择不同的 Tomcat 版本。</p>

<p>我的服务器目前选择的是 Tomcat.8.0.38, 要求 JDK7 以上，那么就下载吧，同样我已经下载好且上传到微云了，有需要的可以直接下载。</p>

<h2 id="-0x3-安装启动">[0x3]安装&启动</h2>

<p>安装：Tomcat 的安装其实就是解压到一个目录就表示安装完成了。下面就是解压的命令。</p>

```
<pre><code class="language-bash highlight-chroma"><span class="highlight-line"><span class="highlight-cl"><span class="highlight-o">[</span>web@ebs-29770 ~<span class="highlight-o">]</span></span>$ tar -zxf tomcat8.0.38-base.tar.gz</span></span></code></pre>
```

<p>启动：进入刚刚解压的 Tomcat 目录，然后在进入 bin 目录，执行如下命令：</p>

```
<pre><code class="language-bash highlight-chroma"><span class="highlight-line"><span class="highlight-cl"><span class="highlight-o">[</span>web@ebs-29770 bin<span class="highlight-o">]</span></span>$ ./startup.sh</span></span></code></pre>
```

<p>下面是启动日志</p>

```
<pre><code class="language-bash highlight-chroma"><span class="highlight-line"><span class="highlight-cl"><span class="highlight-o">[</span>web@ebs-29770 tomcat8.0.38<span class="highlight-o">]</span></span>$ ./bin/startup.sh</span></span><span class="highlight-line"><span class="highlight-cl">Using CATALINA_</span></span>ASE: /home/web/tomcat8.0.38</span></span><span class="highlight-line"><span class="highlight-cl">Using CATALINA_</span></span>OME: /home/web/tomcat8.0.38</span></span><span class="highlight-line"><span class="highlight-cl">Using CATALINA_</span></span>MPDIR: /home/web/tomcat8.0.38/temp</span></span><span class="highlight-line"><span class="highlight-cl">Using JRE_HOME:</span></span> apps/java/jdk/jdk1.8.0_112/jre</span></span><span class="highlight-line"><span class="highlight-cl">Using CLASSPATH:</span></span> /home/web/tomcat8.0.38/bin/bootstrap.jar:/home/web/tomcat8.0.38/bin/tomcat-juli.jar</span></span><span class="highlight-line"><span class="highlight-cl">Tomcat started.</span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-o">[</span>web@ebs-29770 tomcat8.0.38<span class="highlight-o">]</span></span>$ tail -f</span></span> ogs/catalina.out</span></span><span class="highlight-line"><span class="highlight-cl">14-Nov-2016 23:1</span></span>:37.559 INFO <span class="highlight-o">[</span>main<span class="highlight-o">]</span></span>rg.apache.catalina.startup.VersionLoggerListener.log Command line argumenttemp</span></span><span class="highlight-line"><span class="highlight-cl">14-Nov-2016 23:1</span></span>:37.560 INFO <span class="highlight-o">[</span>main<span class="highlight-o">]</span></span>rg.apache.catalina.core.AprLifecycleListener.lifecycleEvent The APR based optimal performanc</span></span> in production environments was not found on the java.library.path: /usr/java/packages/lib/a</span></span><span class="highlight-line"><span class="highlight-cl">14-Nov-2016 23:1</span></span>:37.781 INFO <span class="highlight-o">[</span>main<span class="highlight-o">]</span></span></pre>
```

```
rg.apache.coyote.AbstractProtocol.init Initializing ProtocolHandler <span class="highlight-o"
[</span> <span class="highlight-s2"> "http-nio-8080" </span> <span class="highlight-o">]</
pan>
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> 14-Nov-2016 23:1
:37.803 INFO <span class="highlight-o">[</span>main<span class="highlight-o">]</span>
rg.apache.tomcat.util.net.NioSelectorPool.getSharedSelector Using a shared
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> 14-Nov-2016 23:1
:37.809 INFO <span class="highlight-o">[</span>main<span class="highlight-o">]</span>
rg.apache.coyote.AbstractProtocol.init Initializing ProtocolHandler <span class="highlight-o"
[</span> <span class="highlight-s2"> "ajp-nio-8009" </span> <span class="highlight-o">]</s
an>
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> 14-Nov-2016 23:1
:37.811 INFO <span class="highlight-o">[</span>main<span class="highlight-o">]</span>
rg.apache.tomcat.util.net.NioSelectorPool.getSharedSelector Using a shared
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> 14-Nov-2016 23:1
:37.815 INFO <span class="highlight-o">[</span>main<span class="highlight-o">]</span>
rg.apache.catalina.startup.Catalina.load Initialization processed in <span class="highlight-m"
902</span> m
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> 14-Nov-2016 23:1
:37.866 INFO <span class="highlight-o">[</span>main<span class="highlight-o">]</span>
rg.apache.catalina.core.StandardService.startInternal Starting service Cat
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> 14-Nov-2016 23:1
:37.866 INFO <span class="highlight-o">[</span>main<span class="highlight-o">]</span>
rg.apache.catalina.core.StandardEngine.startInternal Starting Servlet Engi
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> 14-Nov-2016 23:1
:37.878 INFO <span class="highlight-o">[</span>localhost-startStop-1<span class="highligh
-o">]</span> org.apache.catalina.startup.HostConfig.deployDirectory Dep/tomcat8.0.38/web
pps/ROOT
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> 14-Nov-2016 23:2
:42.357 INFO <span class="highlight-o">[</span>localhost-startStop-1<span class="highligh
-o">]</span> org.apache.catalina.util.SessionIdGeneratorBase.createSecu <span class="highl
ght-k">for</span> session ID generation using <span class="highlight-o">[</span>SHA1P
NG<span class="highlight-o">]</span> took <span class="highlight-o">[</span>303,955<
pan class="highlight-o">]</span> milliseconds.
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> 14-Nov-2016 23:2
:42.389 INFO <span class="highlight-o">[</span>localhost-startStop-1<span class="highligh
-o">]</span> org.apache.catalina.startup.HostConfig.deployDirectory Dep/web/tomcat8.0.38
webapps/ROOT has finished in 304,510 ms
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> 14-Nov-2016 23:2
:42.393 INFO <span class="highlight-o">[</span>main<span class="highlight-o">]</span>
rg.apache.coyote.AbstractProtocol.start Starting ProtocolHandler <span class="highlight-o">
</span> <span class="highlight-s2"> "http-nio-8080" </span> <span class="highlight-o">]</
pan>
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> 14-Nov-2016 23:2
:42.400 INFO <span class="highlight-o">[</span>main<span class="highlight-o">]</span>
rg.apache.coyote.AbstractProtocol.start Starting ProtocolHandler <span class="highlight-o">
</span> <span class="highlight-s2"> "ajp-nio-8009" </span> <span class="highlight-o">]</s
an>
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> 14-Nov-2016 23:2
:42.402 INFO <span class="highlight-o">[</span>main<span class="highlight-o">]</span>
rg.apache.catalina.startup.Catalina.start Server startup in <span class="highlight-m">304587
/span> ms
</span> </span> </code> </pre>
<h2 id="-0x4-启动巨慢问题分析">[0x4]启动巨慢问题分析</h2>
```

上面的 Tomcat 是刚下载解压出来的默认配置(只保留 Tomcat 自带), 但根据上面的启动日志, 然花了 3 分多钟, 然后仔细观察日志, 主要是卡在初始化 Session。开始我以为云服务器的问题, 然我在本地虚拟机上同样操作, 问题重现, 并且更换了 Tomcat7, 问题仍然出现, 所以。通过搜索和析, Tomcat 的 SessionID 是通过 SHA1PRNG 算法计算得到的, SHA1 算法需要一个密钥, 这个密在 Tomcat 启动的时候随机生成一个, 生成是使用了 Linux 随机函数生成器 `/dev/random`。读取它相当于生成随机数字。搜索[/dev/random](https://ld246.com/forward?goto=https%3%2F%2Fzh.wikipedia.org%2Fwiki%2F%2Fdev%2Frandom), 大概知道是什么鬼了: `/dev/random` 会根据 **噪**产生随机数, 如果 **噪音**不够它就会阻塞。Linux 是通过 I/O, 键盘端、内存使用量、CPU 利用率等方式来收集 **噪音**的, 如果 **噪音**不够生成随机数的时候就会被 **阻塞**。

[0x5]解决问题

解决上述问题有两个方案, 一个使用伪随机函数生成器, 另外一个加大 `/dev/random` 的熵池, 下面分别介绍这两种方案。

A.使用伪随机函数生成器 `/dev/unrandom`

`/dev/urandom` 并不是真正的随机行为(其实一般不容易重复), 主要有两个地可以修改。

- 通过修改 Tomcat 启动文件 `-Djava.security.egd=file:/dev/urandom`
- 通过修改 JRE 中的 java.security 文件 `securerandom.source=file:/dev/urandom`

B.增大 `/dev/random` 的熵池 (推荐)

问题的原因是由于熵池不够大, 所以增大它是最彻底的方法。我们可以通过软件的方法实现, 下是软件的安装和配置流程。

- 安装熵服务

```
yum install rng-tools
```

- 启动熵服务

```
systemctl start rngd
```

- 如果你的 CPU 不支持 DRNG 特性或者像我一样使用虚拟机, 可以使用 `/dev/unrandom` 来模拟。

```
cp /usr/lib/systemd/system/rngd.service /etc/systemd/system
```

```
vim /etc/systemd/system/rngd.service
```

```
#以下是编辑内容
```

```
ExecStart=/sbin/rngd -f -r /dev/urandom
```

- 重新载入服务


```
<pre><code class="language-bash highlight-chroma"><span class="highlight-line"><span class="highlight-cl"> systemctl daemon-reload
</span></span><span class="highlight-line"><span class="highlight-cl"> systemctl restart
ngd
</span></span></code></pre>
```

<p>经过上面的修改，我们再观察 <code>/proc/sys/kernel/random/entropy_avail</code> 基本上在 3000 左右。这个时候重新启动 Tomcat，发现启动时间正常。</p>

<h2 id="-0x6-总结">[0x6]总结</h2>

<p>上面的 Tomcat 启动问题还是比较少见的，所以有时候不好找，所以专门做个笔记记录下。</p>