



链滴

FIRST.Solo 博客搭建日记

作者: [someone53869](#)

原文链接: <https://ld246.com/article/1571928981925>

来源网站: [链滴](#)

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

<p></p>

<h4 id="搭建Solo博客的过程-其重点不在于代码的修改上-而是网站运维端的一些细节-">搭建 Solo 博客的过程，其重点不在于代码的修改上，而是网站运维端的一些细节。</h4>

<p>下面对本次 Solo 的成功部署及升级 https 过程，进行一个简要的文字总结，对于后期网站的部署和升级也有一定借鉴意义。</p>

<hr>

<p>下面正文开始，先列一个部署的前提清单：</p>

<table>

<thead>

<tr>

<th>名称</th>

<th>详情</th>

<th>备注</th>

</tr>

</thead>

<tbody>

<tr>

<td>云服务器</td>

<td>若干台</td>

<td>可单机、可集群</td>

</tr>

<tr>

<td>域名</td>

<td>1 个</td>

<td>配置 dns 解析</td>

</tr>

<tr>

<td>SSL 证书</td>

<td>绑定域名</td>

<td>升级 https 用</td>

</tr>

<tr>

<td>Solo 源代码</td>

<td>Solo</td>

<td>git 到本地后进行调整并用 maven 打包</td>

</tr>

<tr>

<td>系统环境</td>

<td>Linux+Java+MySQL+Tomcat+Nginx</td>

<td>windows 系统也可行</td>

</tr>

</tbody>

</table>

<p>从清单可以大致看出，本次我们想要部署方法的是非官方推荐的 Docker 部署，而是使用 Tomcat 的一种方式，同时使用 Nginx 进行了均衡负载实现，并加强了网站安全性，使用 https 对网站进行加密。</p>

<hr>

<p>前三个东西需要预先从各种平台购买、获取和配置（包括 DNS 解析及备案），这里不再详细描述，从 Solo 项目开始。</p>

<ol start="0">

Linux 基本运行环境安装和数据库建库

<p>在 Linux 环境下先后需要安装 Java、MySQL、Tomcat、Nginx，这里我先进行基本安装，默认大伙都知道基本的使用，基础配置不再描述，关键配置后面再讲；</p>

<p>Solo 的运行可以自行指定数据库类型，其中官方文档提供了两种方式 H2 和 MySQL，这里我们择 MySQL；</p>

<p>登录 MySQL 并执行 sql 语句（数据库名称自定）：</p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">CREATE DATABASE database_name DEFAULT CHARACTER SET UTF8MB4 COLLATE UTF8B4_UNICODE_CI;</span></span></code></pre>
```


代码配置修改

<p></p>

<p>从 Git 到本地并向成功导入 IDE 的代码中，我们修改了蓝色的几个文件：</p>

<p>在 latke 中修改了请求协议以及运行环境（其中请求方式改为 https 会影响网站部署的静态资源请求地址，https 下是不允许网站混合请求 http 的），如下：</p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">##### Server #####</span></span></pre>
```

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl"># Browser visit protocol</span></span></pre>
```

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">serverScheme=https #如果不做https升级可以默认http</span></span></pre>
```

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl"># Browser visit domain name</span></span></pre>
```

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">serverHost=localhost</span></span></pre>
```

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl"></span></span></pre>
```

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">##### Runtime Mode #####</span></span></pre>
```

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">### runtimeMode=DEVELOPMENT</span></span></pre>
```

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">runtimeMode=PRODUCTION</span></span></pre>
```

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl"></span></span></pre>
```

<p>在 local 中我们修改了链接数据库的相关参数（根据用户名、密码参数自行配置）；</p>

<ol start="2">

小坑.....皮肤在另外一个项目中

<p>这里是一个坑，卤煮在第一次部署时发现，tomcat 日志竟然报错，显示找不到皮肤，一查原因原始项目中已经不包含皮肤，但是却被默认指向一个 Pinghsu 的皮肤。经过查找，原来皮肤项目在里：Solo-Skins</p>

<p>git 过来后，将里面内容 Copy 到 webapp/skins 下面，目录存放如下图：</p>

<p></p>

<ol start="3">

打 war 包

<p>带上 dev 配置，此配置使用了 Aliyun 的仓库镜像，比较快，推荐~</p>

<p></p>

<p>打包成功! </p>

<p></p>

<ol start="4">

war 包上传

<p>将 war 包上传到 Tomcat 目录下, 启动 Tomcat, war 包会自动部署; </p>

<p>观察日志, Solo 会自动初始化数据库, 并提示成功: </p>

<blockquote>

<p>[INFO]-[2019-09-14 16:13:05]-[org.b3log.solo.SoloServletListener:99]: Solo is booting [ve=3.6.4, servletContainer=Apache Tomcat/xxx, os=Linux, isDocker=false, markdownHttpAvailale=false, pid=26689, runtimeDatabase=MYSQL, runtimeMode=PRODUCTION, jdbc.username xxx, jdbc.URL=jdbc:]

[INFO]-[2019-09-14 16:13:05]-[com.zaxxer.hikari.HikariDataSource:110]: HikariPool-2 - Starti g...

[INFO]-[2019-09-14 16:13:06]-[com.zaxxer.hikari.HikariDataSource:123]: HikariPool-2 - Start c mpleted.

[INFO]-[2019-09-14 16:13:06]-[org.b3log.solo.SoloServletListener:138]: Solo is running</p>

</blockquote>

<p>至此, 通过域名到 Tomcat 的指定端口应该已经可以正常访问! </p>

<hr>

<ol start="5">

高级部分-Nginx 负载均衡

<p>以上基础步骤完成, 现在想要进行高级点的试验——负载均衡, 这个时候就是 Nginx 派上用场时候了; </p>

<p>在 Linux 下, 修改 nginx.conf 文件: </p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">upstream tomcatServer { #负载均衡到不同的服务器上</span></span><span class="highlight-line"><span class="highlight-cl">    ip_hash;</span></span><span class="highlight-line"><span class="highlight-cl">    server ip1:port1</span></span><span class="highlight-line"><span class="highlight-cl">    server ip2:port2</span></span><span class="highlight-line"><span class="highlight-cl">    server ip3:port3</span></span><span class="highlight-line"><span class="highlight-cl">    server ip4:port4</span></span><span class="highlight-line"><span class="highlight-cl">    server ip5:port5</span></span><span class="highlight-line"><span class="highlight-cl">    server ip6:port6</span></span><span class="highlight-line"><span class="highlight-cl">    server ip7:port7</span></span><span class="highlight-line"><span class="highlight-cl">    server ip8:port8</span></span><span class="highlight-line"><span class="highlight-cl">    server ip9:port9</span></span><span class="highlight-line"><span class="highlight-cl">    server ip10:port10</span></span><span class="highlight-line"><span class="highlight-cl">    server ip11:port11</span></span><span class="highlight-line"><span class="highlight-cl">    server ip12:port12</span></span><span class="highlight-line"><span class="highlight-cl">    server ip13:port13</span></span><span class="highlight-line"><span class="highlight-cl">    server ip14:port14</span></span><span class="highlight-line"><span class="highlight-cl">    server ip15:port15</span></span><span class="highlight-line"><span class="highlight-cl">    server ip16:port16</span></span><span class="highlight-line"><span class="highlight-cl">    server ip17:port17</span></span><span class="highlight-line"><span class="highlight-cl">    server ip18:port18</span></span><span class="highlight-line"><span class="highlight-cl">    server ip19:port19</span></span><span class="highlight-line"><span class="highlight-cl">    server ip20:port20</span></span><span class="highlight-line"><span class="highlight-cl">    server ip21:port21</span></span><span class="highlight-line"><span class="highlight-cl">    server ip22:port22</span></span><span class="highlight-line"><span class="highlight-cl">    server ip23:port23</span></span><span class="highlight-line"><span class="highlight-cl">    server ip24:port24</span></span><span class="highlight-line"><span class="highlight-cl">    server ip25:port25</span></span><span class="highlight-line"><span class="highlight-cl">    server ip26:port26</span></span><span class="highlight-line"><span class="highlight-cl">    server ip27:port27</span></span><span class="highlight-line"><span class="highlight-cl">    server ip28:port28</span></span><span class="highlight-line"><span class="highlight-cl">    server ip29:port29</span></span><span class="highlight-line"><span class="highlight-cl">    server ip30:port30</span></span><span class="highlight-line"><span class="highlight-cl">    server ip31:port31</span></span><span class="highlight-line"><span class="highlight-cl">    server ip32:port32</span></span><span class="highlight-line"><span class="highlight-cl">    server ip33:port33</span></span><span class="highlight-line"><span class="highlight-cl">    server ip34:port34</span></span><span class="highlight-line"><span class="highlight-cl">    server ip35:port35</span></span><span class="highlight-line"><span class="highlight-cl">    server ip36:port36</span></span><span class="highlight-line"><span class="highlight-cl">    server ip37:port37</span></span><span class="highlight-line"><span class="highlight-cl">    server ip38:port38</span></span><span class="highlight-line"><span class="highlight-cl">    server ip39:port39</span></span><span class="highlight-line"><span class="highlight-cl">    server ip40:port40</span></span><span class="highlight-line"><span class="highlight-cl">    server ip41:port41</span></span><span class="highlight-line"><span class="highlight-cl">    server ip42:port42</span></span><span class="highlight-line"><span class="highlight-cl">    server ip43:port43</span></span><span class="highlight-line"><span class="highlight-cl">    server ip44:port44</span></span><span class="highlight-line"><span class="highlight-cl">    server ip45:port45</span></span><span class="highlight-line"><span class="highlight-cl">    server ip46:port46</span></span><span class="highlight-line"><span class="highlight-cl">    server ip47:port47</span></span><span class="highlight-line"><span class="highlight-cl">    server ip48:port48</span></span><span class="highlight-line"><span class="highlight-cl">    server ip49:port49</span></span><span class="highlight-line"><span class="highlight-cl">    server ip50:port50</span></span><span class="highlight-line"><span class="highlight-cl">    server ip51:port51</span></span><span class="highlight-line"><span class="highlight-cl">    server ip52:port52</span></span><span class="highlight-line"><span class="highlight-cl">    server ip53:port53</span></span><span class="highlight-line"><span class="highlight-cl">    server ip54:port54</span></span><span class="highlight-line"><span class="highlight-cl">    server ip55:port55</span></span><span class="highlight-line"><span class="highlight-cl">    server ip56:port56</span></span><span class="highlight-line"><span class="highlight-cl">    server ip57:port57</span></span><span class="highlight-line"><span class="highlight-cl">    server ip58:port58</span></span><span class="highlight-line"><span class="highlight-cl">    server ip59:port59</span></span><span class="highlight-line"><span class="highlight-cl">    server ip60:port60</span></span><span class="highlight-line"><span class="highlight-cl">    server ip61:port61</span></span><span class="highlight-line"><span class="highlight-cl">    server ip62:port62</span></span><span class="highlight-line"><span class="highlight-cl">    server ip63:port63</span></span><span class="highlight-line"><span class="highlight-cl">    server ip64:port64</span></span><span class="highlight-line"><span class="highlight-cl">    server ip65:port65</span></span><span class="highlight-line"><span class="highlight-cl">    server ip66:port66</span></span><span class="highlight-line"><span class="highlight-cl">    server ip67:port67</span></span><span class="highlight-line"><span class="highlight-cl">    server ip68:port68</span></span><span class="highlight-line"><span class="highlight-cl">    server ip69:port69</span></span><span class="highlight-line"><span class="highlight-cl">    server ip70:port70</span></span><span class="highlight-line"><span class="highlight-cl">    server ip71:port71</span></span><span class="highlight-line"><span class="highlight-cl">    server ip72:port72</span></span><span class="highlight-line"><span class="highlight-cl">    server ip73:port73</span></span><span class="highlight-line"><span class="highlight-cl">    server ip74:port74</span></span><span class="highlight-line"><span class="highlight-cl">    server ip75:port75</span></span><span class="highlight-line"><span class="highlight-cl">    server ip76:port76</span></span><span class="highlight-line"><span class="highlight-cl">    server ip77:port77</span></span><span class="highlight-line"><span class="highlight-cl">    server ip78:port78</span></span><span class="highlight-line"><span class="highlight-cl">    server ip79:port79</span></span><span class="highlight-line"><span class="highlight-cl">    server ip80:port80</span></span><span class="highlight-line"><span class="highlight-cl">    server ip81:port81</span></span><span class="highlight-line"><span class="highlight-cl">    server ip82:port82</span></span><span class="highlight-line"><span class="highlight-cl">    server ip83:port83</span></span><span class="highlight-line"><span class="highlight-cl">    server ip84:port84</span></span><span class="highlight-line"><span class="highlight-cl">    server ip85:port85</span></span><span class="highlight-line"><span class="highlight-cl"&
```

```

</span></span><span class="highlight-line"><span class="highlight-cl"> proxy_set_header
Host $http_host;
</span></span><span class="highlight-line"><span class="highlight-cl"> proxy_set_header
X-Real-IP $remote_addr;
</span></span><span class="highlight-line"><span class="highlight-cl"> proxy_set_header
X-Forwarded-For $proxy_add_x_forwarded_for;
</span></span><span class="highlight-line"><span class="highlight-cl"> }
</span></span><span class="highlight-line"><span class="highlight-cl">>
</span></span></code></pre>
<p>然后进行 nginx Reload: </p>
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">nginx -s reload
</span></span></code></pre>
<ol start="6">
<li><strong>高级部分-网站升级 HTTPS</strong></li>
</ol>
<p>此部分可谓是艰辛，网站升级 https 是一件非常麻烦的事，各种坑都有可能遇见，本文列出详细置，以供参考；</p>
<p>首先下载 nginx 证书，放到特定目录下，并对 nginx 进行 443 端口的 https 配置：</p>
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl"># 以下属性中以ssl开头的属性代表与证书配置有关，其他属性请根据自己的需要进行配置
</span></span><span class="highlight-line"><span class="highlight-cl">server {
</span></span><span class="highlight-line"><span class="highlight-cl">listen 443;
</span></span><span class="highlight-line"><span class="highlight-cl">server_name loca
host; # localhost修改为您证书绑定的域名
</span></span><span class="highlight-line"><span class="highlight-cl">ssl on; #设置为o
启用SSL功能。
</span></span><span class="highlight-line"><span class="highlight-cl">root html;
</span></span><span class="highlight-line"><span class="highlight-cl">index index.html
ndex.htm;
</span></span><span class="highlight-line"><span class="highlight-cl">ssl_certificate /xx
/domain.pem; #将domain.pem替换成您证书的文件名，xxx为路径
</span></span><span class="highlight-line"><span class="highlight-cl">ssl_certificate_key
/xxx/domain.key; #将domain.key替换成您证书的密钥文件名，xxx为路径
</span></span><span class="highlight-line"><span class="highlight-cl">ssl_session_timeo
t 5m;
</span></span><span class="highlight-line"><span class="highlight-cl">ssl_ciphers ECDH
-RSA-AES128-GCM-SHA256:ECDHE:ECDH:AES:HIGH:!NULL:!aNULL:!MD5:!ADH:!RC4; #使用此
密套件
</span></span><span class="highlight-line"><span class="highlight-cl">ssl_protocols TLS
1 TLSv1.1 TLSv1.2; #使用该协议进行配置
</span></span><span class="highlight-line"><span class="highlight-cl">ssl_prefer_server_
iphers on;
</span></span><span class="highlight-line"><span class="highlight-cl">location / {
</span></span><span class="highlight-line"><span class="highlight-cl">#root html; #
点目录
</span></span><span class="highlight-line"><span class="highlight-cl">#index index.htm
index.htm;
</span></span><span class="highlight-line"><span class="highlight-cl">proxy_pass https:
/tomcatServer;
</span></span><span class="highlight-line"><span class="highlight-cl">}
</span></span><span class="highlight-line"><span class="highlight-cl">location ~.* {
</span></span><span class="highlight-line"><span class="highlight-cl">proxy_pass https:
/tomcatServer;

```

```

</span></span><span class="highlight-line"><span class="highlight-cl"> proxy_set_header
Host $http_host;
</span></span><span class="highlight-line"><span class="highlight-cl"> proxy_set_header
X-Real-IP $remote_addr;
</span></span><span class="highlight-line"><span class="highlight-cl"> proxy_set_header
X-Forwarded-For $proxy_add_x_forwarded_for;
</span></span><span class="highlight-line"><span class="highlight-cl"> proxy_set_header
X-Forwarded-Proto https;
</span></span><span class="highlight-line"><span class="highlight-cl"> }
</span></span><span class="highlight-line"><span class="highlight-cl"> }
</span></span></code></pre>
<p>修改原 80 端口的配置，强制跳转 https: </p>
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">server {
</span></span><span class="highlight-line"><span class="highlight-cl"> listen 80;
</span></span><span class="highlight-line"><span class="highlight-cl"> server_name loca
host;
</span></span><span class="highlight-line"><span class="highlight-cl"> rewrite ^(.*)$ htt
s://$host$1 permanent; #强行跳转
</span></span><span class="highlight-line"><span class="highlight-cl"> location / {
</span></span><span class="highlight-line"><span class="highlight-cl"> #index index.htm
index.htm;
</span></span><span class="highlight-line"><span class="highlight-cl"> proxy_pass https:
/tomcatServer;
</span></span><span class="highlight-line"><span class="highlight-cl"> }
</span></span><span class="highlight-line"><span class="highlight-cl"> location ~.* {
</span></span><span class="highlight-line"><span class="highlight-cl"> proxy_pass https:
/tomcatServer;
</span></span><span class="highlight-line"><span class="highlight-cl"> proxy_set_header
Host $http_host;
</span></span><span class="highlight-line"><span class="highlight-cl"> proxy_set_header
X-Real-IP $remote_addr;
</span></span><span class="highlight-line"><span class="highlight-cl"> proxy_set_header
X-Forwarded-For $proxy_add_x_forwarded_for;
</span></span><span class="highlight-line"><span class="highlight-cl"> }
</span></span><span class="highlight-line"><span class="highlight-cl"> }
</span></span></code></pre>
<p>然后 reload nginx; </p>
<p>此时，直接访问域名应该已经可以看到 nginx 界面，并且提示网站证书是安全的; </p>
<p>接下来，进行 Tomcat 容器的 https 的配置，这里以 <strong>Tomcat8.0 及以上版本</stron
>为例，其他版本可能有差别: </p>
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">&lt;!-- xxx.pfx是你的tomcat所用ssl证书的位置及文件名，password是证书密码，type为加密
式，默认RSA --&gt;
</span></span><span class="highlight-line"><span class="highlight-cl">&lt;Connector por
="8443" protocol="org.apache.coyote.http11.Http11NioProtocol" maxThreads="150" SSLEna
led="true"&gt;
</span></span><span class="highlight-line"><span class="highlight-cl">      &lt;SSLHost
onfig&gt;
</span></span><span class="highlight-line"><span class="highlight-cl">      &lt;Certific
te certificateKeystoreFile="xxx.pfx" type="RSA" certificateKeystorePassword="xxx" /&gt;
</span></span><span class="highlight-line"><span class="highlight-cl">      &lt;/SSLHost
onfig&gt;
</span></span><span class="highlight-line"><span class="highlight-cl">&lt;/Connector&g

```

```
;
```

```
</span></span></code></pre>
```

<p>这里我们保留了 Tomcat 的 8080 端口和 8443 端口</p>

<p>重启 Tomcat.....</p>

<p>好了，大功告成，如果一切顺利的话，网站已经实现了负载均衡和 https! </p>

<p>介于细节较多，文章只列举了核心的内容，其中滋味体验了才能知道，快来试试吧~不得不说 Sol 的界面和项目设计都是非常赏心悦目的，这里给点个大大的赞! :+1: :+1: :+1: :+1: :+1:</p>

<p>转载请注明出处，如有疑问欢迎留言~</p>