



黑客派

# 【笔记】如何使用宝塔面板 + Tomcat 搭建 Solo 博客。

作者: [honeymoe](#)

原文链接: <https://hacpai.com/article/1560170337771>

来源网站: [黑客派](#)

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

<p>概要：使用宝塔面板 +Tomcat 安装 Solo 博客的一次成功尝试的笔记。</p>

### 

<p>尽管不生产有价值的内容，甚至连写日记都难以坚持，但拥有一个自己的博客总是有趣的。<br>于是我购买域名，租用虚拟服务器，安装博客程序，终于能够从域名访问到自己的网站啦。<br>从开始尝试网络上普遍推荐的 Inmp+Wordpress，以至于 OpenLiteSpeed、Typecho 等等并不那么流的方案。对 Python 感兴趣的时候考虑过 Django，试图涉猎 Go 语言的时候看到 Hugo，而这次试图使用 GitHub 上 Star 数名列前茅的 Solo 来重建自己空荡荡的博客 w。（项目地址：<a href="https://link.hacpai.com/forward?goto=https%3A%2F%2Fgithub.com%2Fb3log" target="\_blank" rel="nofollow ugc">b3log</a>/<strong><a href="https://link.hacpai.com/forward?goto=https%3A%2F%2Fgithub.com%2Fb3log%2Fsolo" target="\_blank" rel="nofollow ugc">solo</a></trong>）</p>

<p>正如知乎 dboy 君所指出的：<a href="https://link.hacpai.com/forward?goto=https%3A%2F%2Fwww.zhihu.com%2Fquestion%2F29771102%2Fanswer%2F212218184" target="\_blank" rel="nofollow ugc">个人博客选择 wordpress 还是 typecho 好呢？</a></p>

<blockquote>

<p>你们这些孩子哪里是真为了写博客。<br>你们只是想体会换新工具的愉快感觉，顺便体会一下定掉旧工具，觉得自己特别的小欣喜。</p>

</blockquote>

<p>目前 Solo 官方推荐的安装方式为 Docker 安装，比较省事。（然而在使用宝塔面板时，我仅能行 Docker 容器，却难以解决域名管理和端口的问题，向各位赐教。）<br>因执着于宝塔面板所提的可视化（Windows 积习），小白经过一些踩坑尝试，遂有此一篇笔记。</p>

### 

<p>服务器：购买或租用一个虚拟服务器，使用 SSH 登录。<br>域名：购买域名，设置 DNS 指向服务器 IP。<br>宝塔面板：参考<a href="https://link.hacpai.com/forward?goto=https%3A%2F%2Fwww.bt.cn%2F" target="\_blank" rel="nofollow ugc">宝塔面板</a>官方网站的<a href="https://link.hacpai.com/forward?goto=https%3A%2F%2Fwww.bt.cn%2Fbbs%2Fthread-19376-1-1.html" target="\_blank" rel="nofollow ugc">安装教程</a>。安装 Inmp 环境。<br>（略）</p>

### 

<p>参考文章：<a href="https://link.hacpai.com/forward?goto=https%3A%2F%2Fwww.daniao.org%2F5135.html" target="\_blank" rel="nofollow ugc">宝塔面板 – Java 项目管理器安装 Solo 教程（大鸟博客）</a><br>由于我的宝塔面板 Java 项目管理器无法安装运行 Tomcat9，遂直接用软件中心的 Tomcat9。</p>

#### 

<p>（1）访问 Solo 的 Github-Release 页面：<a href="https://link.hacpai.com/forward?goto=https%3A%2F%2Fgithub.com%2Fb3log%2Fsolo%2Freleases" target="\_blank" rel="nofollow ugc">https://github.com/b3log/solo/releases</a>&nbsp;，获取最新 war 包地址。<br>（2）下载 war 包至服务器的/www/server/tomcat/webapps/目录下，由于 Tomcat 的配置，war 包将自动被压缩。（如未自动解压，尝试使用宝塔的文件管理器手动解压，或在 SSH 中执行 unzip 命令）</p>

#### 

<p><a href="https://link.hacpai.com/forward?goto=https%3A%2F%2Fimgchr.com%2Fi%2FVcOgO" target="\_blank" rel="nofollow ugc"><br>如图所示，需要设置的地方有：<br>（1）域名<br>（2）根目录，设置为解压后 solo 所在路径<br>（3）创建一个 SQL 数据库，编码选择 utf8mb4。（也可以之后进入数据库菜单建）<br>（注：此处选择 UTF-8 亦可使用，按网上的说法选择 utf8mb4 可以覆盖包括 emoji 在内更多字符，因此直接 utf8mb4 就好啦。参考：<a href="https://link.hacpai.com/forward?goto=ht">

p%3A%2F%2Fourmysql.com%2Farchives%2F1402" target="\_blank" rel="nofollow ugc">清官mysql 中 utf8 和 utf8mb4 区别</a>等) <br> (4) Solo 是一个 Java 程序, PHP 设置为纯静态</p>

#### <h4 id="3-配置Solo程序">3、配置 Solo 程序</h4>

<p> (1) 将宝塔面板解压生成的 solo 目录及子目录所有者设置为 www (原本为 root) , 权限为 750。 <br> (2) 进入 solo/WEB-INF/classes 目录, 修改 latke.properties 和 local.properties</p>

#### <h4 id="latke-properties">latke.properties</h4>

```
<pre><code class="highlight-chroma">#### Server ####
```

```
# Browser visit protocol
```

```
serverScheme=https #按需设置为http或https (需配置SSL)
```

```
serverHost=shikkaku.com
```

```
#访问域名, 注意把注释#删除
```

```
#serverPort= #留空, 不过其实默认被注释了。有些教程设置为Tomcat9默认端口8083后反向代理此处不采用这一方案。
```

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

#### <h4 id="local-properties">local.properties</h4>

```
<pre><code class="highlight-chroma">#### MySQL runtime ####
```

```
runtimeDatabase=MYSQL
```

```
jdbc.username=shikkaku_com #数据库用户名
```

```
jdbc.password=yYaBYcP #数据库密码
```

```
jdbc.driver=com.mysql.cj.jdbc.Driver
```

```
jdbc.URL=jdbc:mysql://localhost:3306/solo?useUnicode=yes&amp;characterEncoding=UTF-8&amp;useSSL=false&amp;serverTimezone=UTC
```

```
#3306/solo?应该是要与自己的数据库名一致, 此处数据库名为solo。宝塔创建的默认可能是shikkaku_com (域名) , 保持一致。
```

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

#### <h4 id="4-检查与发布">4、检查与发布</h4>

<p>打开 Tomcat 确认其配置 (在宝塔软件页面 Tomcat 管理-配置修改, 或者直接打开/www/serve/tomcat/conf/server.xml) <br> 确认以下代码无误: </p>

```
<pre><code class="highlight-chroma">    &lt;Host autoDeploy="true" name="shikkaku.com" unpackWARs="true" xmlNamespaceAware="false" xmlValidation="false"&gt;
```

```
    &lt;Context crossContext="true" docBase="/www/server/tomcat/webapps/solo-v3.6.2" path="" reloadable="true" /&gt;
```

```
#name=域名, docBase=solo程序路径。
```

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

<p>宝塔面板-网站-点击网站名-开启 Tomcat, 如图所示: <br> <a href="https://link.hacpai.com/forward?goto=https%3A%2F%2Fimgchr.com%2Fi%2FVBgJr4" target="\_blank" rel="nofollow ugc"></a></p>

<p>好啦 ~ 折腾暂时告一段落, 访问网站开始愉快写博客吧 w。 <br> (然而真诚猜想不折腾面板, 接 docker+docker-compose 的部署应该是最轻松愉快的, 舍近求远可谓极矣。修改本文时 solo 从 .6.1 更新至 3.6.2, 安装更新配置又折腾了几分钟 w。) <br> 仅作为自己一番瞎折腾的一个记录。如给诸位小白朋友们一点参考, 幸甚至哉。 <br> 兴趣使然, 不求甚解。啊, 当时满脑子想的都是至少起来啊不要 404 不要 502 啊 Orz。 </p>

### <h3 id="可选-">可选: </h3>

#### <h4 id="1-设置SSL安全证书">1、设置 SSL 安全证书</h4>

<p> (1) 确定域名解析生效且正确解析至服务器地址, 关闭 CDN 等会导致 IP 指向其他 IP 地址的务 (文件验证可能可以规避这一点, 诸君多福)。 <br> (2) 使用宝塔面板傻瓜布置。 <br> <a href="https://link.hacpai.com/forward?goto=https%3A%2F%2Fimgchr.com%2Fi%2FVcf7Bq" target="\_blank" rel="nofollow ugc"></a><br> (图中域名为演示用临时新建的, 请不要在意。) <br> (3) 确认前述 Solo 配置中, solo/WEB-INF/classes/latke.properties 中 serverScheme=https <br> (4) 完成! 尝试使用 <a href="https://link.hacpai.com/forward?goto=https%3A%2F%2Fshikkaku.com" target="\_blank" rel="nofollow ugc">

c"><https://shikkaku.com></a> 访问你的网站~! </p>  
<h4 id="2-Cloudflare免费CDN">2、Cloudflare 免费 CDN</h4>  
<p>网络上有很多介绍、推荐 Cloudflare 的文章，此处不赘述。<br>不才愚见，简单想象成一个速访问的免费代理服务器就好啦。<br>使用 Cloudflare 的优点，大致有：（1）有免费版（2）隐服务器真实 IP，降低被攻击风险（3）缓存网站的静态内容，加速访问（4）说不定真的比直连快呢（）据说国内备份主机域名可以使用与百度合作的特别版<br><strong>建议：推荐在完成 Solo 配置 SSL 配置后再进行设置，可以减少一些折腾。</strong></p>  
<h5 id="1--注册Cloudflare账号">1. 注册 Cloudflare 账号</h5>  
<p>访问 Cloudflare 官网，先免费注册一个账号 ~ </p>  
<h5 id="2--添加域名和DNS解析">2. 添加域名和 DNS 解析</h5>  
<p>添加自己的网站，并且设置 DNS 解析。（Cloudflare 君将成为 NameServer（名称服务器）！</p></p>  
<h5 id="3--更改域名的名称服务器">3. 更改域名的名称服务器</h5>  
<p>按 Cloudflare 的提示，去域名提供商那里把域名对应的 DNS（域名系统）修改为 Cloudflare 定义的 ray.ns.cloudflare.com 和 rose.ns.cloudflare.com（以官网为准，嗯。）</p>  
<h5 id="4--设置CDN和https">4. 设置 CDN 和 https</h5>  
<p><a href="https://link.hacpai.com/forward?goto=https%3A%2F%2Fimgchr.com%2Fi%2FVHPQU" target="\_blank" rel="nofollow ugc"></a><br>DNS 中添加 A 记录，域名指向服务器 IP（图中马赛克部分）。右侧橘黄色小云为开启 CloudflareCDN 状态，可以点击变为灰色（关闭 CDN，直连，只提供 DNS 服务）。</p>  
<p><a href="https://link.hacpai.com/forward?goto=https%3A%2F%2Fimgchr.com%2Fi%2FVHSJ0" target="\_blank" rel="nofollow ugc"></a><br>在完成上述 1.设置 SSL 安全证书后，设置 SSL 为 Full。<br>关于 Flexible，Full，Strict 的区别，网络上有相关解释。<br>此处选择 Full，服务器与 Cloudflare 之间的数据传输通过 SSL 密但不严格验证证书发行者，同时 Cloudflare 到网站访问者之间的数据传输也通过 SSL 加密。<br>可以避免 Flexible 服务器与 Cloudflare 之间的数据传输不通过 SSL 加密，则服务器设置为无 SSL 加（Nginx、Tomcat）。当访问者通过 https 访问网站时，可能出现静态内容（图片、JavaScript 看娘）无法显示的情况。（推测为不安全的内容被过滤了）<br>而此处选择 Strict 则会导致服务器与 Cloudflare 之间无法连接，以至于网页无法访问。（免费的 SSL 玩家不能通过验证啦。）</p>  
<h5 id="5--少女祈祷中-等就对了">5. <del>少女祈祷中</del> 等就对了</h5>  
<p><a href="https://link.hacpai.com/forward?goto=https%3A%2F%2Fimgchr.com%2Fi%2FVq0aj" target="\_blank" rel="nofollow ugc"></a><br>随便放张图，啦啦啦啦啦。<br>更新名称服务器和开启 Cloudflare 功能需要一些时间。</p></p>  
<h4 id="3-动-Tomcat-静-Nginx-分离等-">3、动（Tomcat）静（Nginx）分离等。</h4>