



黑客派

solo 博客 - 七牛云改腾讯云图床（对象存储）

作者: [Alicization](#)

原文链接: <https://hacpai.com/article/1545662942250>

来源网站: 黑客派

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

```

<h2 id="solo博客---七牛云改腾讯云图床-对象存储-">solo 博客 - 七牛云改腾讯云图床（对象存储
</h2>
<script async src="https://pagead2.googlesyndication.com/pagead/js/adsbygoogle.js"></scr
pt>
<!-- 黑客派PC帖子内嵌-展示 -->
<ins class="adsbygoogle" style="display:block" data-ad-client="ca-pub-5357405790190342"
data-ad-slot="8316640078" data-ad-format="auto" data-full-width-responsive="true"></in
>
<script>
  (adsbygoogle = window.adsbygoogle || []).push({});
</script>
<ul>
  <li> <p>solo 默认提供七牛云的对象存储，跑去注册发现还要实名，真麻烦。。 腾讯云估计大家都
名过了，那顺便就可以用了</p> </li>
  <li> <p>首先去 GitHub 把 solo 博客源码 clone 到本地</p> </li>
</ul>
<blockquote>
  <p><a href="https://link.hacpai.com/forward?goto=https%3A%2F%2Fgithub.com%2Fb3lo
%2Fsolo" target="_blank" rel="nofollow ugc">https://github.com/b3log/solo</a></p>
</blockquote>
<h2 id="1-在数据库里插入几个数据">1.在数据库里插入几个数据</h2>
<pre><code class="highlight-chroma">INSERT INTO `solo`.`b3_solo_option`(`old`,`optionVal
e`,`optionCategory`) VALUES ('qcloudBucket','aliciz-123','qcloud');
INSERT INTO `solo`.`b3_solo_option`(`old`,`optionValue`,`optionCategory`) VALUES ('qcloudD
main','https://www.aliciz.com','qcloud');
INSERT INTO `solo`.`b3_solo_option`(`old`,`optionValue`,`optionCategory`) VALUES ('qcloudR
gion','ap-guangzhou','qcloud');
INSERT INTO `solo`.`b3_solo_option`(`old`,`optionValue`,`optionCategory`) VALUES ('qcloudS
cretId','AKIDBgsgdsgm113424WX','qcloud');
INSERT INTO `solo`.`b3_solo_option`(`old`,`optionValue`,`optionCategory`) VALUES ('qcloudS
cretKey','sjfoijafSASAFÅASFA','qcloud');

</code></pre>

<p></
>
<p>至于这几个东西什么意思，可以看看腾讯云文档</p>
<p>链接：<a href="https://link.hacpai.com/forward?goto=https%3A%2F%2Fcloud.tencent.c
m%2Fdocument%2Fproduct%2F436%2F6225" target="_blank" rel="nofollow ugc">腾讯云-
对象存储-基本概念</a></p>
<h2 id="2-新增一个配置类">2.新增一个配置类</h2>
<pre><code class="language-go highlight-chroma"><span class="highlight-kn">package</
span> <span class="highlight-nx">org</span><span class="highlight-p">.</span><span cla
s="highlight-nx">b3log</span><span class="highlight-p">.</span><span class="highlight-
x">solo</span><span class="highlight-p">.</span><span class="highlight-nx">model</sp
n><span class="highlight-p">{</span>

<span class="highlight-nx">public</span> <span class="highlight-nx">class</span> <span
class="highlight-nx">QCloudOption</span> <span class="highlight-p">{</span>

<span class="highlight-cm">/**

</span><span class="highlight-cm">

```

* 腾讯云-对象存储

```

</span> <span class="highlight-cm">
                                */</span>
<span class="highlight-nx">public</span> <span class="highlight-nx">static</span> <span class="highlight-nx">final</span> <span class="highlight-nx">String</span> <span class="highlight-nx">CATEGORY_C_QCLOUD</span> <span class="highlight-p">=</span> <span class="highlight-s">"qcloud"</span> <span class="highlight-p"></span>
<span class="highlight-nx">public</span> <span class="highlight-nx">static</span> <span class="highlight-nx">final</span> <span class="highlight-nx">String</span> <span class="highlight-nx">ID_C_QCLOUD_REGION</span> <span class="highlight-p">=</span> <span class="highlight-s">"qcloudRegion"</span> <span class="highlight-p"></span>
<span class="highlight-nx">public</span> <span class="highlight-nx">static</span> <span class="highlight-nx">final</span> <span class="highlight-nx">String</span> <span class="highlight-nx">ID_C_QCLOUD_SECRET_ID</span> <span class="highlight-p">=</span> <span class="highlight-s">"qcloudSecretId"</span> <span class="highlight-p"></span>
<span class="highlight-nx">public</span> <span class="highlight-nx">static</span> <span class="highlight-nx">final</span> <span class="highlight-nx">String</span> <span class="highlight-nx">ID_C_QCLOUD_SECRET_KEY</span> <span class="highlight-p">=</span> <span class="highlight-s">"qcloudSecretKey"</span> <span class="highlight-p"></span>
<span class="highlight-nx">public</span> <span class="highlight-nx">static</span> <span class="highlight-nx">final</span> <span class="highlight-nx">String</span> <span class="highlight-nx">ID_C_QCLOUD_DOMAIN</span> <span class="highlight-p">=</span> <span class="highlight-s">"qcloudDomain"</span> <span class="highlight-p"></span>
<span class="highlight-nx">public</span> <span class="highlight-nx">static</span> <span class="highlight-nx">final</span> <span class="highlight-nx">String</span> <span class="highlight-nx">ID_C_QCLOUD_BUCKET</span> <span class="highlight-p">=</span> <span class="highlight-s">"qcloudBucket"</span> <span class="highlight-p"></span>
<span class="highlight-p">}</span>
</code> </pre>

```

```

<script async src="https://pagead2.googlesyndication.com/pagead/js/adsbygoogle.js"></script>

```

<!-- 黑客派PC帖子内嵌-展示 -->

```

<ins class="adsbygoogle" style="display:block" data-ad-client="ca-pub-5357405790190342" data-ad-slot="8316640078" data-ad-format="auto" data-full-width-responsive="true"></ins>
<script>
    (adsbygoogle = window.adsbygoogle || []).push({});
</script>
<h2 id="3-修改controller">3.修改 controller</h2>
<blockquote>
    <p>修改这个方法 org.b3log.solo.processor.FileUploadProcessor.uploadFile()</p>
</blockquote>
<pre> <code class="highlight-chroma">
//前面省略 ...

```

wwAuth auth;

```

UploadManager uploadManager = null;
String uploadToken = null;
JSONObject qiniu = null;
JSONObject qcloud = null;
COSClient cosClient = null;
final String date = DateFormatUtils.format(System.currentTimeMillis(), "yyyy/MM");
if (QN_ENABLED) {
    //      try {
    //          final BeanManager beanManager = BeanManager.getInstance();
    //          final OptionQueryService optionQueryService = beanManager.getReference(OptionQueryService.class);
    //          qiniu = optionQueryService.getOptions(Option.CATEGORY_C_QINIU);
    //          if (null == qiniu) {
    //              final String msg = "Qiniu settings failed, please visit https://hacpai.com/article/442418791213 for more details";
    //              LOGGER.log(Level.ERROR, msg);
    //              context.renderMsg(msg);
    //
    //              return;
    //          }
    //
    //          auth = Auth.create(qiniu.optString(Option.ID_C_QINIU_ACCESS_KEY), qiniu.optString(Option.ID_C_QINIU_SECRET_KEY));
    //          uploadToken = auth.uploadToken(qiniu.optString(Option.ID_C_QINIU_BUCKET), null, 3600 * 6, null);
    //          uploadManager = new UploadManager(new Configuration());
    //      } catch (final Exception e) {
    //          final String msg = "Qiniu settings failed, please visit https://hacpai.com/article/142418791213 for more details";
    //          LOGGER.log(Level.ERROR, msg);
    //          context.renderMsg(msg);
    //
    //          return;
    //      }
    final BeanManager beanManager = BeanManager.getInstance();
    final OptionQueryService optionQueryService = beanManager.getReference(OptionQueryService.class);

    qcloud = optionQueryService.getOptions(QCloudOption.CATEGORY_C_QCLOUD);

```

```

COSCredentials cred = new BasicCOSCredentials(qcloud.optString(QCloudOption.ID_C_QCLOUD_SECRET_ID), qcloud.optString(QCloudOption.ID_C_QCLOUD_SECRET_KEY));
ClientConfig clientConfig = new ClientConfig(new Region(qcloud.optString(QCloudOption.ID_C_QCLOUD_REGION)));
cosClient = new COSClient(cred, clientConfig);

}

for (int i = 0; i < files.length; i++) {
    if (!foundRenderFunctionForNode(files[i], Tokens)) {
        final FileUpload file = files[i];
        String originalName = file.getHeader().getFileName();
        originalName = originalName.replaceAll("\\W", "");
        try {
            String suffix = StringUtils.substringAfterLast(file.getHeader().getFileName(), ".");
            final String contentType = file.getHeader().getContentType();
            if (StringUtils.isBlank(suffix)) {
                String[] exts = MimeTypes.findExtensionsByMimeType(contentType, false);
                if (exts != null) {
                    suffix = exts[0];
                } else {
                    suffix = StringUtils.substringAfter(contentType, "/");
                }
            }

            // uploadManager.put(file.getInputStream(), originalName, uploadToken, null, contentType);
            // succMap.put(originalName, qiniu.optString(QiniuOption.ID_C_QINIU_DOMAIN) + "/" + originalName);
        } else {
            try (final OutputStream output = new FileOutputStream(Solos.UPLOAD_DIR_PATH + originalName);
                 final InputStream input = file.getInputStream()) {
                IOUtils.copy(input, output);
            }
            succMap.put(originalName, Latkes.getServePath() + "/upload/" + originalName);
        }
    }
}

```

```
} catch (final Exception e) {  
    LOGGER.log(Level.WARN, "Uploads file failed", e);  
  
}
```

//省略部分代码 ..

</code> </pre>

```
<script async src="https://pagead2.googlesyndication.com/pagead/js/adsbygoogle.js"></script>
```

```
<!-- 黑客派PC帖子内嵌-展示 -->
```

```
<ins class="adsbygoogle" style="display:block" data-ad-client="ca-pub-5357405790190342"  
data-ad-slot="8316640078" data-ad-format="auto" data-full-width-responsive="true"></in  
>
```

```
<script>  
    (adsbygoogle = window.adsbygoogle || []).push({});  
</script>
```

```
<h2 id="4-修改nginx-conf">4.修改 nginx.conf</h2>
```

```
<pre><code class="highlight-chroma">server {  
    listen 443;  
    server_name www.aliciz.com;
```

```
}
```

</code> </pre>

<p>这样就完成啦，记得源码打包替换的时候，把原有的服务器配置备份一下。
 其实 solo 在理这个文件上传的地方可以再写得扩展性更强一些，这样就可以切换多家云服务厂商</p>