Java ProGuard代码混淆

1. 问题描述

工作中需要把一套java web项目打包卖给客户

由于java项目太容易反编译了，所以考虑了几种解决方案：

1. 对代码进行混淆，增加代码的阅读难度
2. 对class文件进行加密，当java虚拟机读取的class文件的时候再解密

下面是我对代码进行混淆过程

1. 使用ProGuard进行代码混淆

<!-- ProGuard混淆插件-->

<plugin>

<groupId>com.github.wvengen</groupId>

<artifactId>proguard-maven-plugin</artifactId>

<version>2.0.11</version>

<executions>

<execution>

<!-- 混淆时刻，这里是打包的时候混淆-->

<phase>package</phase>

<goals>

<!-- 使用插件的什么功能，当然是混淆-->

<goal>proguard</goal>

</goals>

</execution>

</executions>

<configuration>

<!-- 是否将生成的PG文件安装部署-->

<attach>true</attach>

<!-- 是否混淆-->

<obfuscate>true</obfuscate>

<!-- 指定生成文件分类 -->

<attachArtifactClassifier>pg</attachArtifactClassifier>

<options>

<!-- JDK目标版本1.7-->

<option>-target 1.7</option>

<!-- 不做收缩（删除注释、未被引用代码）-->

<option>-dontshrink</option>

<!-- 不做优化（变更代码实现逻辑）-->

<option>-dontoptimize</option>

<!-- 不路过非公用类文件及成员-->

<option>-dontskipnonpubliclibraryclasses</option>

<option>-dontskipnonpubliclibraryclassmembers</option>

<!-- 优化时允许访问并修改有修饰符的类和类的成员 -->

<option>-allowaccessmodification</option>

<!-- 确定统一的混淆类的成员名称来增加混淆-->

<option>-useuniqueclassmembernames</option>

<!-- 不混淆所有包名，本人测试混淆后WEB项目问题实在太多，毕竟Spring配置中有大量固定写法的包名-->

<option>-keeppackagenames</option>

<!-- 不混淆所有特殊的类-->

<option>-keepattributes Exceptions,InnerClasses,Signature,Deprecated,SourceFile,LineNumberTable,LocalVariable\*Table,\*Annotation\*,Synthetic,EnclosingMethod</option>

<!-- 不混淆所有的set/get方法，毕竟项目中使用的部分第三方框架（例如Shiro）会用到大量的set/get映射-->

<option>-keepclassmembers public class \* {void set\*(\*\*\*);\*\*\* get\*();}</option>

<!-- 不混淆job包下的所有类名，且类中的方法也不混淆-->

<option>-keep class com.chinatelecom.gz.wy.zhukun.shiro\_spring.job.\*\* { &lt;methods&gt;; }</option>

<!-- 不混淆filter包下的所有类名，这里主要是对Shiro的路踢人过滤器混淆，对类的属性和方法进行了混淆-->

<option>-keep class com.chinatelecom.gz.wy.zhukun.shiro\_spring.filter.\*\* </option>

<!-- 不混淆凭证包下的所有类名，但对类中的属性、方法进行混淆，原因是Spring配置中用到了这个类名-->

<option>-keep class com.chinatelecom.gz.wy.zhukun.shiro\_spring.credntials.\*\* </option>

<!-- 混淆目的同上，这个是拦截器的包，包中有防止重复提交的拦截器-->

<option>-keep class com.chinatelecom.gz.wy.zhukun.shiro\_spring.interceptor.\*\* </option>

<!-- 混淆目的同上，这个是域包，包中有用户登录域-->

<option>-keep class com.chinatelecom.gz.wy.zhukun.shiro\_spring.realm.\*\* </option>

<!-- 不混淆model包中的所有类以及类的属性及方法，实体包，混淆了会导致ORM框架及前端无法识别-->

<option>-keep class com.chinatelecom.gz.wy.zhukun.shiro\_spring.model.\*\* {\*;}</option>

<!-- 以下两个包因为大部分是Spring管理的Bean，不对包类的类名进行混淆，但对类中的属性和方法混淆-->

<option>-keep class com.chinatelecom.gz.wy.zhukun.shiro\_spring.service.\*\* </option>

<option>-keep class com.chinatelecom.gz.wy.zhukun.shiro\_spring.dao.\*\*</option>

</options>

<outjar>${project.build.finalName}-pg.jar</outjar>

<!-- 添加依赖，这里你可以按你的需要修改，这里测试只需要一个JRE的Runtime包就行了 -->

<libs>

<lib>${java.home}/lib/rt.jar</lib>

</libs>

<!-- 加载文件的过滤器，就是你的工程目录了-->

<inFilter>com/chinatelecom/gz/wy/zhukun/shiro\_spring/\*\*</inFilter>

<!-- 对什么东西进行加载，这里仅有classes成功，毕竟你也不可能对配置文件及JSP混淆吧-->

<injar>classes</injar>

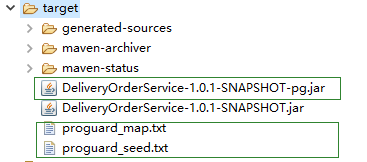
<!-- 输出目录-->

<outputDirectory>${project.build.directory}</outputDirectory>

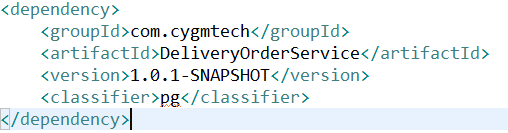
</configuration>

</plugin>

打包完后会多出三个文件



1. 使用classifier指定打包是否用混淆代码



当我们使用classifier 指定包的版本后，那么项目会引用

DeliveryOrderService-1.0.1-SNAPSHOT-pg.jar 这个包

提示：最好不要用spring项目，问题太多。