



链滴

使用 jasypt 加密配置

作者: [96XL](#)

原文链接: <https://ld246.com/article/1638330383494>

来源网站: [链滴](#)

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

前言

  项目的配置文件中数据库密码以及redis密码是明文形式，要求改造成密文网上查了一下，决定使用jasypt进行加密改造，记录一下改造过程。

正文

1. pom文件中引入jar包，附上项目github地址 [ulisesbocchio/jasypt-spring-boot: Jasypt integration for Spring boot \(github.com\)](https://github.com/ulisesbocchio/jasypt-spring-boot)，版本可自行更换。

```
<!-- jasypt加密jar -->
<dependency>
  <groupId>com.github.ulisesbocchio</groupId>
  <artifactId>jasypt-spring-boot-starter</artifactId>
  <version>2.1.2</version>
</dependency>
```

2. 配置文件中添加密钥配置。

```
jasypt.encryptor.password=XXX
```

3. 对明文密码进行加密，获取密文字符串，jasypt默认使用StringEncryptor来进行加解密，我们也以自定义自己的加解密类来替换它，具体使用可自行百度。

```
@Autowired
private StringEncryptor stringEncryptor;
```

```
@Test
public void encrypt() {
  System.out.println(stringEncryptor.encrypt("password"));
}
```

4. 修改配置文件中密码，格式为ENC(密文)，如：

```
spring.datasource.password=ENC(OJkStww+GLtKyErZNQBtKNeXCHCFQs5X)
```

  到此为止加密改造已经实现了，改动量并不大，还是非常简单的。jasypt除最基本的密钥配置，还有其他的一些配置，这些配置没有的话会使用默认的值，这些配置参数的意义自行百度。在启动项目时观察日志可以看到，如果没有特殊要求，使用默认即可。

```
2021-11-10 17:08:44.109 INFO 29344 --- | restartedMain | org.springframework.boot.autoconfigure.jasypt.JasyptStringEncryptor : String Encryptor custom bean not found with name 'jasyptStringEncryptor'. Initializing default String Encryptor
2021-11-10 17:08:44.101 INFO 29344 --- | restartedMain | org.springframework.boot.autoconfigure.jasypt.JasyptStringEncryptor : Encryptor config not found for property jasypt.encryptor.algorithm, using default value: AESwithPAD512and256
2021-11-10 17:08:44.101 INFO 29344 --- | restartedMain | org.springframework.boot.autoconfigure.jasypt.JasyptStringEncryptor : Encryptor config not found for property jasypt.encryptor.keyObtentionIterations, using default value: 1000
2021-11-10 17:08:44.102 INFO 29344 --- | restartedMain | org.springframework.boot.autoconfigure.jasypt.JasyptStringEncryptor : Encryptor config not found for property jasypt.encryptor.password, using default value: j
2021-11-10 17:08:44.102 INFO 29344 --- | restartedMain | org.springframework.boot.autoconfigure.jasypt.JasyptStringEncryptor : Encryptor config not found for property jasypt.encryptor.providerClassName, using default value: null
2021-11-10 17:08:44.102 INFO 29344 --- | restartedMain | org.springframework.boot.autoconfigure.jasypt.JasyptStringEncryptor : Encryptor config not found for property jasypt.encryptor.saltGeneratorClassName, using default value: org.jasypt.salt
  RandomSaltGenerator
2021-11-10 17:08:44.104 INFO 29344 --- | restartedMain | org.springframework.boot.autoconfigure.jasypt.JasyptStringEncryptor : Encryptor config not found for property jasypt.encryptor.ivGeneratorClassName, using default value: org.jasypt.salt
  NoIvGenerator
2021-11-10 17:08:44.105 INFO 29344 --- | restartedMain | org.springframework.boot.autoconfigure.jasypt.JasyptStringEncryptor : Encryptor config not found for property jasypt.encryptor.stringOutputType, using default value: base64
```

注：1.加密时，多次加密结果会变化，均可使用，无需在意。

  2.密钥放在配置文件中可能会被泄露，生产环境建议使用启动参数的形式传密钥。

  3.非springboot项目使用此jar包也可以实现，具体使用可自行百度。