

Spring cloud config encryption decryption

<https://www.appsdeveloperblog.com/spring-cloud-config-asymmetric-encryption-and-decryption/>

```
spring.cloud.config.server.git.uri=https://github.com/mithunmohan912/git-localconfig-repo
spring.cloud.config.server.git.username=mithunmohan912@gmail.com
spring.cloud.config.server.git.password={cipher}73e73ca258882e0ae1bfe6c0fcfbdea1046c64b3eba177f
b3d9765c4149fa6fb
encrypt.key=jsakhdfkhgwegqwekjhwkjqwjkeggjwekjg132 >>>> this is the salt value
```

Then <http://localhost:9092/encrypt> > config server encrypt endpoint POST body with data to encrypt  
<http://localhost:9092/decrypt> > config server decrypt endpoint POST body with data to decrypt

Place the git password as encrypted one you got from /encrypt api and append {cipher} to it as done above

JMS

<https://www.devglan.com/spring-boot/spring-boot-jms-activemq-example>

Zuul

[https://cloud.spring.io/spring-cloud-netflix/multi/multi\\_\\_router\\_and\\_filter\\_zuul.html](https://cloud.spring.io/spring-cloud-netflix/multi/multi__router_and_filter_zuul.html)

Zuul with JWT

<https://www.baeldung.com/spring-security-zuul-oauth-jwt>

Internationalization >

<https://www.javatpoint.com/restful-web-services-internationalization>

Jpa+ hazelcaste

<https://dzone.com/articles/jpa-caching-with-hazelcast-hibernate-and-spring-bo>

Bom > as expaliained to handle multiple versions

[https://maven.apache.org/guides/introduction/introduction-to-dependency-mechanism.html#Dependency\\_Scope](https://maven.apache.org/guides/introduction/introduction-to-dependency-mechanism.html#Dependency_Scope)