RoarCTF2019web题-easy_java writeup

原创

silencediors 于 2019-10-16 10:00:02 发布 5982 收藏 7

文章标签： RoarCTF2019 黑客大赛 网络安全 WEB安全

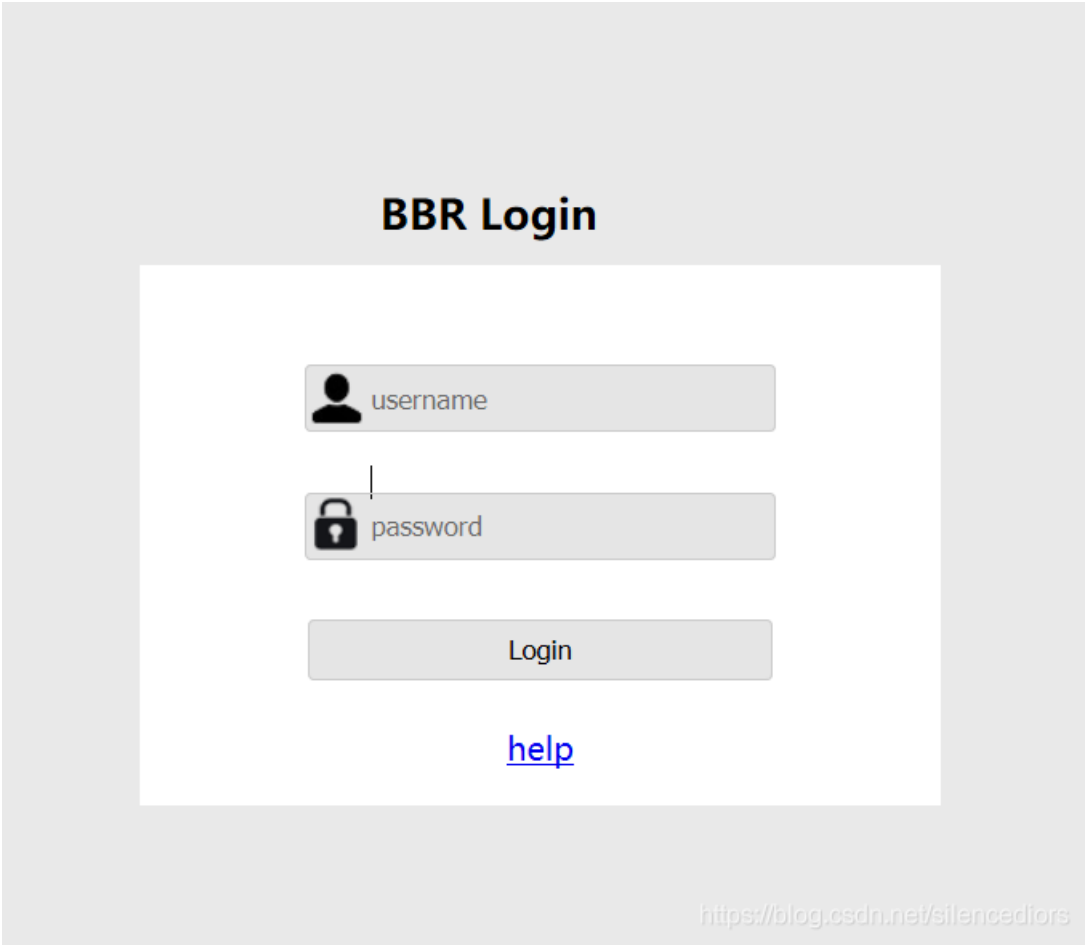
本文链接：https://blog.csdn.net/silencediors/article/details/102579567

RoarCTF2019web题-easy_java

作为团队最强的web选手（就我一个），这次的比赛也没时间打，把简单的题放一下，前面要说下，做web宁可把题目想简单，也不要想复杂，因为现在的web题出题人越来越**厉害（gou）**（怀念以前查看源码就得flag的时代），出题的把握可能也没那么好，思维多次引导的话就可能走岔路，就像这题我开始一直在想反序列化和upload流上传的事，甚至是tomcat那个版本的RCE。。。。。最后发现就是个包含。。。话不多说，上正题。

此题环境见https://buuoj.cn/challenges

首先是一个登录界面，如下图所示：



然后，猜了下账号密码，一次性猜对了admin/admin888，但是进去没什么卵用，如下图所示

然后啥也没有，只能看到个图片的真实路径，就没了，真的没了，然后出题人还各种恶意引导，真的够够的。看登录界面有个help，点进去看无法下载：
结构是hxxp://127.0.0.1/Downfile?filename=help.docx，回显如下

```
java.io.FileNotFoundException:{help.docx}
```


这种形式有经验的都会换下请求方式，结果就可以了，初步推测此处的利用包含漏洞找flag文件。
首先报错是tomcat，包含下tomcat的web,xml试试，结果真的可以
hxxp://127.0.0.1/Downfile?filename=WEB-INF/web.xml

```
<web-app xmlns="http://xmlns.jcp.org/xml/ns/javaee" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:sc
hemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-app_4_0.xsd" version="4.
0">
<welcome-file-list>
<welcome-file>Index</welcome-file>
</welcome-file-list>
<servlet>
<servlet-name>IndexController</servlet-name>
<servlet-class>com.wm.ctf.IndexController</servlet-class>
</servlet>
<servlet-mapping>
<servlet-name>IndexController</servlet-name>
<url-pattern>/Index</url-pattern>
</servlet-mapping>
<servlet>
<servlet-name>LoginController</servlet-name>
<servlet-class>com.wm.ctf.LoginController</servlet-class>
</servlet>
<servlet-mapping>
<servlet-name>LoginController</servlet-name>
<url-pattern>/Login</url-pattern>
</servlet-mapping>
<servlet>
<servlet-name>DownloadController</servlet-name>
<servlet-class>com.wm.ctf.DownloadController</servlet-class>
</servlet>
<servlet-mapping>
<servlet-name>DownloadController</servlet-name>
<url-pattern>/Download</url-pattern>
</servlet-mapping>
<servlet>
<servlet-name>FlagController</servlet-name>
<servlet-class>com.wm.ctf.FlagController</servlet-class>
</servlet>
<servlet-mapping>
<servlet-name>FlagController</servlet-name>
<url-pattern>/Flag</url-pattern>
</servlet-mapping>
</web-app>
```

注意观察有个FlagController处理/Flag，请求/Flag试试：

HTTP Status 500 – Internal Server Error

Type Exception Report

Message com/wm/ctf/FlagController (wrong name: FlagController)

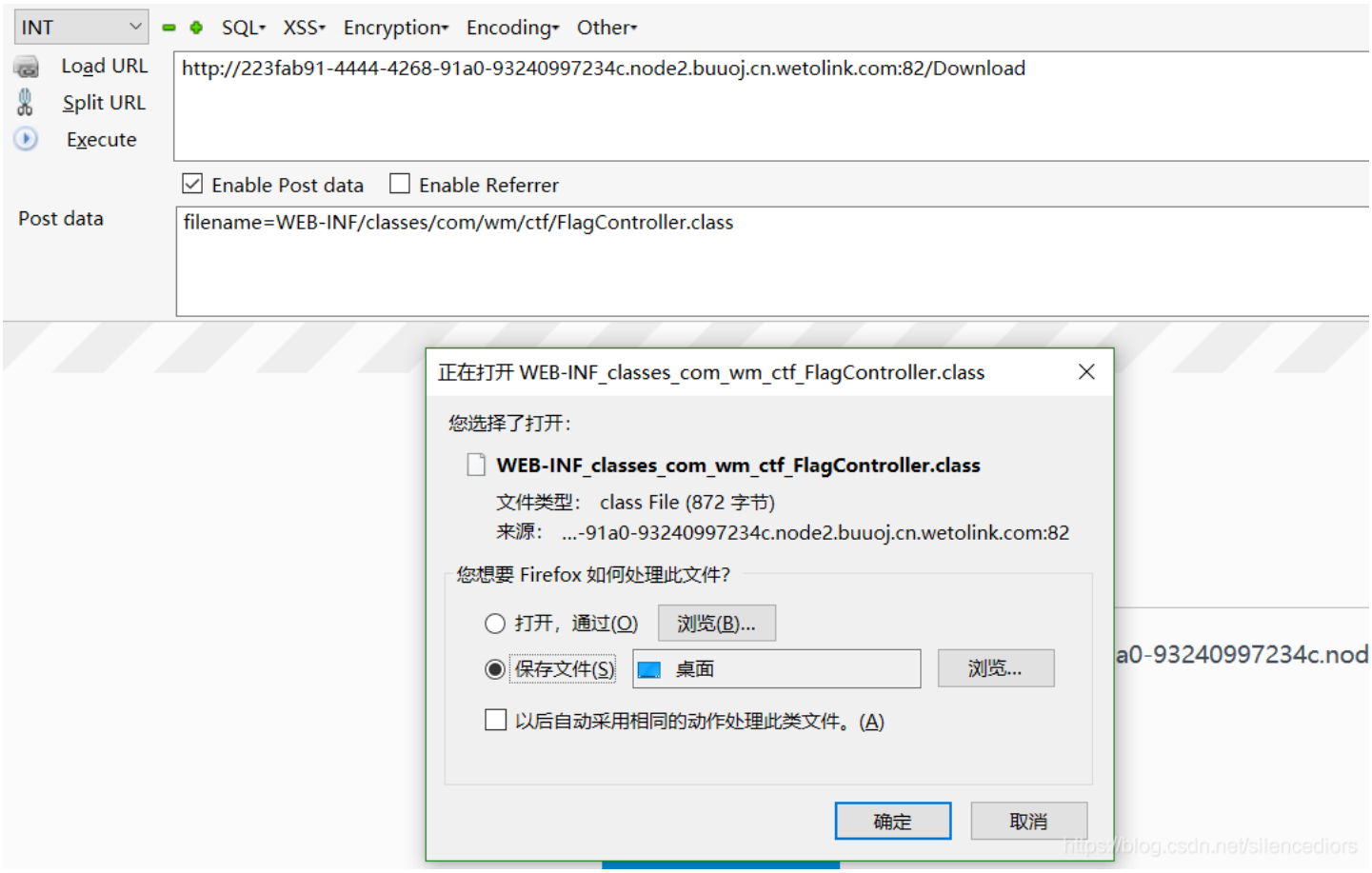
Description The server encountered an unexpected condition that prevented it from fulfilling the request.

Exception

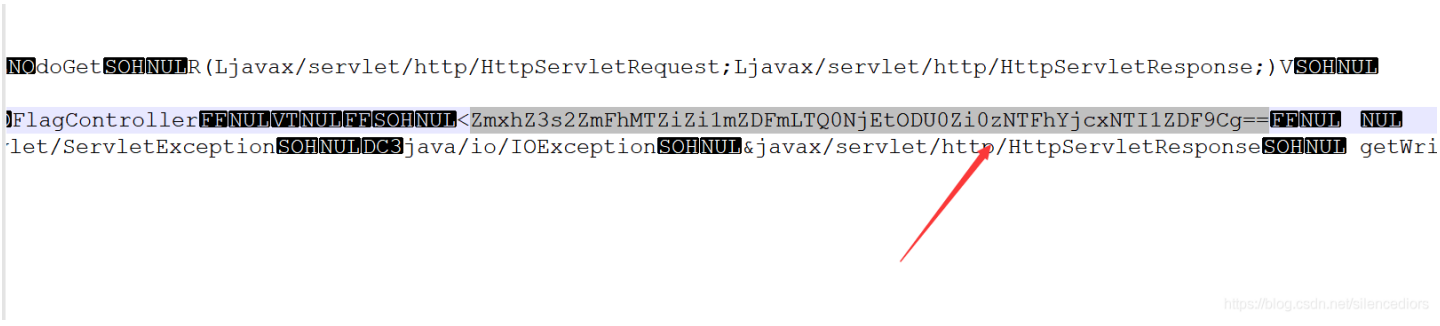
```
java.lang.NoClassDefFoundError: com/wm/ctf/FlagController (wrong name: FlagController)
        java.lang.ClassLoader.defineClass1(Native Method)
        java.lang.ClassLoader.defineClass(ClassLoader.java:763)
        java.security.SecureClassLoader.defineClass(SecureClassLoader.java:142)
        org.apache.catalina.loader.WebappClassLoaderBase.findClassInternal(WebappClassLoaderBase.java:228
        org.apache.catalina.loader.WebappClassLoaderBase.findClass(WebappClassLoaderBase.java:811)
        org.apache.catalina.loader.WebappClassLoaderBase.loadClass(WebappClassLoaderBase.java:1260)
        org.apache.catalina.loader.WebappClassLoaderBase.loadClass(WebappClassLoaderBase.java:1119)
        org.apache.catalina.authenticator.AuthenticatorBase.invoke(AuthenticatorBase.java:488)
        org.apache.catalina.valves.ErrorReportValve.invoke(ErrorReportValve.java:81)
        org.apache.catalina.valves.AbstractAccessLogValve.invoke(AbstractAccessLogValve.java:650)
```

```
org.apache.catalina.connector.CoyoteAdapter.service(CoyoteAdapter.java:342)
org.apache.coyote.http11.Http11Processor.service(Http11Processor.java:803)
org.apache.coyote.AbstractProcessorLight.process(AbstractProcessorLight.java:66)
org.apache.coyote.AbstractProtocol$ConnectionHandler.process(AbstractProtocol.java:790)
org.apache.tomcat.util.net.NioEndpoint$SocketProcessor.doRun(NioEndpoint.java:1459)
```

注意抛错路径，然后我们结合tomcat的项目存放路径经验试试下载FlagController.class试试
果然是有的



发现类似base64编码的东西，解码果然是flag




```
flag{6faa16bf-fd1f-4461-854f-351ab71525d1}
```

```
ZmxhZ3s2ZmFhMTZiZi1mZDFmLTQ0NjEtODU0Zi0zNTFhYjcxNTI1ZDF9Cg==
```

总结下吧，这题叫easy_java，的确是一道很简单的web题，但是需要对java容器和项目存放位置比较了解，所以作为web选手，一定要对几大语言的容器，项目环境，有所了解，也说明出题人越来越**厉害（gou）**了。