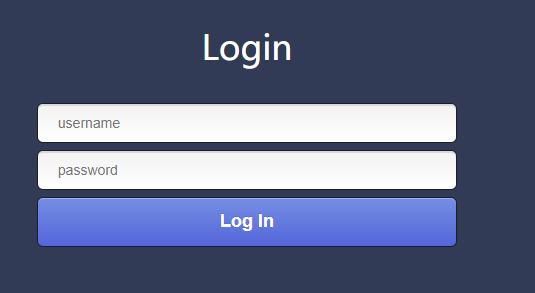
Bugku的一道web题目（login3(SKCTF)）



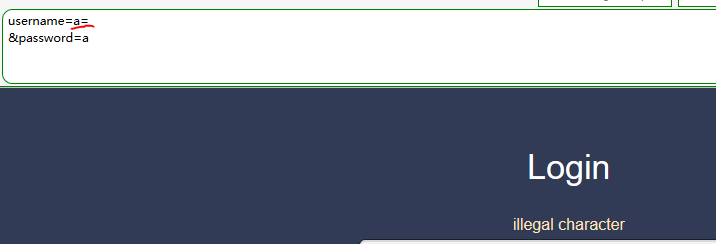
[链接](http://118.89.219.210:49167/)

题目界面



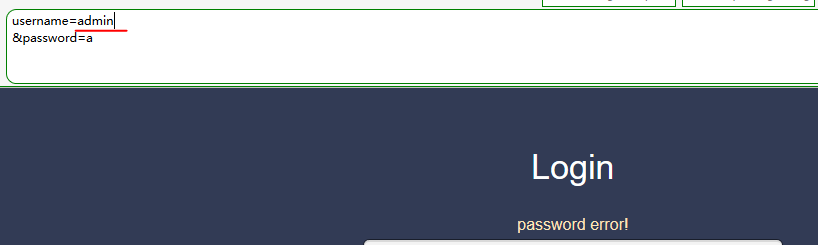
看题目的提示，盲注，就是这个是没有回显的，那就是说，注入点就在这两个参数之间，测试一下有什么过滤的，最后发现过滤了 = ，and like union where exists 空格 等等

那最后我们能利用的就很少了，但是要记住出题人永远都会故意留一些函数可以执行的



看到这些登陆框的，试一下常见用户名

这里尝试了admin，发现用户名对了

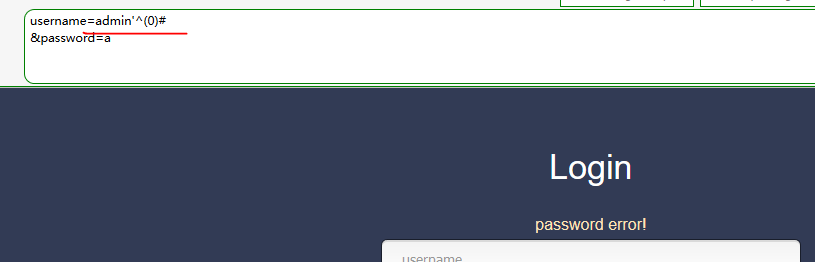


提示密码错误

看到这里，有个想法，只要在用户名这里添加注入语句，只要令用户名这里查询正确，那么就可以知道怎样进行一个注入了

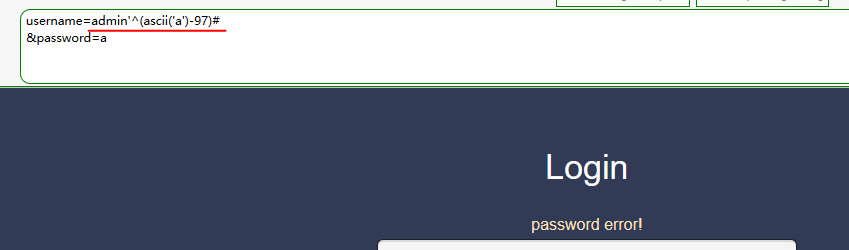
这里用到了一个异或的注入

构造username=admin'^(0)#



然后发现用户名正确了，那就是当括号里面为0，也就是为假的时候，用户名正确（如果不理解的话，查一下异或是什么意思就行了）

由于这里不能用 = 号，那就用 - 号，只要相减等于0，那就可以了，例如：username=admin'^(ascii('a')-97)#



按照这个思路，首先跑出user()





查到了用户名，那就可以进行跑其他数据了（为什么要跑user()，这个是个人习惯...）

但是由于这里过滤了 , 空格 所以我没有想到用什么来代替这两个

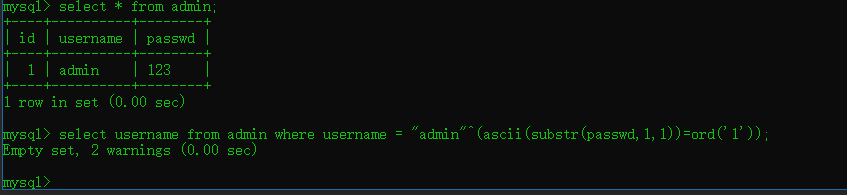


这段语句不知道为什么不对

这里行不通，那就走另外的

我们按照平时写代码的习惯，这里把用户名输入，然后在数据库，如果能查到，那就是正确的，那就是说，这里的代码里面已经写好了表了，我们不需要找那个表，我们直接查那个值就好了

例子：



这里查到是空的，那是因为后面的值为1了，不要弄混

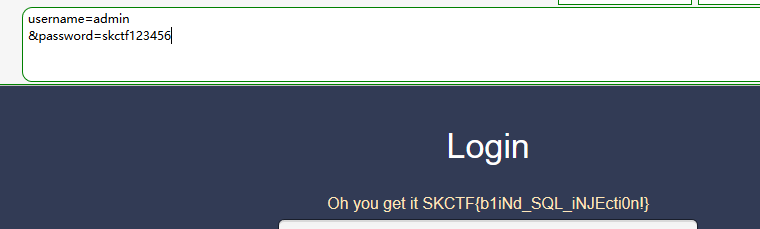
那这道题就解得很简单了，直接查就好了

最后查到password：



这段明显得MD5，然后就查一下，推荐[md5](https://somd5.com/)网站

然后登陆



[脚本](https://github.com/niechaojun/Bugku-web-login3-SKCTF-)