




mysql猜测出口令_[GXYCTF2019] BabySqli WriteUp

原创

惊奇影像  于 2021-02-28 11:15:39 发布  49  收藏

文章标签: [mysql猜测出口令](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: https://blog.csdn.net/weixin_35910783/article/details/114882658

版权

这道题肯定让很多人都一头雾水, 不知从何下手, 其实这道题和 WeChall 里的一道题基本相同, 考察的是用户名和密码分开检验。高血压的这道题和 WeChall 的 Training: MySQL II 解题思路互通, Payload也互通。

大家如果去做一下 WeChall 的下面这两道题的话, 做这道题的思路就会清晰很多。



然后再来简单讲讲这道题,

首先我们可以看到返回密码错误的页面源码里有一串字符串, base32再base64解密之后是 `select * from user where username = '$name'`

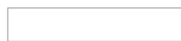


然后用常规注入的手段可以测出user这个表一共有三列, 猜测分别为id, username, password。

之前我们有提到这道题考的是用户名和密码分开检验, 也就是说它是先检验username, 把username对应的所有字段都查出来后, 再检验密码能不能和查出来的密码对上, 检验密码的过程可能会有一个md5的加密。

登录验证的流程已经说清楚了, 先做一个小测试。

用mysql创建一个表叫user, 创建三个列 id, username, password, 这时如果执行一个查询语句: `select * from user where username = 0 union select 1,'admin',md5('abc');` 则会返回以下结果:



这样的话思路就很清晰了, 我们先在用户名处输入 `1' union select 1,'admin','900150983cd24fb0d6963f7d28e17f72'#`, 得到的是上图的结果。密码处我们再输入一个上图密码md5加密之前的密码 也就是abc 即可绕过检验, 成功登陆admin账户

Payload:

```
username = 1' union select 1,'admin','900150983cd24fb0d6963f7d28e17f72'#
```

```
password = abc
```