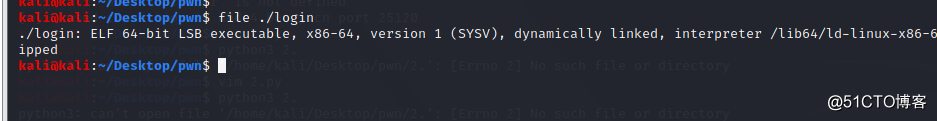
<https://blog.51cto.com/u_15127510/3756086>

阅读数172

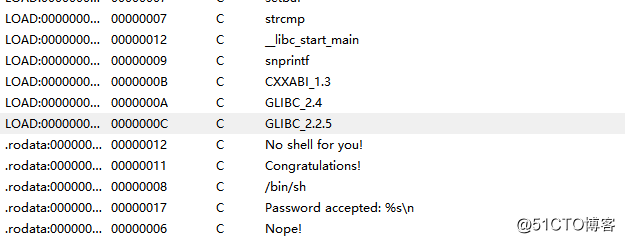
学了一段时间的堆溢出现在继续做题，

例行检查一下



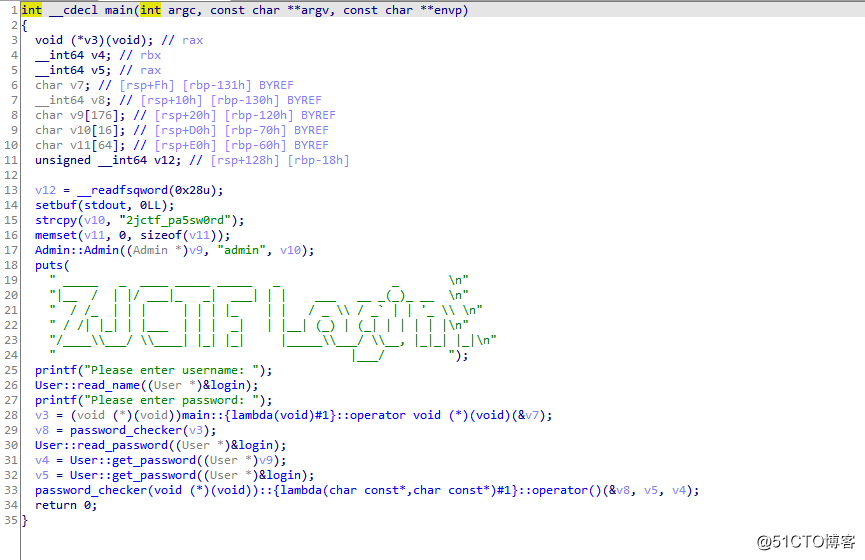
 64位的程序放入ida中

shift+f12查看程序函数



 可以看到非常明显的后门程序

查看主函数

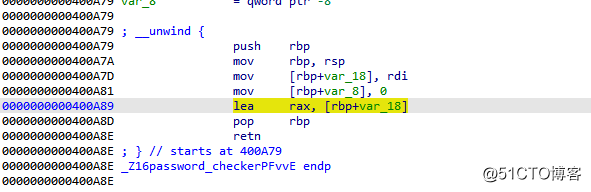


 发现了程序给的账号和密码，但是没有看到明显的栈溢出漏洞，经过查询，知道了漏洞的位置

在密码的检查页面，我们进入汇编查看

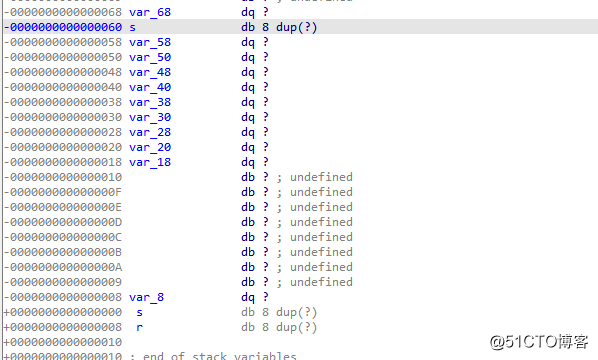


可以看到程序是call rax后跳转到程序的，我们往下查看rax的源头



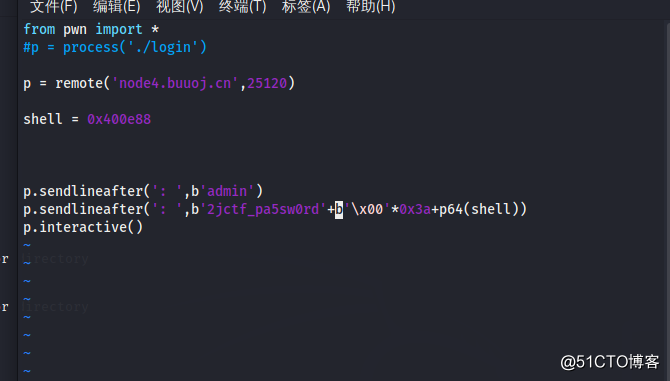
可以看到rax是var\_18的值于是我们可以通过栈溢出覆盖var\_18的值

分别进入输入密码和输入账号的程序中查看，可以看到输入密码的s的栈下面有var\_18



因为密码的长度是11所以溢出0x3a个字节再覆盖后门拿到shell

完整exp如下

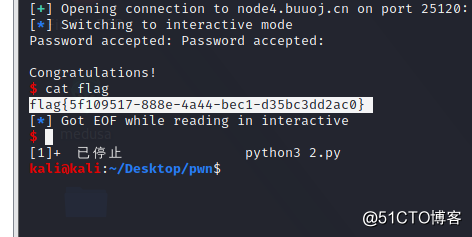


注意！！！ \x00是为了截断strcmp



 只是str类函数的特性

成功获得flag



 结束！！！！！