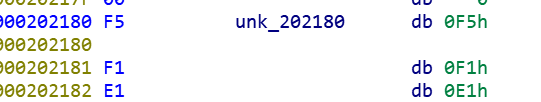
## **<https://www.cnblogs.com/harmonica11/p/13073221.html>**

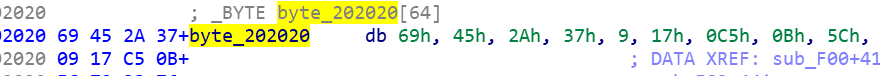
## **[GWCTF 2019]babyvm[#](https://www.cnblogs.com/harmonica11/p/13073221.html" \l "2649228407)**

这题有坑，就给了vm的实现，opcode和flag以及check函数都是假的

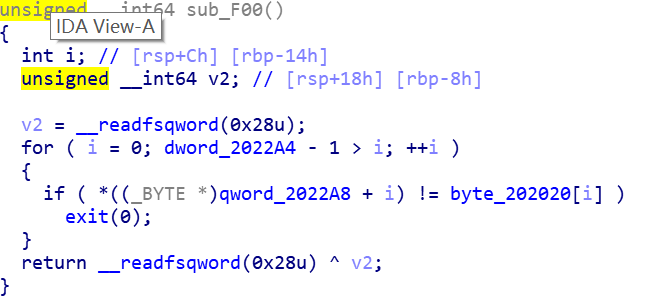
真正的opcode在

[](https://img2020.cnblogs.com/blog/1770991/202006/1770991-20200621180219307-1363790677.png)

flag在

[](https://img2020.cnblogs.com/blog/1770991/202006/1770991-20200621180310522-779994665.png)

函数在

[](https://img2020.cnblogs.com/blog/1770991/202006/1770991-20200621180543420-335363981.png)

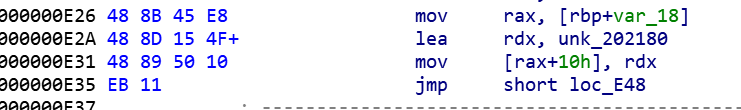
直接用angr会跑出假flag：This\_is\_not\_flag\_233

直接分析vm也可以，但是我懒。。。

patch一下程序

如下图前两个位置把vm字节码改为unk\_202180,第三张图将验证函数修改为sub\_f00

[](https://img2020.cnblogs.com/blog/1770991/202006/1770991-20200621180936567-443575875.png)

[](https://img2020.cnblogs.com/blog/1770991/202006/1770991-20200621181000496-910383637.png)

[](https://img2020.cnblogs.com/blog/1770991/202006/1770991-20200621181023384-882128731.png)

[IMG_262](https://www.cnblogs.com/harmonica11/p/javascript:void(0);)

import angr

def main():

p=angr.Project("attachment",auto\_load\_libs=False)

sm=p.factory.simulation\_manager(p.factory.entry\_state())

sm.explore(find=0x401081)

return sm.found[0].posix.dumps(0)

if \_\_name\_\_=='\_\_main\_\_':

print(main())