2024.12.3

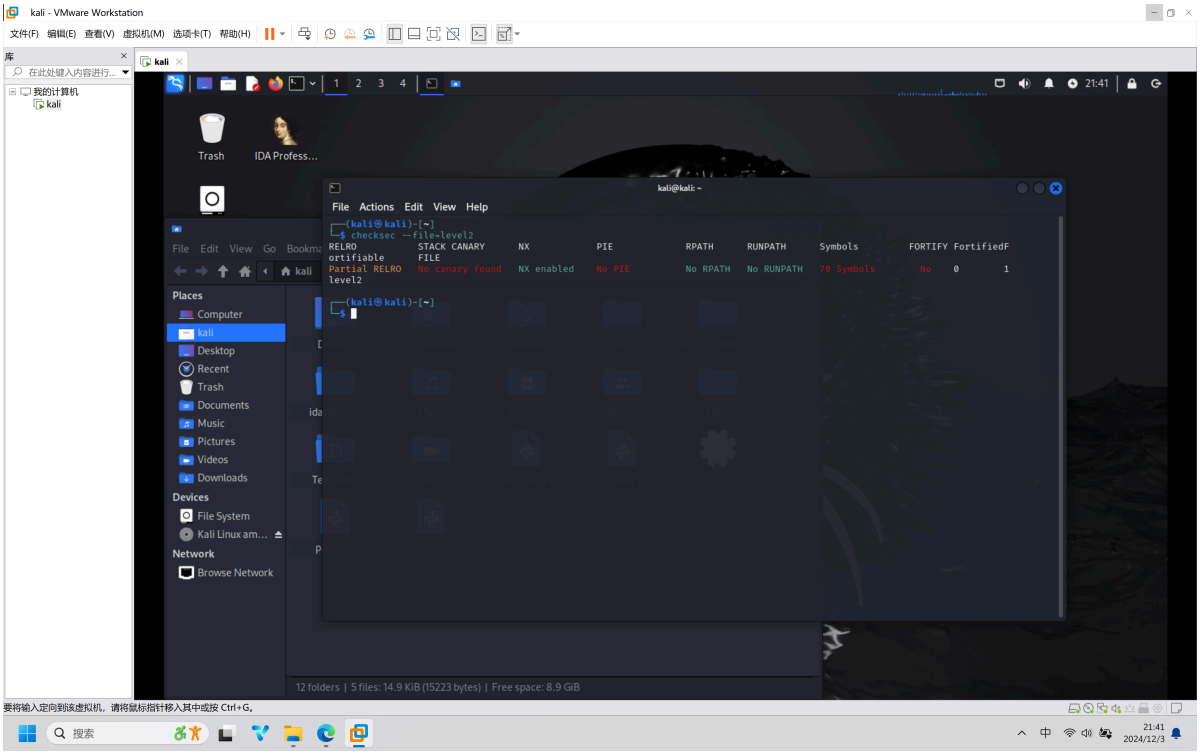
PWN

jarvisoj_level2

> 做题人:焦昱淇
> 题目url:https://buuoj.cn/challenges#jarvisoj_level2
> 知识点:栈溢出

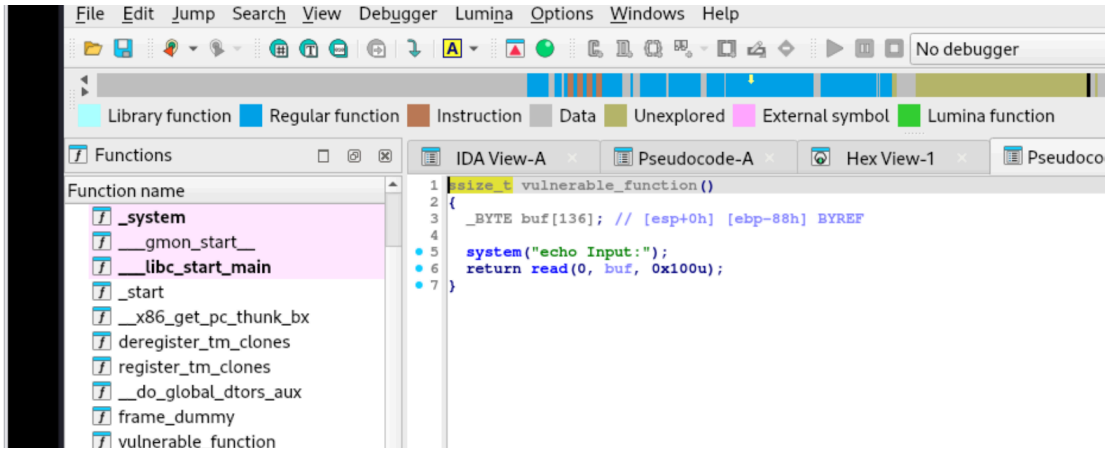
checksec查看

无保护 32位

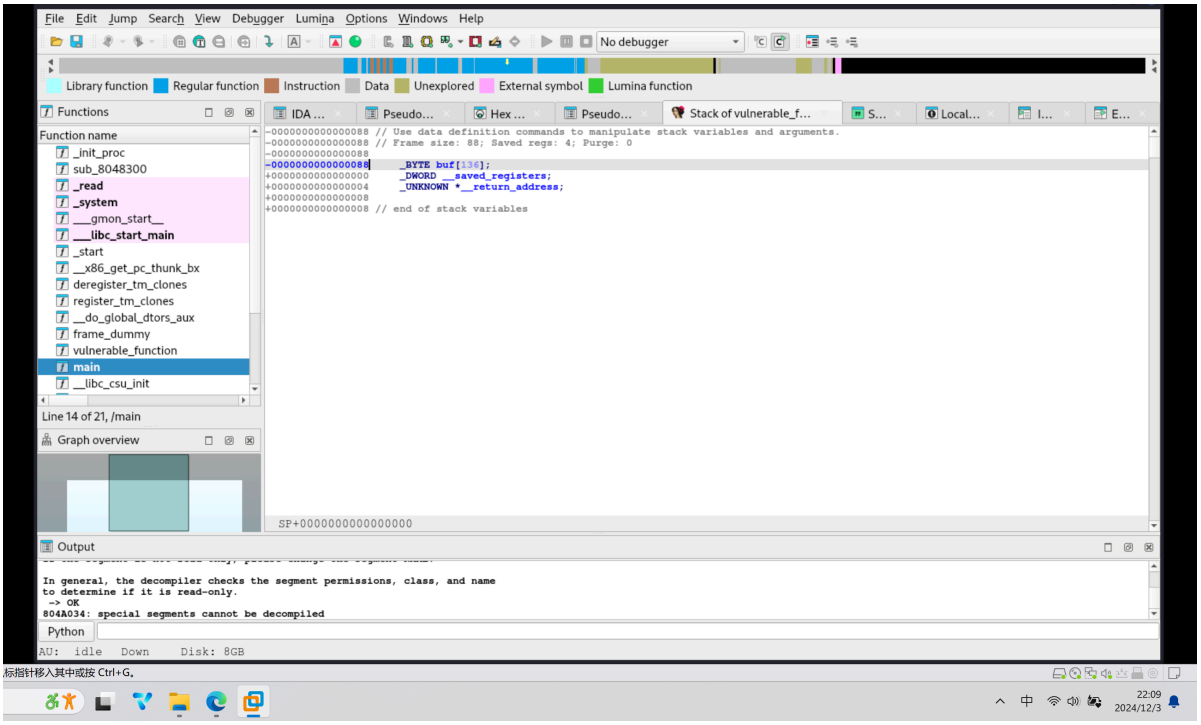


IDA查看

发现可执行的漏洞 read buf可以溢出

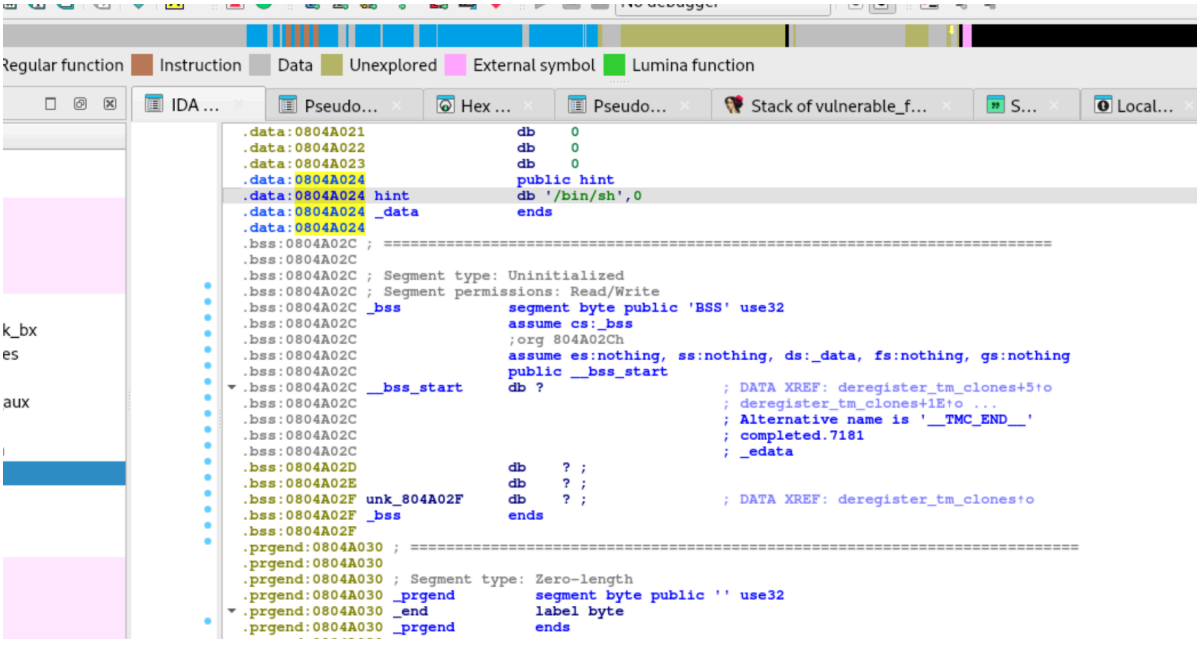


发现起止位置

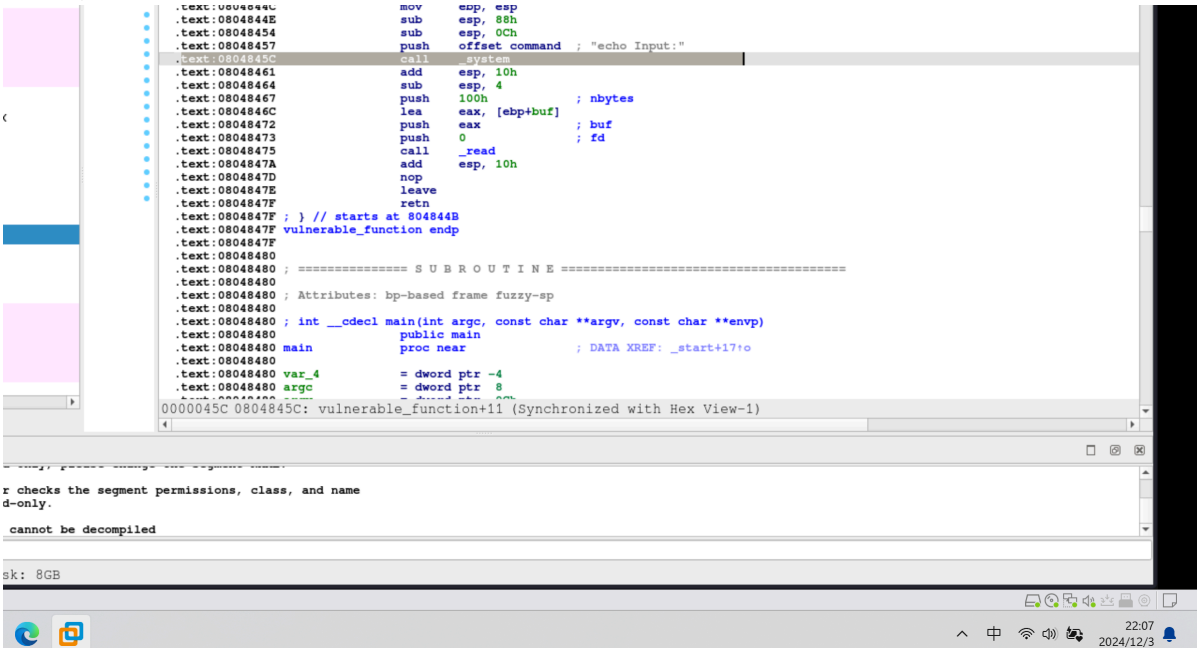


寻找后门

发现/bin/sh

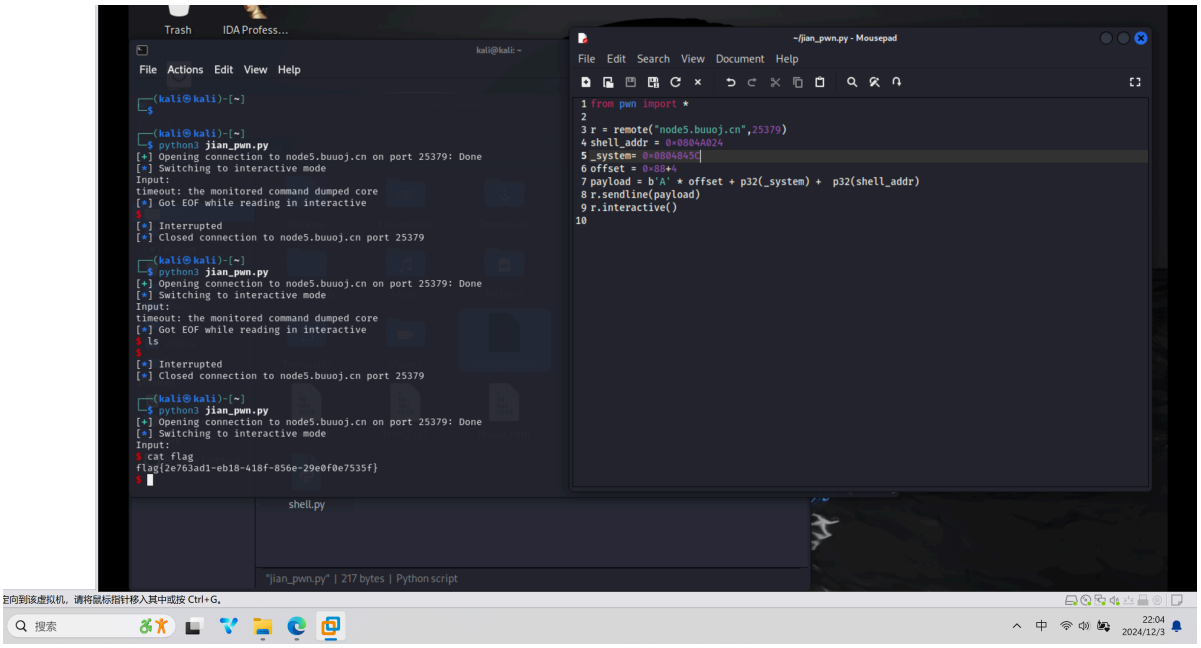


发现_system



写脚本

32位脚本是挺难写的哈



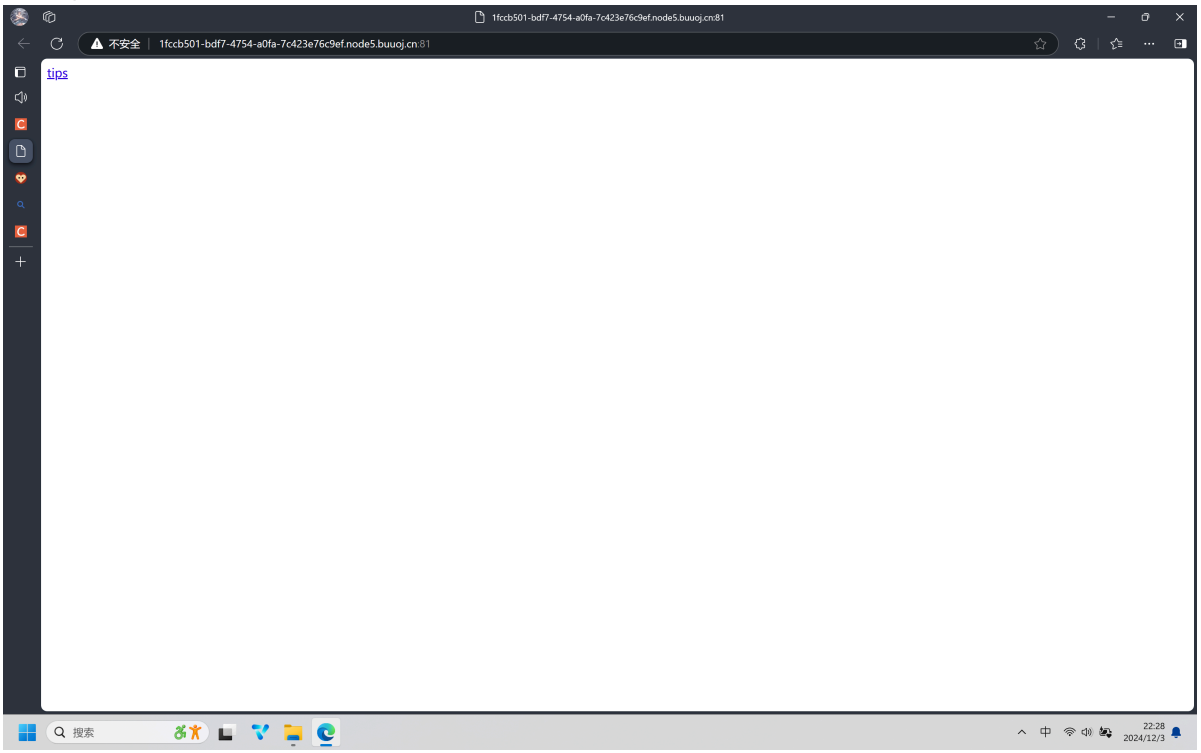
web

[ACTF2020 新生赛]Include

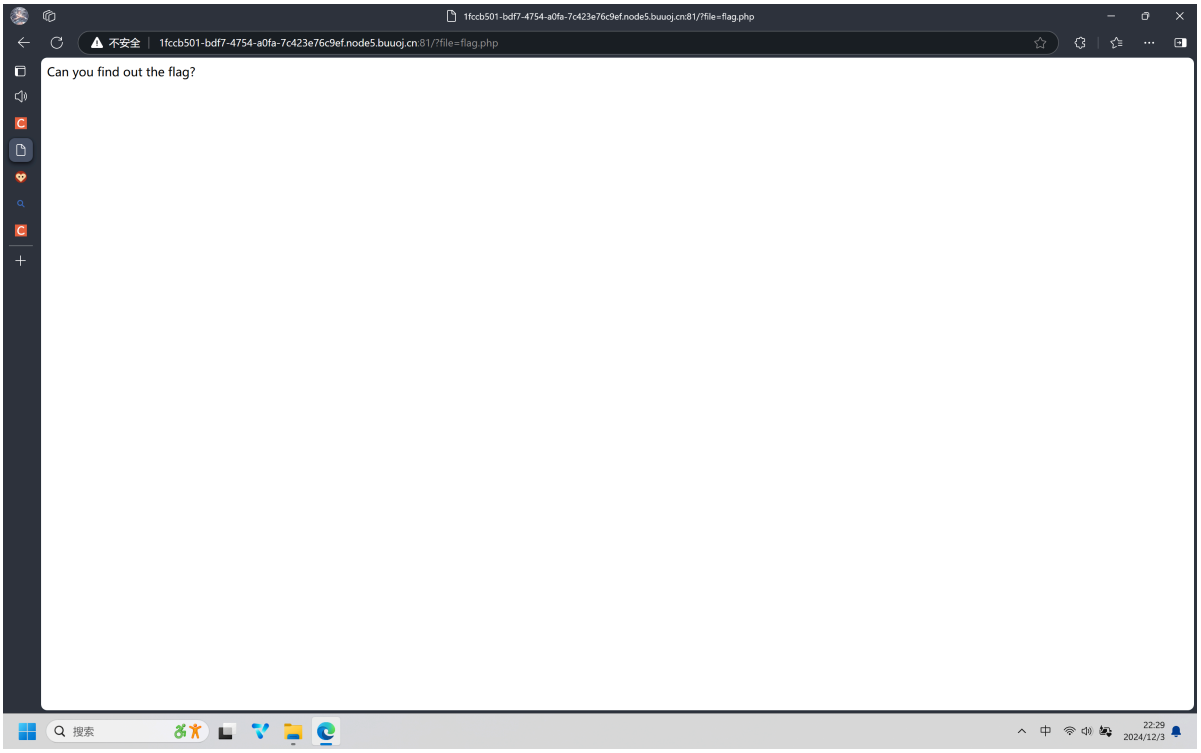
> 做题人:焦昱淇
> 题目url:https://buuoj.cn/challenges#[ACTF2020%20%E6%96%B0%E7%94%9F%E8%B5%9B]Include
> 知识点:PHP封装协议

打开

发现tips，点进去



发现后缀为flag.php，?file=flag.php 猜测文件包含漏洞



看网上说的

重要的知识点——PHP封装协议：

php://filter/read=convert.base64-encode/resource=xxx.php

php://filter 是php中独有的一个协议，可以作为一个中间流来处理其他流，可以进行任意文件的读取；根据名字filter，可以很容易想到这个协议可以用来过滤一些东西； 使用不同的参数可以达到不同的目的和效果：

resource=<要过滤的数据流> 指定了你要筛选过滤的数据流。 必选

read=<读链的筛选列表>可以设定一个或多个过滤器名称，以管道符（|）分隔。 可选

write=<写链的筛选列表> 可以设定一个或多个过滤器名称，以管道符（|）分隔。 可选

<；两个链的筛选列表> 任何没有以 read= 或write=作前缀 的筛选器列表会视情况应用于读或写链。
php://filter与包含函数结合时，php://filter流会被当作php文件执行。所以我们一般对其进行编码，阻止其不执行。从而导致任意文件读取。
read=convert.base64-encode，用base64编码输出，不然会直接当做php代码执行，看不到源代码内容。

于是使用封装协议这个漏洞

base64解码