

# HGAME 2022 Week4 writeup by nerowander

眨眼之间，week4 就已经结束了，期间还是学到了不少东西，希望在接下来的 hgame-final 能不交白卷，学以致用。

## MISC

---

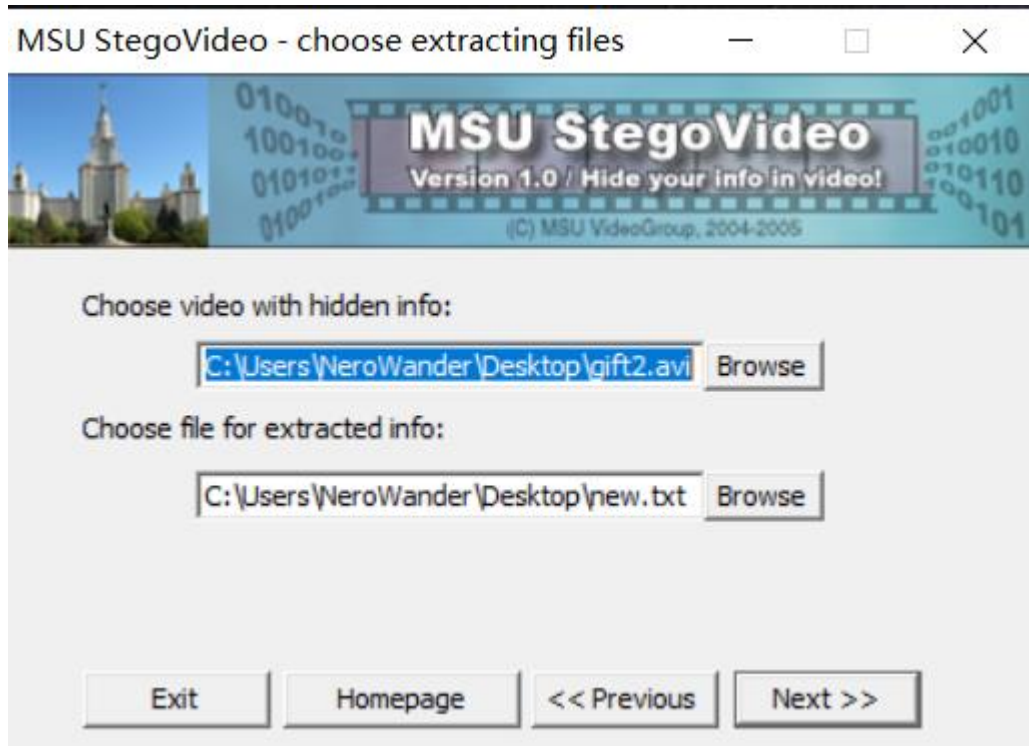
### At0m 的给你们的(迟到的)情人节礼物

首先观察第一个视频，其实它不仅是 go 语言那道题的 hint（虽然 go 语言那道题我没做出来）

可能是学长故意露出的一些细节，让我发现了一些提示



问题在于，想用这个工具提取文件中的密文，需要一个 6 位数字的密码，看了一下 8889 端口，尝试了 888999 密码，然后就成功了



啊这，我也没想到这就猜出来了，还是需要一定的猜测能力的（情人节的浪漫）

new - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

hgame{Q1ng\_R3n\_J1e\_Da\_Sh4\_CTF}

## CRYPTO

### ECC

思路如下，解出 flag，关键是掌握基本的概念和网上的一些解法就可以了

```
task.sage的解 - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
p = 74997021559434065975272431626618720725838473091721936616560359000648651891507
a = 61739043730332859978236469007948666997510544212362386629062032094925353519657
b = 87821782818477817609882526316479721490919815013668096771992360002467657827319
k = 93653874272176107584459982058527081604083871182797816204772644509623271061231
E = EllipticCurve(GF(p),[a,b])
c1 = E(14455613666211899576018835165132438102011988264607146511938249744871964946084,25506582570
c2 = E(37554871162619456709183509122673929636457622251880199235054734523782483869931,71392055540
m=c1-k*c2
print(m)
m=(57824879640955326550732559538097319221644125075532201058220628014917816573008 : 544752758661
c=m[0]
d=m[1]
e=cipher_left//m[0]
f=cipher_right//m[1]
cipher_left = 68208062402162616009217039034331142786282678107650228761709584478779998734710
cipher_right = 27453988545002384546706933590432585006240439443312571008791835203660152890619

hgame(Ecc$!s!sO@HaRd)
```

## PRNG

这道题其实挺有意思的，卡了我半天

首先得读懂原来的 python 代码，可以得出是 flag 转换成数字序列之后以 4 个数字为一个部分（part）和接下来生成的随机数（从第 625 个随机数开始）作异或，异或是可逆的，问题不大，重点是知道接下来的随机数是什么

可以根据已有的 624 个随机数，根据 mt19937 算法生成接下来的随机数，也可以找出这已知的 624 个随机数的 seed（随机数种子）写出来接下来的随机数，不过我根据网上的方法得出了种子，但是没有找到用种子写的方法，所以还是采用了第一种方法

```
from libnum import n2s,s2n

#这个seed不是真正的seed，我只是懒得改变量名字了，真正的seed是，2073475817
seed=1444384326
a=1444397883
b=a^seed
print(a^seed)
print(seed^a)
print(n2s(b))
#hgame(meRseme!tWiTeR~!S^AwIdely~USEd*pSEU0oErAndM:nUnBEr!GeNErAtion?Algor!ThN)

#437104340, 568103176, 1635844121, 878522509, 1923790547, 1727955782, 1371509288, 3182873539, 156878129, 1757777801, 1472886960, 3486450735, 2307527058, 2958814692, 1817
```

根据已有 624 个随机数推接下来的随机数的方法：

<https://googles.plus/2022/01/16/mt19937-mei-sen-xuan-zhuan-suan-fa/#toc-heading-4>

附上思路和随机数的整理

```
output处理 - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
342928858, 3728958896, 1477077966, 1433151407, 1119405037, 330145973, 3547181160, 2123007249, 3739964132, 1794
2802808043, 2418872990, 1043274549, 144911746, 2312236858, 780373658, 1527499811, 3402753408, 2617924770, 1659
388108494, 1110548082, 2357147660, 2336724751, 4047583630, 2108667879, 2784078579, 1170844412, 3920262445, 356
33070, 1306693906, 2968672077, 2476023772, 2645573325, 3939390068, 2874886754, 4226430090, 2290851636, 3707585
, 518207679, 1662309775, 278933232, 4294256390, 2444117793, 2241879973, 3915962245, 3836532482, 3449260219, 109
2403021083, 424891533, 1887765641, 2090726432, 2897940431, 268403838, 3447542890, 575011346, 2559143209, 53264
, 3977867960, 1263177666, 2159508450, 2704509681, 1540819416, 1836499452, 1667451095, 3799477506, 157146600, 37
[3437104340, 508103176, 1635844121, 878522509, 1923790547, 1727955782, 1371509208, 3182873539, 156878129, 17577

一个一个拼起来: hgame(meRsennetWisTER~iS^A*WIDeLY-USEd^pSEUDo&rAndOM:nUmBEr!GeNErATIon?AlgorIThM)}

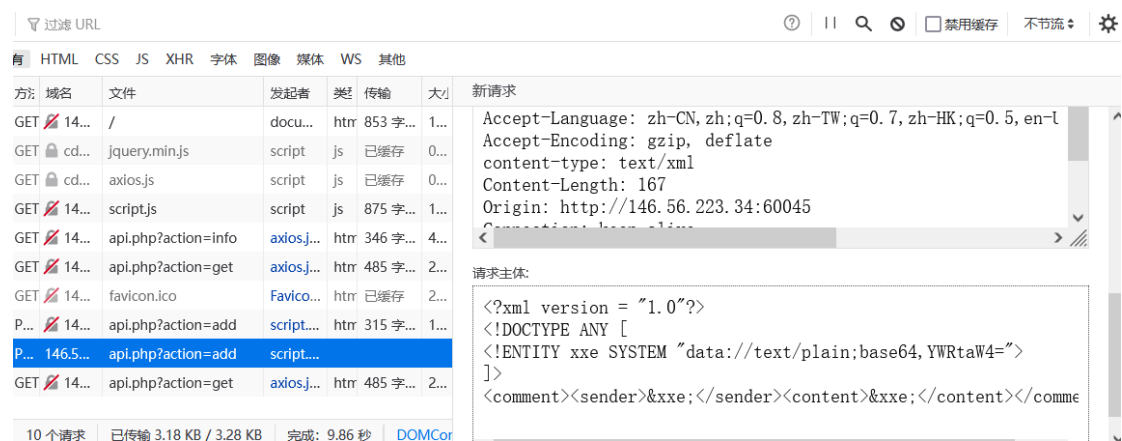
给出mt19937中的10个参数N、M、A、U、S、B、T、C、L、F,
并给出刚刚生成的前N个伪随机数, 求出伪随机数对应的种子seed。
看上去, 这道题正好符合了这种情况, 算了不管了试试
N=624
M=397
A=0x9908b0df(2567483615)
U=11
S=7
B=0x9d2c5680(2636928640)
T=15
C=0xefc60000(4022730752)
L=18
F=0x6c078965(1812433253)
```

```
real - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
888058162
3094055443
1404990361
1012543603
448723884
2580444236
201608779
1062995809
1348787313
2980019361
2245025385
494977308
4042503808
275744301
406611131
142226472
3871761759
3888795536
2617489687
1220227074
342928858
```

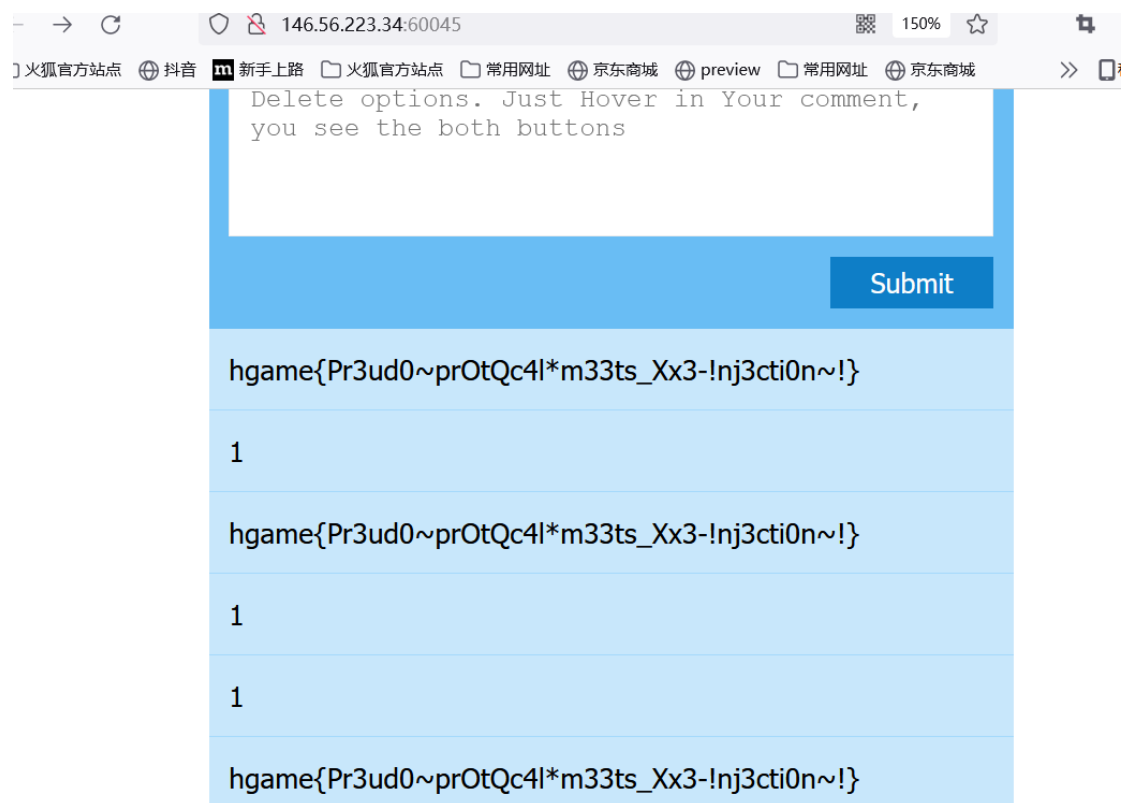
# WEB

## comment

根据 hint, 使用 xml 格式和伪协议, 以及考虑源码过滤的情况, 参考网上的 xxe 漏洞的使用, 以及可以考虑 data 伪协议构造”admin”的其他编码数据绕过检查得到 flag



在一个新的 post 请求主体上修改一下，用 xml 格式，刷新页面得出 flag



事实证明一点：对 web 源码审计的时候，往往需要抓住一部分，因为那一部分就是重点