

Hgame week4 writeup

Hgame week4 writeup

change
crackme
ezKeyboard
总结
附录

change

看了半天，学了ida怎么用

```
PrepareAndResizeContainer((struct std::_Container_base0 *)a3, "am2qas1");// 传入“am2qas1”的地址  
v6 = (__int64 *)std::shared_ptr<__ExceptionPtr>::operator=((__int64)a0, a3);  
AssignAndDestroyString((__int64)a2, v6);  
sub_7FF66D741410(std::cout, "plz input your flag:");  
sub_7FF66D7410F0(std::cin, (__int64)flag); // 输入到unk—128地址  
EncryptWithAlternatingAlgorithms(a2, a1, flag);// 进行特殊的运算，结果再v8中
```

// 基于a1 和 a2 指向的内存区域的内容，开始加密

```
__int64 __fastcall EncryptWithAlternatingAlgorithms(__int64 *a2, __int64 *a1, __int64  
{  
    char *v3; // rax  
    char v4; // a1 /v4是v3的低八位  
    char *v5; // rax  
    int i; // [rsp+20h] [rbp-58h]  
    unsigned int d; // [rsp+28h] [rbp-50h]  
    unsigned int Duration; // [rsp+30h] [rbp-48h]  
    unsigned __int64 v10; // [rsp+48h] [rbp-30h]  
    unsigned __int64 ValueAtOffset16; // [rsp+58h] [rbp-20h]  
  
    std::shared_ptr<__ExceptionPtr>::operator=((__int64)a1, flag);  
    for ( i = 0; i < (unsigned __int64)RetrieveValueAtOffset16((__int64)a1); ++i )  
    {  
        if ( i % 2 )  
        {  
            change_a1((__int64)xor); // 将a1改为xor的函数指针  
            ValueAtOffset16 = RetrieveValueAtOffset16((__int64)a2);  
            Duration = *(char *)add(a2, i % ValueAtOffset16);  
            v5 = (char *)add(a1, i);  
            a1_(*v5, Duration);  
        }  
        else  
        {  
            change_a1((__int64)xor_add_10); // 将a1改为xor_add_10的函数指针  
            v10 = RetrieveValueAtOffset16((__int64)a2);  
            d = *(char *)add(a2, i % v10);  
            v3 = (char *)add(a1, i);  
            a1_(*v3, d);  
        }  
        *(_BYTE *)add(a1, i) = v4;  
    }  
    return a1;  
}
```

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

```
int main() {  
    int enc[] = {0x13, 0x0A, 0x5D, 0x1C, 0x0E, 0x08, 0x23, 0x06, 0x0B, 0x4B, 0x38,  
0x22, 0x0D, 0x1C, 0x48, 0x0C, 0x66, 0x15, 0x48, 0x1B, 0x0D, 0x0E, 0x10, 0x4F,  
0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00};
```

```

    int a1[] = {0x13, 0x0A, 0x5D, 0x1C, 0x0E, 0x08, 0x23, 0x06, 0x0B, 0x4B, 0x38,
0x22, 0x0D, 0x1C, 0x48, 0x0C, 0x66, 0x15, 0x48, 0x1B, 0x0D, 0x0E, 0x10, 0x4F,
0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00};
    int a2[] = {0x61, 0x6D, 0x32, 0x71, 0x61, 0x73, 0x6C, 0, 0, 0, 0, 0, 0, 0, 0,
0, 7, 0, 0, 0, 0, 0, 0, 0, 0, 0x0F, 0, 0, 0, 0, 0, 0, 0};
    for (int i = 0; i < 0x18; i++) {
        if (i % 2) {
            a1[i] ^= a2[i % 7];
        } else {
            a1[i] -= 10;
            a1[i] ^= a2[i % 7];
        }
    }
    for (int i = 0; i < 24; i++) {
        printf("%c", a1[i]);
    }
}

```

crackme

这段代码先用ida简单逆一下

发现里面加密程序有点像base64 (也是现查的)

```

#include <stdio.h>
#include <string.h>

char v18[] = "TXqSL5eBqXGW1I4bfe4s05FQJYNgfXq4GShRUQs7tb9OgF3xkWCvmIYlvPaBo4pj";

char v17[] = "wrDi8EXKnZAczfdy2M/+ue5J0NTL1q6HGShRUQs7tb9OgF3xkWCvmIYlvPaBo4pj";
// char base[] = "wrDi8EXKnZAczfdy2M/+ue5J0NTL1q6H";
char flag[50];
int search(char a) {
    for (int i = 0; i < strlen(v17); i++) {
        if (a == v17[i])
            return i;
    }
}

int main() {
    int i;
    for (i = 0; i < strlen(v18) / 4; i++) {
        flag[i * 3] = (search(v18[0 + 4 * i]) << 2) + (search(v18[1 + 4 * i]) >>
4);
        flag[i * 3 + 1] = (search(v18[2 + 4 * i]) >> 2) + ((search(v18[1 + 4 * i]) &
0xf) << 4);
        flag[i * 3 + 2] = ((search(v18[2 + 4 * i]) & 0x3) << 6) + search(v18[3]);
    }
    printf("%d\n", i);

    for (i = 0; i < 16 * 3; i++) {
        printf("%c", flag[i]);
    }
}

```

出flag了

```
hgameath!s_a5_aaka_fa4gaY[?]展
```

owo, 好像是个假的。

发现MEMORY[0]好像触发了异常处理里面调用了VirtualProtect, 然后就做不出了

```
int __fastcall main(int argc, const char **argv, const char **envp)
{
    int v3; // eax
    const char *v4; // rcx
    char v6[72]; // [rsp+30h] [rbp-48h] BYREF

    sub_7FF7AD6035C4("%50s", v6);
    MEMORY[0] = 1;
    v3 = encry((__int64)v6);
    v4 = "right flag!";
    if ( !v3 )
        v4 = "wrong flag!";
    puts(v4);
    return 0;
}
```

发现hint: SMC (代码动态加密), 查了解决方案有二, 一种是dump动调时的内存, 另一种时ida的脚本

这里选ida脚本 (dump做题时不知道, 现在也不会)

下面这段脚本有个问题就是定义变量时没写全应该是 0x0000xxxxxxxx, 然后卡了一个中午

```
# 导入必要的模块
import ida_bytes
import idc

# 定义变量
encry_addr = 0x7FF6F0D1105C
des_addr = 0x7FF6F0D16000
loop_count = 0x246A

# 循环进行异或操作
for i in range(loop_count):
    # 获取des_addr+i处的一个字节
    des_byte = ida_bytes.get_byte(des_addr + i)
    # 获取encry_addr+i处的一个字节
    encry_byte = ida_bytes.get_byte(encry_addr + i)
    # 执行异或操作
    xor_result = encry_byte ^ des_byte
    # 将异或后的结果写回encry_addr+i处
    ida_bytes.patch_byte(encry_addr + i, xor_result)
```

出来的差不多了, 正好我发现一个网站就是这个,

[Self-Modified Code - CTF Wiki \(ctf-wiki.org\)](http://ctf-wiki.org/)

划水的时候发现了[z3 - CTF Wiki](http://z3.ctf-wiki.org/)这个工具, 有点像数学建模时遇到的线性规划

还发现了[Hgame Ctf write up \(3&4\) - 『脱壳破解区』 - 吾爱破解 - LCG - LSG | 安卓破解 | 病毒分析 | www.52pojie.cn](http://www.52pojie.cn/)

详细代码请见附录，这里改变量用了半天

应该是本周开始做题的第一题，不爆0

```
usb.device_address == 2 && usbhid.data && usb.src == 1.2.3
```

```
tshark -r t2.pcap -T fields -e usb.capdata > t2.txt
```

发现有错误，大小写不对，应该是对modified key 的hid编码识别不对

[illegible]

总结

pwn, 密码还没看懂

附录

```

x.add(177 * flag[7] + 129 * flag[20] + 117 * flag[22] + 143 * flag[1] + 65 *
flag[28] + 137 * flag[27] + 215 * flag[19] + 93 * flag[13] + 235 * flag[17] + 203
* flag[2] + 15 * (flag[8] + 17 * flag[14])) + 2 * (flag[4] + 91 * flag[12] + 95 *
flag[10] + 51 * flag[5] + 81 * flag[15] + 92 * flag[9] + 112 * (flag[3] +
flag[23])) + 32 * (flag[6] + 2 * (flag[25] + flag[16])) + 6 * (flag[21] + 14 *
flag[24] + 19 * flag[18]) + 83 * flag[0] + 53 * flag[29] + 123 * flag[30]) +
flag[26] + 175 * flag[11] + 183 * flag[31] == 448573)
x.add(113 * flag[30] + 74 * flag[31] + 238 * flag[23] + 140 * flag[21] + 214 *
flag[20] + 242 * flag[28] + 160 * flag[19] + 136 * flag[16] + 209 * flag[12] +
220 * flag[13] + 50 * flag[4] + 125 * flag[3] + 175 * flag[15] + 23 * flag[17] +
137 * flag[6] + 149 * flag[9] + 83 * (flag[29] + 2 * flag[14]) + 21 * (9 *
flag[10] + flag[24]) + 59 * (4 * flag[11] + flag[26]) + 41 * (flag[25] + flag[5])
+ 13 * (flag[8] + 11 * (flag[7] + flag[18])) + 6 * flag[22] + 4 * (flag[1] + 2 *
flag[2]) + flag[1] + 2 * flag[2] + 17 * flag[0]) + 36 * flag[27] == 384306)
x.add(229 * flag[19] + 78 * flag[25] + flag[21] + flag[12] + 133 * flag[11] + 74
* flag[23] + 69 * flag[20] + 243 * flag[8] + 98 * flag[1] + 253 * flag[28] + 142
* flag[27] + 175 * flag[13] + 105 * flag[5] + 221 * flag[3] + 121 * flag[17] +
218 * (flag[30] + flag[10]) + 199 * (flag[4] + flag[14]) + 33 * (flag[7] + 7 *
flag[26]) + 4 * (27 * flag[15] + 50 * flag[2] + 45 * flag[9] + 19 * (flag[31] +
flag[22]) + flag[24] + 16 * flag[16] + 52 * flag[29]) + 195 * flag[6] + 211 *
flag[0] + 153 * flag[18] == 424240)
x.add(181 * flag[27] + 61 * flag[21] + 65 * flag[19] + 58 * flag[13] + 170 *
flag[10] + 143 * flag[4] + 185 * flag[3] + 86 * flag[2] + 97 * flag[6] + 235 *
(flag[16] + flag[11])) + 3 * (53 * flag[5] + 74 * (flag[28] + flag[31])) + 13 *
(flag[22] + 6 * flag[12]) + 11 * (flag[17] + 7 * flag[15]) + 15 * (flag[9] + 4 *
flag[26]) + flag[8] + 35 * flag[25] + 29 * flag[18]) + 4 * (57 * flag[23] + 18 *
(flag[0] + (2 * flag[20]))) + flag[1] + 17 * flag[24] + 55 * flag[14]) + 151 *
flag[7] + 230 * flag[29] + 197 * flag[30] == 421974)
x.add((209 * flag[19] + 249 * flag[14] + 195 * flag[21] + 219 * flag[27] + 201 *
flag[17] + 85 * flag[9] + 213 * (flag[26] + flag[13])) + 119 * (flag[2] + 2 *
flag[5]) + 29 * (8 * flag[4] + flag[7] + 4 * flag[11] + flag[11]) + 2 * (flag[28]
+ 55 * (2 * flag[10] + flag[30])) + 3 * (flag[3] + 39 * flag[12] + 2 * (flag[23] +
20 * flag[15])) + 35 * flag[8]) + 4 * (flag[0] + 31 * flag[22] + 28 * flag[31]) +
26 * flag[1] + 46 * ((2 * flag[20]) + flag[24]) + 98 * flag[25]) + 53 * flag[16]
+ 171 * flag[18] + 123 * flag[29] == 442074))
x.add((162 * flag[30] + 74 * flag[0] + 28 * flag[11] + 243 * flag[22] + 123 *
flag[1] + 73 * flag[28] + 166 * flag[16] + 94 * flag[4] + 113 * flag[2] + 193 *
flag[6] + 122 * (flag[23] + 2 * flag[8])) + 211 * (flag[3] + flag[27])) + 21 *
(flag[26] + 7 * flag[5]) + 11 * (flag[29] + 23 * (flag[24] + flag[17])) + 2 *
(flag[7] + 5 * flag[14] + 2 * (2 * flag[9] + flag[10])) + 2 * flag[9] + flag[10]))
+ 5 * (46 * flag[12] + 26 * flag[15] + 4 * (flag[13] + 2 * flag[19]) + flag[18] +
27 * flag[21] + 10 * flag[25]) + 36 * (flag[31] + 5 * flag[20]) == 376007))
x.add((63 * flag[30] + 143 * flag[0] + 250 * flag[23] + 136 * flag[21] + 214 *
flag[7] + 62 * flag[20] + 221 * flag[22] + 226 * flag[8] + 171 * flag[1] + 178 *
flag[28] + 244 * flag[16] + (flag[12] * 128) + 150 * flag[13] + 109 * flag[10] +
70 * flag[5] + 127 * flag[15] + 204 * flag[17] + 121 * flag[6] + 173 * flag[9] +
69 * (flag[27] + flag[14] + flag[11])) + 74 * (flag[24] + 2 * flag[18] + flag[18])
+ 22 * (7 * flag[4] + flag[26] + 10 * flag[2]) + 40 * (flag[25] + 4 * flag[19] +
flag[19]) + 81 * flag[3] + 94 * flag[29] + 84 * flag[31] == 411252))

```

```

x.add(229 * flag[18] + 121 * flag[29] + 28 * flag[14] + 206 * flag[24] + 145 *
flag[11] + 41 * flag[25] + 247 * flag[23] + 118 * flag[20] + 241 * flag[1] + 79 *
flag[28] + 102 * flag[27] + 124 * flag[16] + 65 * flag[12] + 68 * flag[13] + 239
* flag[26] + 148 * flag[4] + 245 * flag[17] + 115 * flag[2] + 163 * flag[6] + 137
* flag[9] + 53 * (flag[0] + 2 * flag[10])) + 126 * (flag[7] + 2 * flag[3]) + 38 *
(flag[8] + flag[19] + 4 * flag[8] + 6 * flag[5]) + 12 * (flag[21] + 16 *
flag[22]) + 109 * flag[15] + 232 * flag[31] + 47 * flag[30] == 435012)
x.add(209 * flag[19] + 233 * flag[7] + 93 * flag[25] + 241 * flag[21] + 137 *
flag[28] + 249 * flag[26] + 188 * flag[10] + 86 * flag[4] + 246 * flag[3] + 149 *
flag[15] + 99 * flag[2] + 37 * flag[6] + 219 * flag[9] + 17 * (flag[23] + 10 *
flag[27])) + 49 * (flag[0] + 3 * flag[31] + 4 * flag[1] + flag[1]) + 5 * (16 *
flag[17] + 11 * (flag[5] + 2 * flag[11] + flag[11])) + 12 * flag[8] + flag[13] +
30 * flag[24] + 27 * flag[30]) + 18 * (flag[16] + 2 * (flag[29] + flag[20] + 2 *
flag[29])) + flag[29] + flag[20] + 2 * flag[29]) + 24 * flag[12] + 109 * flag[22]
+ 183 * flag[14] + 154 * flag[18] == 392484)
x.add((155 * flag[18] + 247 * flag[7] + 157 * flag[1] + 119 * flag[16] + 161 *
flag[26] + 133 * flag[15] + 85 * flag[6] + 229 * (flag[8] + flag[4]) + 123 * (2 *
flag[13] + flag[22])) + 21 * (flag[5] + 12 * flag[14]) + 55 * (flag[12] + flag[0]
+ flag[9] + 2 * flag[0])) + 15 * (flag[31] + 16 * flag[3] + 9 * flag[19])) + 2 *
(flag[21] + 115 * flag[10] + 111 * flag[24] + 26 * flag[23] + 88 * flag[28] + 73
* flag[17] + 71 * flag[2] + 28 * (flag[20] + 2 * (flag[27] + 2 * flag[25]))) + 51
* flag[11] + 99 * flag[29] + 125 * flag[30]) == 437910))
x.add(220 * flag[31] + 200 * flag[29] + 139 * flag[18] + 33 * flag[0] + 212 *
flag[14] + 191 * flag[24] + 30 * flag[11] + 233 * flag[25] + 246 * flag[23] + 89
* flag[21] + 252 * flag[7] + 223 * flag[22] + 19 * flag[27] + 141 * flag[19] +
163 * flag[12] + 185 * flag[26] + 136 * flag[13] + 46 * flag[4] + 109 * flag[3] +
217 * flag[17] + 75 * flag[6] + 157 * flag[9] + 125 * (flag[2] + flag[30]) + 104
* ((2 * flag[5]) + flag[15]) + 43 * (flag[1] + 2 * flag[10] + flag[10])) + 32 *
(flag[28] + flag[8] + 2 * flag[28] + 2 * (flag[16] + flag[20])) == 421905)
x.add(211 * flag[4] + 63 * flag[18] + 176 * flag[0] + 169 * flag[24] + 129 *
flag[11] + 146 * flag[7] + 111 * flag[20] + 68 * flag[22] + 39 * flag[27] + 188 *
flag[16] + 130 * flag[12] + (flag[13] * 64) + 91 * flag[5] + 208 * flag[15] + 145
* flag[17] + 247 * flag[9] + 93 * (flag[6] + flag[26]) + 71 * (flag[23] + 2 *
flag[2])) + 103 * (flag[28] + 2 * flag[14]) + 6 * (flag[19] + 10 * flag[1] + 28 *
flag[8] + 9 * flag[10] + 19 * flag[21] + 24 * flag[25] + 22 * flag[31]) + 81 *
flag[3] + 70 * flag[29] + 23 * flag[30] == 356282)
x.add((94 * flag[22] + 101 * flag[21] + 152 * flag[7] + 200 * flag[8] + 226 *
flag[28] + 211 * flag[16] + 121 * flag[4] + 74 * flag[2] + 166 * flag[9] +
((flag[23] + 3 * flag[1]) * 64) + 41 * (4 * flag[12] + flag[19])) + 23 * (flag[17]
+ 11 * flag[5])) + 7 * (flag[15] + 10 * flag[27] + 2 * (flag[3] + 2 * (flag[13] +
4 * (flag[10] + flag[26]))) + flag[13] + 4 * (flag[10] + flag[26])) + (flag[3] + 2
* (flag[13] + 4 * (flag[10] + flag[26]))) + flag[13] + 4 * (flag[10] + flag[26]))))
+ 3 * (78 * flag[14] + 81 * flag[24] + 55 * flag[11] + 73 * flag[25] + 4 *
flag[20] + flag[18] + 85 * flag[31] + 65 * flag[30]) + 62 * flag[6] + 88 *
flag[0] + 110 * flag[29] == 423091))
x.add(133 * flag[6] + 175 * flag[18] + 181 * flag[14] + 199 * flag[24] + 123 *
flag[11] + 242 * flag[25] + 75 * flag[23] + 69 * flag[21] + 153 * flag[7] + 33 *
flag[20] + 100 * flag[22] + 229 * flag[8] + 177 * flag[28] + 134 * flag[13] + 179
* flag[10] + 129 * flag[5] + 14 * flag[3] + 247 * flag[4] + 228 * flag[15] + 92 *
flag[2] + 86 * (flag[12] + (2 * flag[9]))) + 94 * (flag[16] + flag[19]) + 37 *
(flag[26] + 4 * flag[31]) + 79 * (flag[27] + 2 * flag[1]) + 72 * flag[0] + 93 *
flag[17] + 152 * flag[29] + 214 * flag[30] == 391869)

```

```

x.add(211 * flag[4] + 213 * flag[9] + 197 * flag[7] + 159 * flag[27] + 117 *
flag[19] + 119 * flag[12] + 98 * flag[26] + 218 * flag[5] + 106 * flag[17] + 69 *
flag[2] + 43 * (flag[21] + flag[10] + 2 * flag[21])) + 116 * (flag[29] + flag[3] +
(2 * flag[20])) + 5 * (flag[22] + 9 * flag[16] + 35 * flag[15] + 37 * flag[13]) +
11 * (flag[24] + 13 * flag[11] + 5 * flag[0] + 8 * flag[14]) + 6 * (29 * flag[1]
+ 25 * flag[28] + 38 * flag[6] + flag[18] + 13 * flag[25] + 10 * flag[31]) + 136
* flag[8] + 142 * flag[23] + 141 * flag[30] == 376566)
x.add(173 * flag[31] + 109 * flag[18] + 61 * flag[14] + 187 * flag[25] + 79 *
flag[23] + 53 * flag[7] + 184 * flag[19] + 43 * flag[16] + 41 * flag[12] + 166 *
flag[13] + 193 * flag[5] + 58 * flag[4] + 146 * flag[3] + (flag[15] * 64) + 89 *
flag[17] + 121 * flag[2] + 5 * (flag[26] + 23 * flag[28])) + 7 * (29 * flag[9] +
flag[10] + 4 * flag[8]) + 13 * (3 * flag[22] + flag[24] + 7 * flag[20] + 13 *
flag[21]) + 3 * (flag[29] + 83 * flag[0] + 51 * flag[11] + 33 * flag[6] + 8 *
(flag[30] + 4 * flag[1]) + 18 * flag[27]) == 300934)
x.add((78 * flag[25] + 131 * flag[0] + 185 * flag[24] + 250 * flag[7] + 90 *
flag[20] + 129 * flag[22] + 255 * flag[1] + 206 * flag[28] + 239 * flag[27] + 150
* flag[3] + 253 * flag[17] + 104 * flag[6] + 58 * (flag[21] + 2 * flag[8])) + 96 *
(flag[18] + flag[13]) + 117 * (flag[12] + 2 * flag[29]) + 27 * (flag[26] + 8 *
flag[9] + flag[9]) + 19 * (flag[16] + 3 * flag[19] + 4 * flag[10] + flag[10]) + 7
* (22 * flag[5] + 3 * (flag[2] + 11 * flag[4])) + flag[31] + 29 * flag[23] + 14 *
flag[11]) + 109 * flag[15] + 102 * flag[14] + 100 * flag[30] == 401351))
x.add(233 * flag[30] + 71 * flag[0] + 209 * flag[11] + 82 * flag[23] + 58 *
flag[20] + 53 * flag[27] + 113 * flag[16] + 206 * flag[13] + 39 * flag[5] + 163 *
flag[15] + 222 * flag[2] + 191 * flag[9] + 123 * (flag[8] + flag[7]) + 69 *
(flag[12] + 2 * flag[6] + flag[6]) + 9 * (flag[31] + 8 * flag[4] + 7 * (3 *
flag[25] + flag[1])) + 5 * flag[24] + 19 * flag[14]) + 4 * (flag[18] + 26 *
flag[26] + 61 * flag[10] + 43 * flag[22] + 49 * flag[21] + 32 * flag[29]) + 10 *
(7 * (flag[28] + (3 * flag[19]))) + flag[17] + 12 * flag[3]) == 368427)
x.add(139 * flag[14] + 53 * flag[0] + 158 * flag[24] + 225 * flag[25] + 119 *
flag[23] + 67 * flag[21] + 213 * flag[7] + 188 * flag[1] + 152 * flag[28] + 187 *
flag[19] + 129 * flag[16] + 54 * flag[12] + 125 * flag[26] + 170 * flag[4] + 184
* flag[2] + 226 * flag[6] + 253 * flag[9] + 26 * (flag[10] + flag[5]) + 97 *
(flag[29] + 2 * flag[27]) + 39 * (5 * flag[20] + flag[11]) + 21 * (flag[17] + 8 *
flag[22]) + 12 * (17 * flag[3] + flag[13] + 15 * flag[8] + 12 * flag[30]) + 165 *
flag[15] + 88 * flag[18] + 157 * flag[31] == 403881)
x.add(114 * flag[31] + 61 * flag[11] + 134 * flag[7] + 62 * flag[22] + 89 *
flag[12] + 211 * flag[26] + 163 * flag[5] + 66 * flag[4] + 201 * (flag[8] +
flag[9]) + 47 * (5 * flag[24] + flag[6]) + 74 * (flag[29] + flag[13]) + 142 *
(flag[21] + flag[1]) + 35 * (flag[15] + 6 * flag[20]) + 39 * (flag[18] + 6 *
flag[14]) + 27 * (flag[27] + 9 * flag[16] + 8 * flag[23]) + 4 * (flag[19] + 63 *
flag[30] + 2 * (flag[25] + 12 * (flag[3] + flag[0])) + 8 * flag[2] + 26 *
flag[10])) + 10 * (flag[28] + 4 * flag[17] + flag[17]) == 382979)
x.add(122 * flag[27] + 225 * flag[19] + 52 * flag[16] + 253 * flag[12] + 197 *
flag[26] + 187 * flag[13] + 181 * flag[10] + 183 * flag[5] + 47 * flag[15] + 229
* flag[17] + 88 * flag[6] + 127 * (flag[3] + (2 * flag[9])) + 37 * (flag[8] + 3 *
flag[31]) + ((flag[2] + 2 * flag[14] + flag[14]) * 64) + 7 * (21 * flag[28] +
flag[11] + 18 * (flag[29] + flag[25] + (2 * flag[24]))) + 6 * (23 * flag[4] +
flag[20] + 17 * flag[21] + 39 * flag[23]) + 10 * (flag[0] + 11 * flag[1] + 21 *
flag[22]) + 149 * flag[30] + 165 * flag[7] + 121 * flag[18] == 435695)
x.add(165 * flag[15] + 223 * flag[29] + 249 * flag[0] + 199 * flag[25] + 135 *
flag[21] + 133 * flag[20] + 254 * flag[22] + 111 * flag[8] + 189 * flag[1] + 221
* flag[27] + 115 * flag[19] + 186 * flag[12] + 79 * flag[5] + 217 * flag[4] + 122
* flag[2] + 38 * flag[9] + 109 * ((2 * flag[13]) + flag[10]) + 14 * (flag[28] +
17 * flag[7] + 8 * (flag[23] + (2 * flag[24]))) + 4 * (11 * (5 * flag[14] +
flag[17]) + 6 * (flag[3] + 2 * flag[6]) + flag[11] + 52 * flag[26] + 50 *
flag[16]) + 229 * flag[18] + 86 * flag[31] + 234 * flag[30] == 453748)

```



```

x.add(181 * flag[27] + 94 * flag[22] + 125 * flag[25] + 226 * flag[20] + 155 *
flag[8] + 95 * flag[19] + 212 * flag[26] + 91 * flag[13] + 194 * flag[10] + 98 *
flag[4] + 166 * flag[2] + 120 * flag[6] + 59 * flag[9] + 32 * (flag[12] +
flag[28])) + 158 * (flag[23] + flag[0]) + 101 * (flag[5] + flag[30]) + 63 *
(flag[29] + 2 * flag[16]) + 67 * (flag[1] + 2 * flag[15]) + 11 * (flag[17] + 10 *
flag[24] + 11 * flag[3]) + 39 * (flag[14] + 4 * (flag[21] + flag[18])) + 233 *
flag[7] + 56 * flag[11] + 225 * flag[31] == 358321)
x.add(229 * flag[19] + 135 * flag[29] + 197 * flag[18] + 118 * flag[0] + 143 *
flag[24] + 134 * flag[23] + 204 * flag[7] + 173 * flag[20] + 81 * flag[8] + 60 *
flag[1] + 58 * flag[28] + 179 * flag[16] + 142 * flag[12] + 178 * flag[26] + 230
* flag[13] + 148 * flag[10] + 224 * flag[5] + 194 * flag[4] + 223 * flag[3] + 87
* flag[15] + 200 * flag[17] + 233 * flag[2] + 49 * flag[6] + 127 * (flag[27] +
flag[14]) + 31 * (4 * flag[11] + flag[9]) + 42 * (flag[25] + 6 * flag[21]) + 109
* flag[22] + 75 * flag[31] + 165 * flag[30] == 456073)
x.add(41 * flag[29] + 253 * flag[31] + 163 * flag[18] + 193 * flag[14] + 155 *
flag[24] + 113 * flag[11] + 131 * flag[23] + 55 * flag[21] + 21 * flag[7] + 53 *
flag[20] + 13 * flag[28] + 201 * flag[27] + 237 * flag[12] + 223 * flag[13] + 95
* flag[4] + 194 * flag[15] + 62 * flag[17] + 119 * flag[2] + 171 * flag[6] + 135
* flag[9] + 69 * (flag[3] + 3 * flag[1]) + 211 * (flag[25] + flag[10]) + 4 * (43
* flag[8] + flag[22] + 40 * flag[26]) + 6 * (flag[0] + 33 * flag[5] + 20 * (2 *
flag[30] + flag[19]) + 24 * flag[16]) == 407135)
x.add((111 * flag[30] + 190 * flag[31] + 149 * flag[29] + 173 * flag[1] + 118 *
flag[16] + 146 * flag[10] + 179 * flag[3] + 51 * flag[15] + 49 * flag[17] + 61 *
flag[2] + 125 * flag[6] + 162 * flag[9] + 214 * (flag[27] + flag[14]) + 14 * ((2
* flag[13]) + flag[4]) + 178 * (flag[5] + flag[24]) + 11 * (4 * flag[12] +
flag[19] + 17 * flag[22]) + 65 * (flag[20] + flag[26] + 2 * flag[20] + 2 *
flag[0]) + 4 * (flag[8] + 38 * flag[18] + 4 * (flag[23] + flag[25] + 8 * flag[23]
+ 4 * (flag[28] + 2 * flag[11])) + (flag[23] + flag[25] + 8 * flag[23] + 4 *
(flag[28] + 2 * flag[11])) + 8 * flag[7] + 43 * flag[21]) == 369835))
x.add(27 * flag[11] + 223 * flag[23] + 147 * flag[20] + 13 * flag[19] + 35 *
(flag[26] + 7 * flag[29]) + 57 * (flag[30] + (2 * flag[9]) + 3 * flag[2]) + 11 *
(flag[25] + 17 * (flag[12] + flag[0]) + 10 * flag[24] + 3 * flag[13]) + 2 * (53 *
flag[16] + flag[27] + 38 * flag[18] + 43 * flag[22] + 115 * flag[10] + 61 *
flag[6] + 111 * (flag[3] + flag[7]) + 14 * (flag[15] + flag[8] + 2 * flag[8] + 8
* flag[1]) + 109 * flag[21] + 100 * flag[5] + 63 * flag[28]) + 93 * flag[17] +
251 * flag[14] + 131 * flag[31] == 393303)
x.add(116 * flag[12] + 152 * flag[10] + 235 * flag[15] + 202 * flag[9] + 85 *
(flag[28] + 3 * flag[2]) + 221 * (flag[24] + flag[7]) + 125 * ((2 * flag[5]) +
flag[4]) + 7 * (19 * flag[29] + 9 * (flag[3] + 2 * flag[27]) + flag[21] + 33 *
flag[31] + 32 * flag[30]) + 3 * (71 * flag[17] + 43 * flag[6] + 32 * (flag[26] +
flag[20]) + 15 * (flag[0] + flag[23] + 2 * flag[16]) + flag[1] + 74 * flag[13] +
48 * flag[22]) + 10 * (flag[19] + 11 * flag[14] + 16 * flag[18]) + 136 * flag[8]
+ 106 * flag[25] + 41 * flag[11] == 403661)
x.add(127 * flag[29] + 106 * flag[18] + 182 * flag[14] + 142 * flag[0] + 159 *
flag[24] + 17 * flag[25] + 211 * flag[23] + 134 * flag[21] + 199 * flag[8] + 103
* flag[1] + 247 * flag[16] + 122 * flag[12] + 95 * flag[5] + 62 * flag[3] + 203 *
flag[17] + 16 * flag[2] + 41 * (6 * flag[22] + flag[27]) + 9 * (22 * flag[4] +
flag[15] + 27 * flag[13] + 28 * flag[7]) + 10 * (flag[28] + flag[6] + (3 *
flag[19]) + 8 * flag[26] + 2 * (flag[6] + (3 * flag[19]) + 8 * flag[26]) + 13 *
flag[10]) + 6 * (23 * flag[11] + flag[20]) + 213 * flag[9] + 179 * flag[31] + 43
* flag[30] == 418596)

print(x.check())
print(x.model())
# print(v)

```

