Baby format

Check file

babyformat: ELF 32-bit LSB shared object, Intel 80386, version 1 (SYSV), dynamically linked, interpreter /lib/ld-linux.so.2, for GNU/Linux 2.6.32, BuildID[sha1]=873a9025e110b59cc06c1b6eda1ebeacd89efc28, not stripped

checksec

Arch: i386-32-little

RELRO: Full RELRO

Stack: No canary found

NX: NX enabled

PIE: PIE enabled

PE với NX enable rồi :3 sẽ khó nhằn đây

0x000008fe <+47>: call 0x88e <exploit\_me>

Hàm main gọi hàm exploit\_me

Hàm exploit\_me đọc 13 bytes từ bàn phím sau đó in ra

0x565558ab <+29>: push 0xd

0x565558ad <+31>: push 0x5655702c

0x565558b2 <+36>: push 0x0

0x565558b4 <+38>: call 0xf7ed5b00 <read>

0x565558b9 <+43>: add esp,0x10

0x565558bc <+46>: sub esp,0xc

0x565558bf <+49>: push 0x5655702c

0x565558c4 <+54>: call 0xf7e49670 <\_\_printf>

Debug thì thấy printf ko có format string nên bài này đúng tên của nó là babyformat rồi

Sau khi in xong thì nó kiểm tra xem input đầu vào có phải là Exit không true thì văng ra ko thì chạy tiếp

0x56555908 <+57>: push 0x565559f0

0x5655590d <+62>: push 0x5655702c

0x56555912 <+67>: call 0xf7f3c2d0 <\_\_strncmp\_sse4\_2>

0x56555917 <+72>: add esp,0x10

0x5655591a <+75>: test eax,eax

0x5655591c <+77>: je 0x5655592e <main+95>

Hàm main có lưu biến check mỗi khi nhập xong in ra thì nó tăng lên 1

Nếu check > 2 thì out, nhỏ hơn 2 thì quay lại nhập tiếp

0x565558f5 <+38>: mov DWORD PTR [ebp-0xc],0x0

... . . .

0x5655591e <+79>: add DWORD PTR [ebp-0xc],0x1

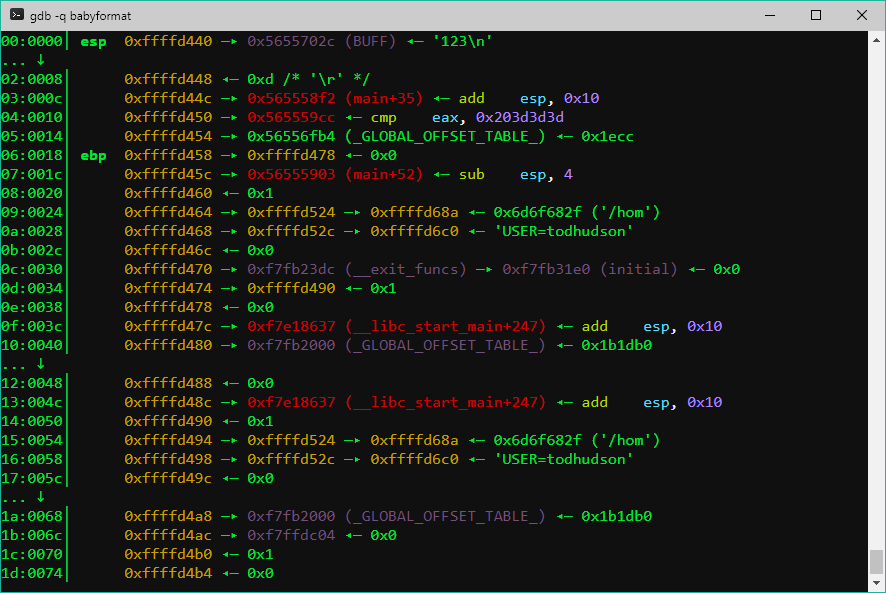
0x56555922 <+83>: mov eax,ds:0x56557008

0x56555927 <+88>: cmp DWORD PTR [ebp-0xc],eax

0x5655592a <+91>: jl 0x565558fe <main+47>

0x5655592c <+93>: jmp 0x5655592f <main+96>

Vậy là mình chỉ nhập được 3 lần, i = 0 ; I < 0x3; mỗi lần nhập được 13 kí tự



Check stack, thì mình có thể leak được libc\_start\_main, từ đó tính được system, vì nx enable nên ko chơi shellcode được

Với 3 lần nhập thì quá ít ỏi để mình exploit được nên mình sẽ reset **variable i** xuống cho nó âm để mình được nhập nhiều lần : )))

Hướng làm:

Leak stack-pointed-stack với libc start main ra

09:0024│ 0xffffd464 —▸ 0xffffd524 —▸ 0xffffd68a ◂— 0x6d6f682f ('/hom')

0a:0028│ 0xffffd468 —▸ 0xffffd52c —▸ 0xffffd6c0 ◂— 'USER=todhudson'

0f:003c│ 0xffffd47c —▸ 0xf7e18637 (\_\_libc\_start\_main+247) ◂— add esp, 0x10

Rồi tính offset biến I, tính ret main ghi system vào

0b:002c│ 0xffffd46c ◂— 0x0

Sau khi ghi xong thì input “EXIT” để ret về system, xong bài.