magicheap

flag

```
FLAG{h34p_0verfl0w_is_e4ay_for_u}
```

參與解題的人

b04902053 鄭淵仁

scripts

- `magicheap.py` ：可以拿到 shell 的 `python3` script。

write-up

漏洞

這支程式提供 `create` 、 `edit` 、 `delete` heap 三種功能，但是在 `edit` heap 的時候，程式又問使用者要輸入幾個 byte。這時候如果回答比原先更大的 size 的話，就會產生 heap overflow了。另外程式的 `RELRO` 的保護是 partial 的。

所以可以使用 heap overflow 搭配來 overwrite fastbin，就可以在下下次 `malloc` 的時候 `malloc` 在有好的 size 的任何一個位置了。

運用 overwrite fastbin 改寫 GOT

仔細觀察 `free@got` (`0x602018`) 附近的位址，會發現有一個 `00000060` 可以當作 fastbin 的 size：

```
0x601ffa:       0x1e28000000000000      0xe168000000000060
0x60200a:       0xe87000007ffff7ff      0x069600007ffff7de
0x60201a:       0x06a6000000000040      0x06b6000000000040
0x60202a:       0x06c6000000000040      0x2800000000000040
0x60203a:       0x422000007ffff7a6      0xd74000007ffff7b0
0x60204a:       0x113000007ffff7a2      0xce7000007ffff7a9
```

所以就使用 overwrite fastbin 來讓下下一次 `malloc` 的位置改到這裡，接下來就可以改 `free@GOT` 的 value 了。

所以首先把 `free@GOT` 改寫成 `printf@plt` ，來用 `free` call `printf` 來 leak `libc` 的資訊。

而為了讓 `printf` 可以 leak 到資訊，我事先把某一塊 chunk 送到 unsorted bin 裡面，再用 overflow 讓上一塊填滿到下一塊的 `fd` 的 address 之前，接下來就可以用 `free` call `printf` 來 leak 到 `main_arena` 的 address，也就可以 leak 出 `_libc_system` 的 address 了。

有了 `_libc_system` 之後，再把 `free@GOT` 改寫成 `_libc_system` ，再讓其中某一塊 chunk 的值是 `"/bin/sh"` ，再 `free` 掉那一塊 chunk 之後，就會執行 `system("/bin/sh")` ，也就可以拿到 shell 了。