題目:final
解法:
助教已經弄好了
有三種方法:
1. UAF:先 release 一個 animal1,在 allocate 另一個 animal2 並且 name size 是 animal 的時候會 get 到上面那個 release 的 animal1,就可以透過改 animal2 name 來改 animal1 裡面的資料，因為她 code 裡面有 call function ptr,我們覆蓋這個 ptr 成 system,在 call 這個 ptr 就得到 shell
2. overwrite 另一個 chunk 的 name 和 len, 讓他寫入 animals[1]的 func ptr 當作 one gadget
3. free hook:release 一個 chunk 之後,改它的 fd(使用上面相同的手法來改)改成 free_hook-8,之後 allocate 的時候就會 allocate 到 free_hook-8 ,allocate 後寫入 system,之後 free 上面存了"/bin/sh" 的 memory 就會 call hook("/bin/sh/"),以現在的 case 就是 call 了 system("/bin/sh")

題目:easyheap
1. Get heap:
新增一個 book,刪掉,在 list 出來
2. 新增一個 0x420 size 的 chunk 並且 delete, 他會進入 unsorted bin
新增一個 book1 在刪掉,在新增另一個 book2 他的 name 會是 book1
並且 edit book2,讓他把 book1 的 name 改成 unsorted bin 裡面 chunk 有 libc 的 address,之後 show 出 book1 的 name 就會得到 libc
3. 之後就是 free_hook 跟上面那題一樣,就不多贅述

題目:beeftalk
操作步驟:
(1) Python3 chat_server.py , 並且 copy 跑出來的 token
(2) Python3 solve.py 並且把(1)的 token 餵進去

1. 目標: get heap base address
首先觀察可以發現 signup 裡面初始化的部份, 除了 name 都是 readstr,都會放 null 在字串後面
所以如果我們想利用 malloc 得到的 chunk 來 leak heap address,只有 name 能用
但是 name 超過 0x20 的話想要多大的 chunk 就要寫入多少個 A,基本上就會把我們要的 heap address 都給蓋掉
所以我們只能使用 0x30 chunk size,我們想要讓 user1 的這個 chunk 的 size 變成 0x30(因為 free 掉之後會有許多 address 殘留在上面,name,desc,job,fifo0,fifo1)
然後新增 user2 且讓 user2->name 是 0x20 的 size,所以會取得 user1,user2->name = user1 ,user2->name 寫入 0x10 個 A,
此時 user1 就會變成 AAAAAAAAAAAAAAAA_address_of_user1_desc
在讓 user2 print 出 name 就行了

- 那麼如何改 user1 的 chunk size?

方法：用 heap overflow 來改：觀察 line 240 可以發現 chat_buf 有 heap overflow 可以拿來改 chat_buf 的下面那塊 chunk 的 size

```
# 先創出一塊 0x110 大小的 chunk,讓 chat_buf 之後可以取得這塊,改這一塊下面的
tok1,_ = signup(r,b"A"*0x100,keep = "y")

# 這塊就是要被改的 user
tok2,_ = signup(r,b"A"*0x3,keep = "y")
tok3,_ = signup(r,b"A"*0x100,keep = "y")
```

```
# free 掉,使之後 chat_buf 可以取得
delete_account(r,tok1)

#  這個是負責去寫 chat_buf 的 user,  因為 chat_buf 會先把 user->name copy 進來
#  如果想要達到 heap overflow,就要 username 是  0x100 個 A  先把 chat_buf 填滿
#  之後就可以填入要 overflow 的值,overflow 的值會是要送的 message copy  進
chat_buf,如下
#  把 tok2 的 user chunk size 改成 0x31
chat_client(r,tok3,b"B"*6 + p64(0x31))

#  此時 free tok2 會讓 tok2 的 user  跑進 0x31 的 tacache
delete_account(r,tok2)
```

2. Leak libc:

因為我們最多只能 malloc 出 0x110 的 chunk size,所以我們無法直接 alloc 出 0x420 的 chunk,讓它進入 unsorted bin,所以我們以塞報 tcache 的形式,讓有些 chunk 進入 smallbin,不過我們下面塞完之後發現有些跑到 unsorted bin 了,反正上面會有 libc 的 address 就好

3. Free hook:有了 heap, libc 之後,就跟我們上一個 lab 的方式一樣,使用 free hook 就可以了