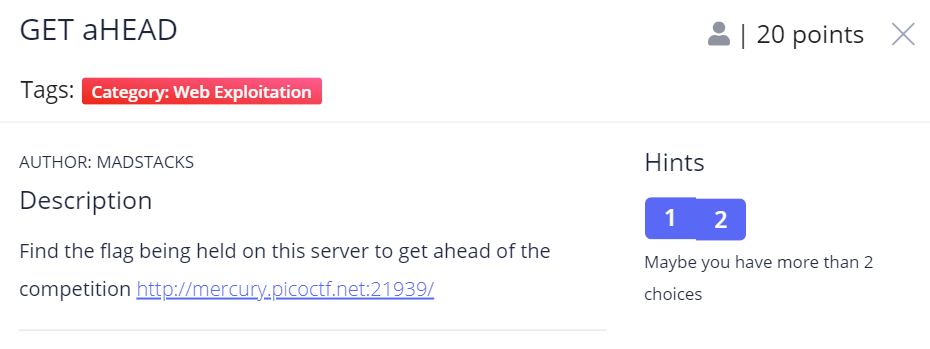
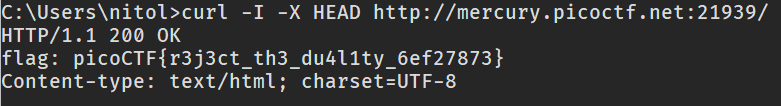
PICOCTF 2021 Writeup WEB

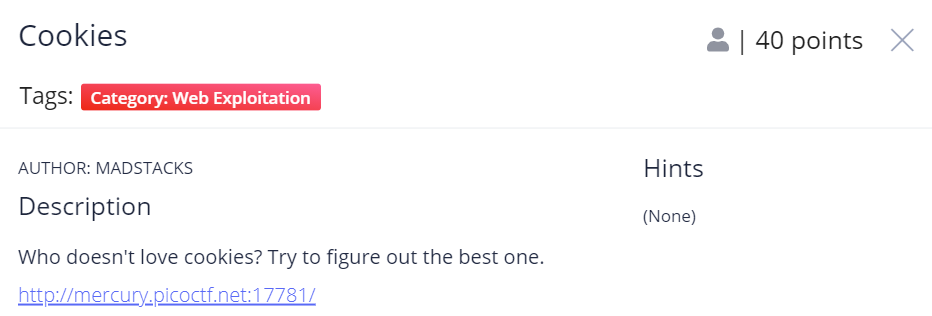
**GET aHEAD**



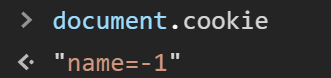
Từ description mình nghĩ ngay đến **HTTP HEAD method requests, dùng curl ta có được flag.**



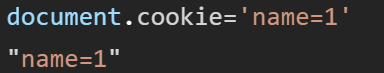
**Cookies**

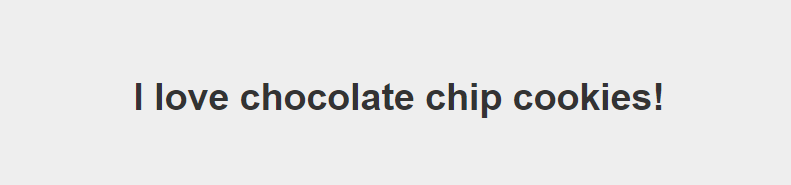


Từ title ta kiểm tra cookie của trang web

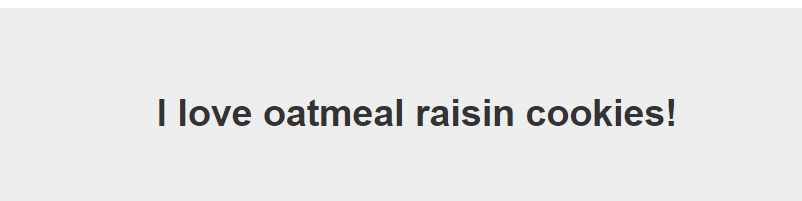


Name = -1 🤔, name=-1 ta thấy vấn đề ở đây thử đổi lại 1:





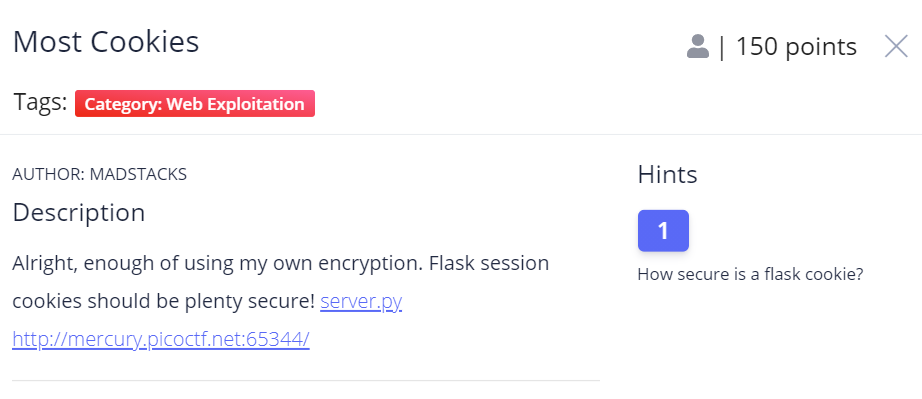
Thử tăng dần name=2:



Có vẻ như mỗi lần tăng dần index ta được kí tự khác nhau:

* Viết code để solve thôi
* 

**MOST COOKIES**



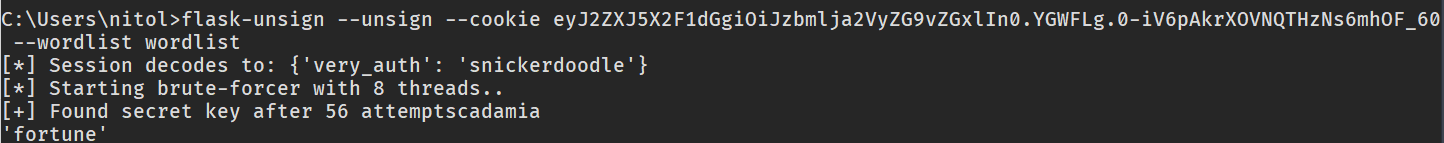
Với tiêu đề và hints ta focus nhanh vào cookie

Secret\_key được chọn random từ list cookie\_names, theo kinh nghiệm solve các challenge từ các giải trước mình chắc chắn 69,96% là dựa vào danh sách trên để usign cookie

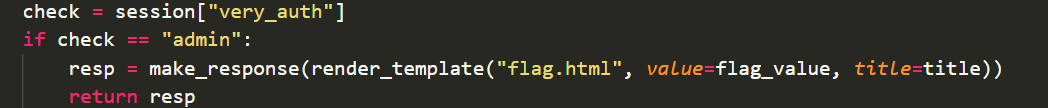
Mình lấy 1 tên bất kì từ danh sách trên để lấy session cookie:

eyJ2ZXJ5X2F1dGgiOiJzbmlja2VyZG9vZGxlIn0.YGWFLg.0-iV6pAkrXOVNQTHzNs6mhOF\_60

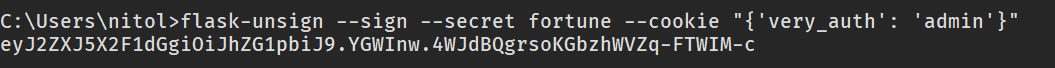
Ta dùng flask-unsign để brute secret với wordlist là danh sách ở biến cookie\_names:



Với session là admin thì trang web sẽ render ra flag !!!



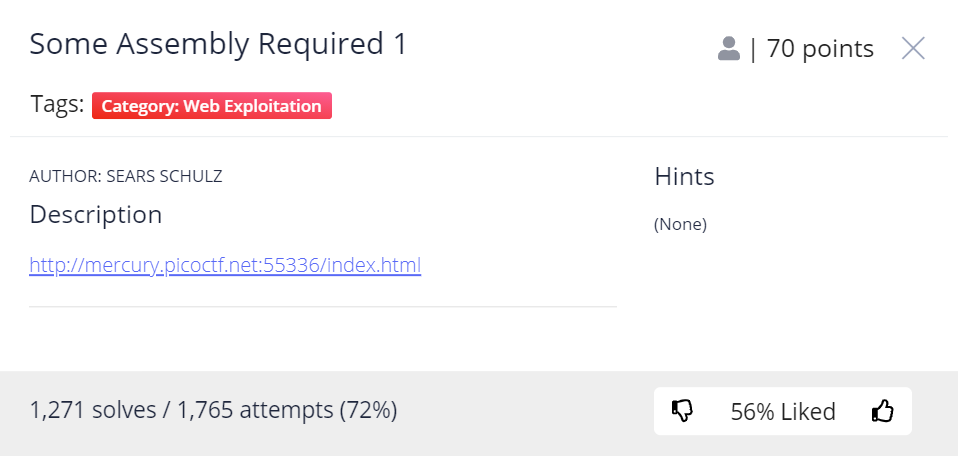
Sign lại với very\_auth=admin bằng secret “fortune”



Change cookie and got flag 😊

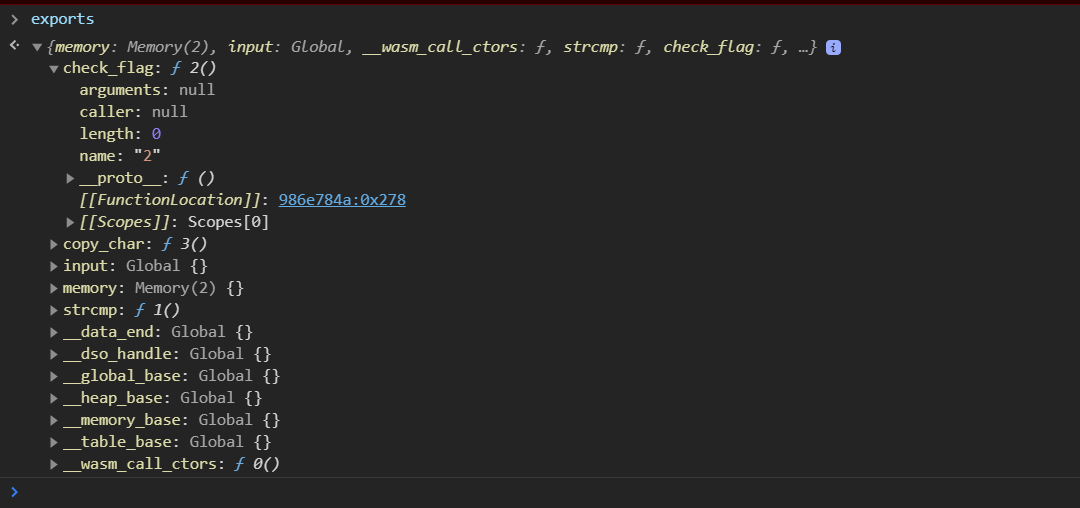


### Some Assembly Required 1



Bài này là dạng WASM khá chuối với người mới tiếp cận asm như mình @@(crypto + re trá hình)

Fuzz từ source ta có khá nhiều thông tin để giải quyết:

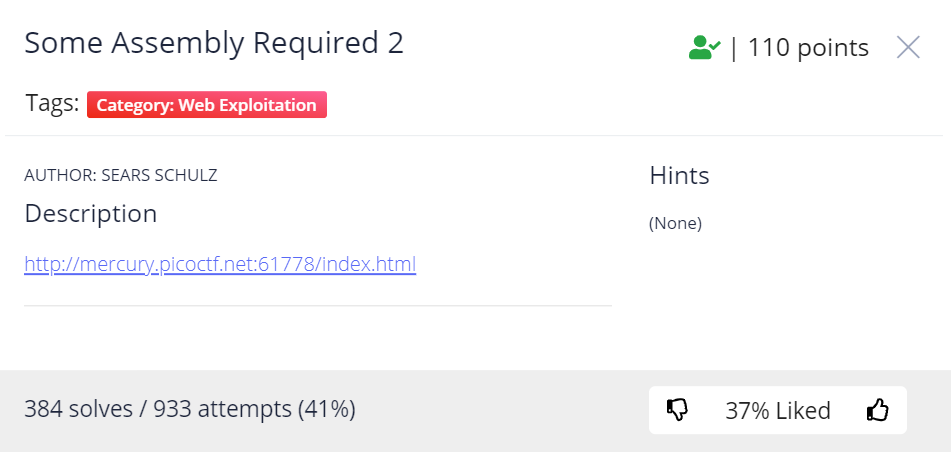
Từ Devtools console dump biến exports, có một hàm khả nghi là check\_flag, nhấn vào functionlocation để xem code wasm



Từ hàm copy\_char có thể hiểu là chương trình tạo ra 1 array tạm sau đó copy từng phần tử từ input

Không có một phép toán nào được thực thi nên mình tin đây là flag và submit.

### Some Assembly Required 2



Giống với bài 1 chỉ thay đổi hàm copy\_char, ý tưởng thuật toán

Lấy mỗi char trong input của user xor với 8 :

Áp dụng tính chất của phép xor A^B=C 🡺 A ^ C = B

Ta chỉ cần xor ngược chuỗi encode với 8 sẽ ra ngược lại flag

Note: ở python cần thêm x vào sau backslash để biểu diễn bytes

```python

data = *b*"xakgK\x5cNs((j:l9<mimk?:k;9;8=8?=0?>jnn:j=lu\x00\x00"

flag = ""

for i in range(0,len(data)):

    flag += chr(data[i] ^ 8)

print(flag)

````

### Some Assembly Required 3

### 

### Download file wasm từ chall

### Bài này khá dài nên mình dùng tool để decomplier sang mã giả javascript

### [WebAssembly/wabt: The WebAssembly Binary Toolkit (github.com)](https://github.com/WebAssembly/wabt)

Mình không có nhiều kiến thức về asm nên sẽ ko giải thích sâu vào phần này

Bài này giống với 2 bài trên chỉ thay đổi hàm copy\_char

Ý tưởng bài toán: cho 2 dữ liệu là chuỗi encoded và chuỗi key,

chủ yếu là xor giữa các index trong mảng, tương tự bài 2

Không vòng lòng belike hải phòng, đọc code mình solve sẽ hiểu

```python

enc = *b*"\x9dn\x93\xc8\xb2\xb9A\x8b\x90\xc2\xddc\x93\x93\x92\x8fd\x92\x9f\x94\xd5b\x91\xc5\xc0\x8ef\xc4\x97\xc0\x8f1\xc1\x90\xc4\x8ba\xc2\x94\xc9\x90"

key=*bytes*.fromhex('f1a7f007ed')

flag = ''

for i in range(0,len(enc)):

  k = 4 - (i % 5)

  flag += chr(enc[i] ^ key[k])

print(flag)

```

picoCTF{730dc4cbcb8e8eab1ca401b6175ff238}