**Write-up cho bài sử dụng lambda**

**Phân tích đề bài**

* Bài toán yêu cầu tìm cách thực thi hàm win() hoặc tái tạo chức năng của nó để đọc file flag.txt và in ra chuỗi mã hex tương ứng.
* Tuy nhiên:
  + Chương trình sử dụng eval() để chạy lệnh bạn nhập.
  + Chương trình tự động thêm () vào cuối input. Ví dụ, nếu bạn nhập print('Hello'), nó sẽ chuyển thành print('Hello')() và gây lỗi cú pháp.
  + Bộ lọc filter() ngăn không cho bạn nhập chuỗi chứa win.

**Hint của bài:**

*"Can you do what win does with your input to the program?"*

=> Điều này có nghĩa là bạn cần tái tạo lại chức năng của win() mà không cần gọi trực tiếp hàm win.

**Ý tưởng giải quyết**

Để vượt qua các hạn chế:

1. Chúng ta cần một cách viết lệnh mà vẫn hợp lệ khi chương trình tự động thêm ().
2. Lambda là giải pháp hoàn hảo, vì:
   * Lambda tạo ra một hàm ẩn danh (anonymous function).
   * Khi chương trình thêm (), lambda sẽ được gọi và thực thi.

**Cách thực hiện**

1. Sử dụng lambda để đọc file flag.txt, xử lý nội dung, và in ra mã hex.
2. Đoạn lệnh đầy đủ:
3. (lambda: print(' '.join([hex(ord(c)) for c in open('flag.txt', 'r').read().strip()])))()

**Giải thích chi tiết**

* **lambda:** Tạo ra một hàm ẩn danh.
* **open('flag.txt', 'r').read()**: Mở và đọc nội dung của file flag.txt.
* **.strip()**: Loại bỏ ký tự thừa, như xuống dòng.
* **[hex(ord(c)) for c in ...]**: Chuyển từng ký tự trong file thành mã hex.
* **' '.join([...])**: Ghép các mã hex thành một chuỗi, cách nhau bởi khoảng trắng.
* **print(...)**: In chuỗi kết quả ra màn hình.

**Nhập lệnh vào chương trình**

Khi kết nối đến server thông qua nc, bạn nhập lệnh sau:

(lambda: print(' '.join([hex(ord(c)) for c in open('flag.txt', 'r').read().strip()])))()

**Kết quả**

Chương trình sẽ đọc nội dung file flag.txt, chuyển đổi từng ký tự thành mã hex, và in ra kết quả. Kết quả này có dạng:

0x70 0x69 0x63 0x6f 0x43 0x54 0x46 0x7b ... 0x31 0x7d

Sau khi chuyển mã hex này về dạng ASCII, ta thu được flag.

**Chuyển đổi mã hex sang ASCII**

Sử dụng Python để chuyển đổi:

ascii\_hex = [

0x70, 0x69, 0x63, 0x6f, 0x43, 0x54, 0x46, 0x7b, 0x66, 0x31, 0x6c, 0x37,

0x33, 0x72, 0x35, 0x5f, 0x66, 0x34, 0x31, 0x6c, 0x5f, 0x63, 0x30, 0x64,

0x33, 0x5f, 0x72, 0x33, 0x66, 0x34, 0x63, 0x37, 0x30, 0x72, 0x5f, 0x6d,

0x31, 0x67, 0x68, 0x37, 0x5f, 0x35, 0x75, 0x63, 0x63, 0x33, 0x33, 0x64,

0x5f, 0x30, 0x62, 0x35, 0x66, 0x31, 0x31, 0x33, 0x31, 0x7d

]

ascii\_string = ''.join(chr(c) for c in ascii\_hex)

print(ascii\_string)

Kết quả:

picoCTF{f1l73r5\_f41l\_c0d3\_r3f4c70r\_m1gh7\_5ucc33d\_0b5f1131}

**Kết luận**

Lambda giúp bạn vượt qua hạn chế thêm () của chương trình và thực hiện được chức năng yêu cầu. Flag cuối cùng là:

picoCTF{f1l73r5\_f41l\_c0d3\_r3f4c70r\_m1gh7\_5ucc33d\_0b5f1131}