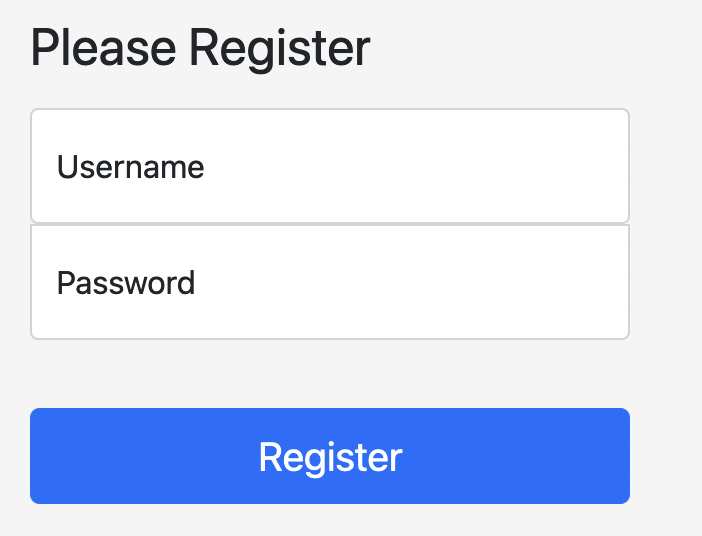
Thử thách này có 4 phần trend[number-of-z]. Mỗi phần sẽ cho bạn 1 flag

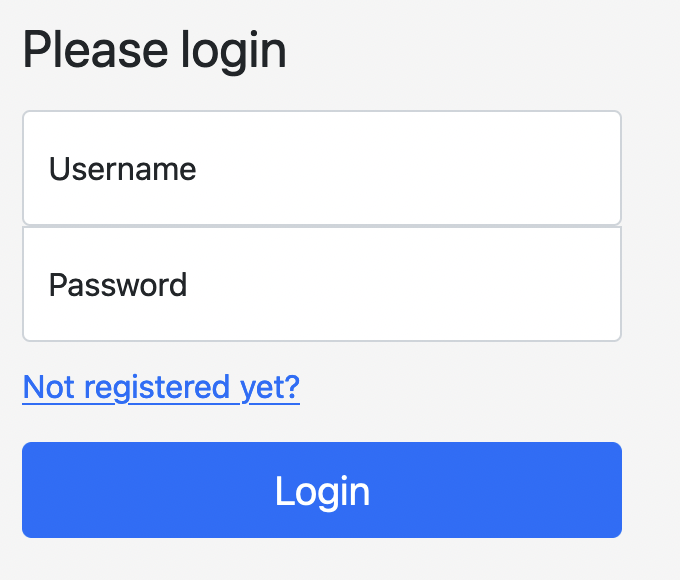
**Phần 1: trendzz**

**Web Enumeration**

http://localhost/register

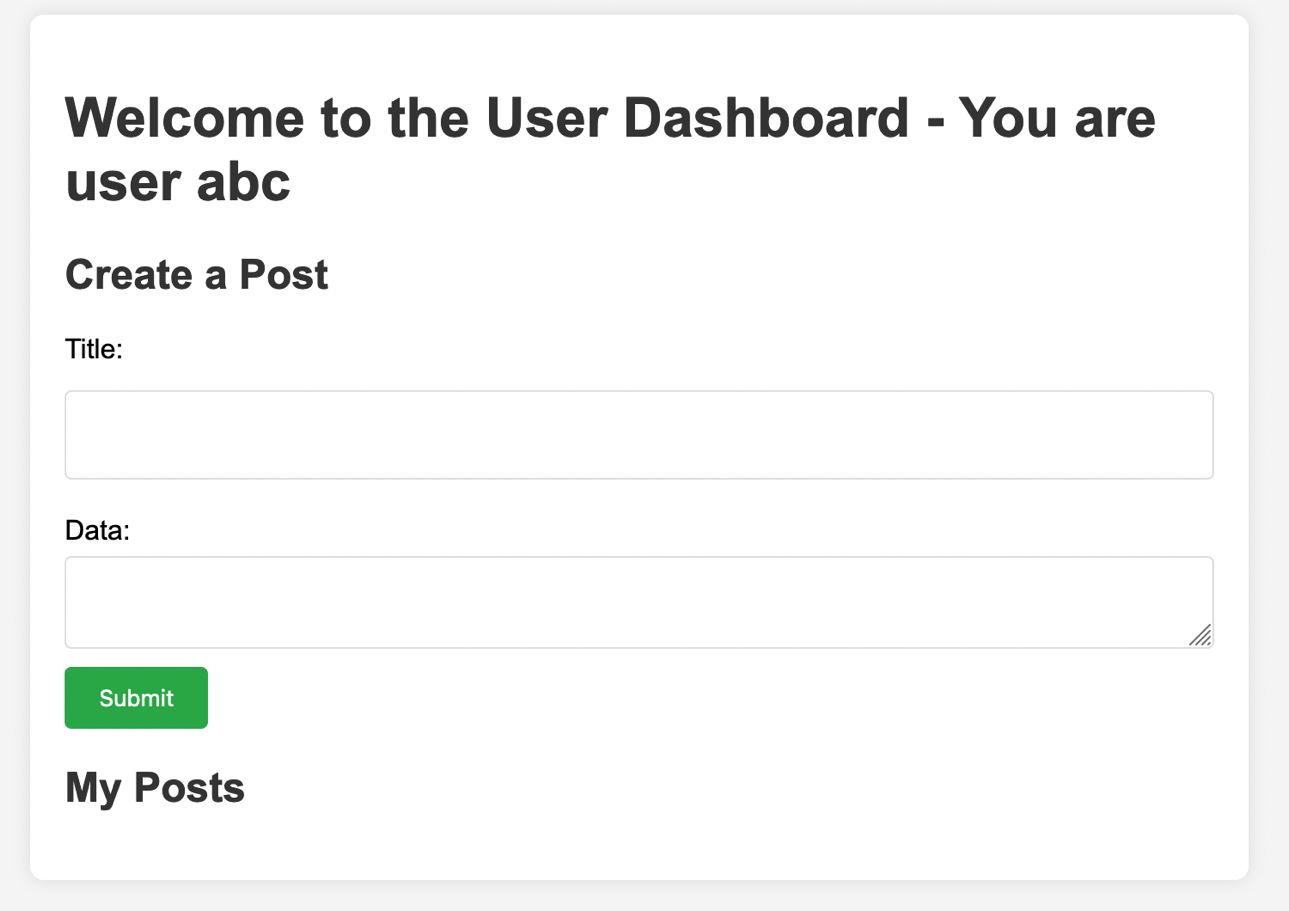


http://localhost/login



Sau khi đăng nhập thành công, bạn sẽ được cấp access\_token: username,role=user

http://localhost/user/dashboard

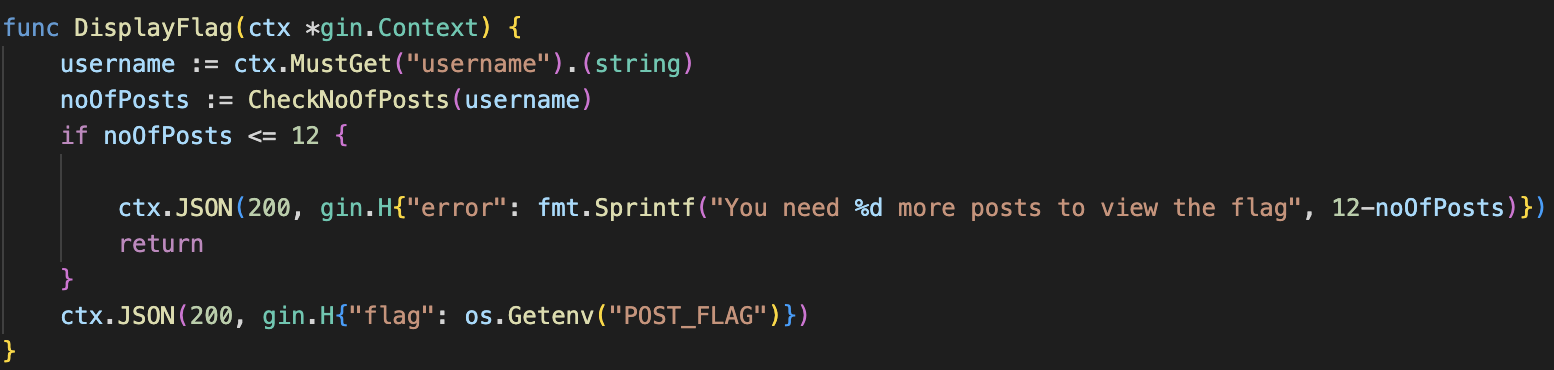


Tạo bài viết: /userposts/create

Xem bài viết: /user/posts/:postid

Xem flag: /user/flag

Muốn xem được flag, số bài viết của bạn phải lớn hơn 12.



Nhưng bạn chỉ tạo được tối đa 10 bài viết

func CreatePost(…){

…

noOfPosts := CheckNoOfPosts(username)

if noOfPosts >= 10 {

ctx.JSON(200, gin.H{"error": "You have reached the maximum number of posts"})

return

}

…

}

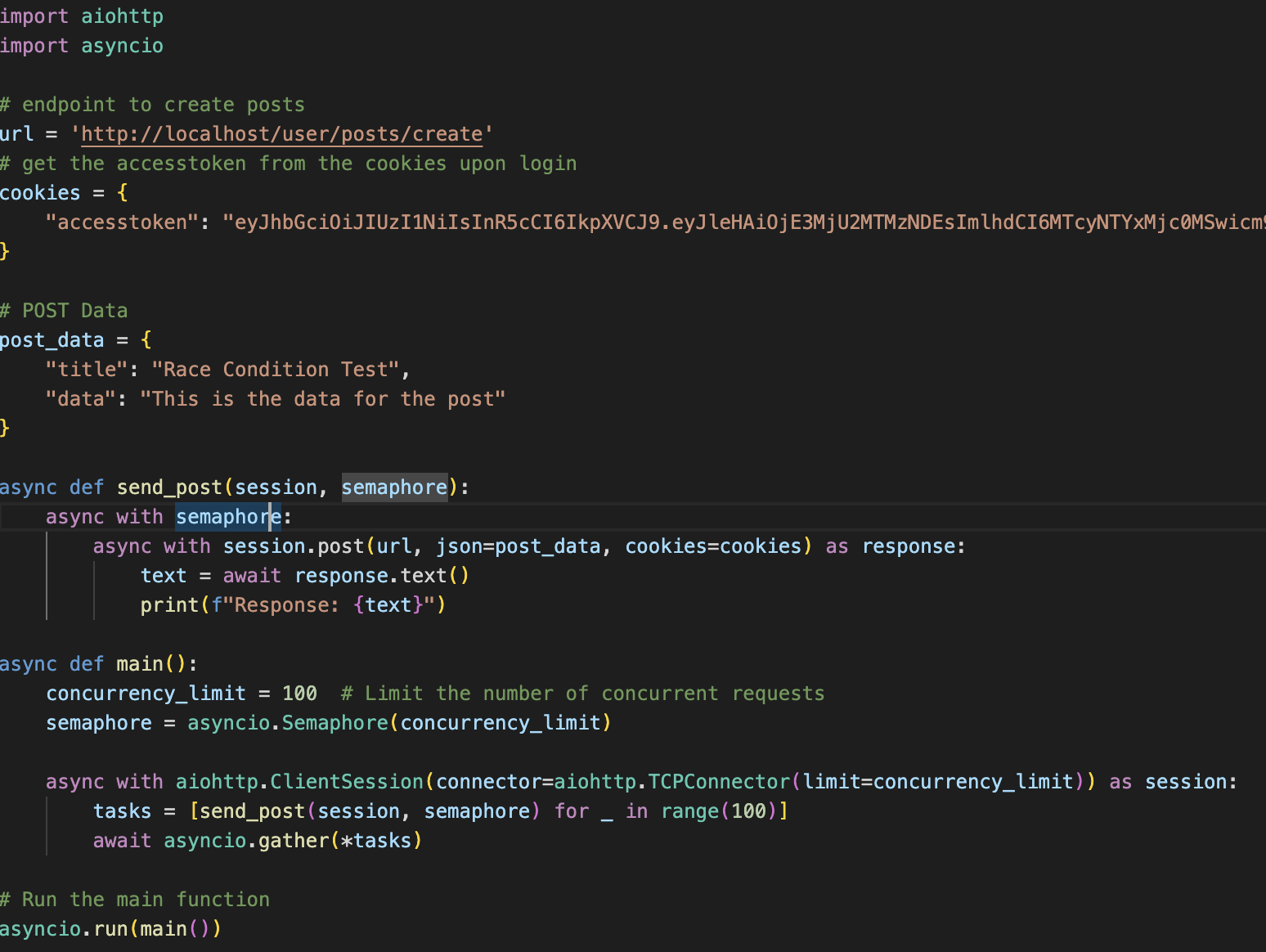
Làm sao để bạn vượt qua giới hạn 10 bài viết?

Nếu bạn nhìn code kĩ hơn, bạn sẽ biết hàm CreatePost bị dính lỗ hổng race conditions vì nó kiểm tra số bài viết trước khi thêm bài viết mới

**Khai thác**

Chúng ta có thể tạo python script gửi nhiều yêu cầu cùng 1 lúc để tạo nhiều hơn 12 viết trước khi nó đếm lớn hơn 10.

solve.py



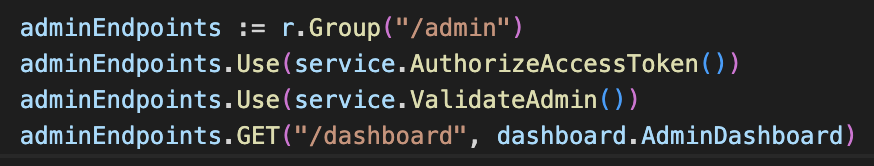
Truy cập /user/flag để nhận flag:

**Flag: CSCTF{flag2}**

**Phần 2: Trendz**

Flag nằm ở trang admin 🡪 access\_token: role=admin

main.go





ValidateAdmin.go

func ValidateAdmin(){

…

if claims["role"] == "admin" || claims["role"] == "superadmin" {

fmt.Println(claims)

} else {

fmt.Println("Token is not valid")

c.AbortWithStatusJSON(403, gin.H{"error": "User Unauthorized"})

return

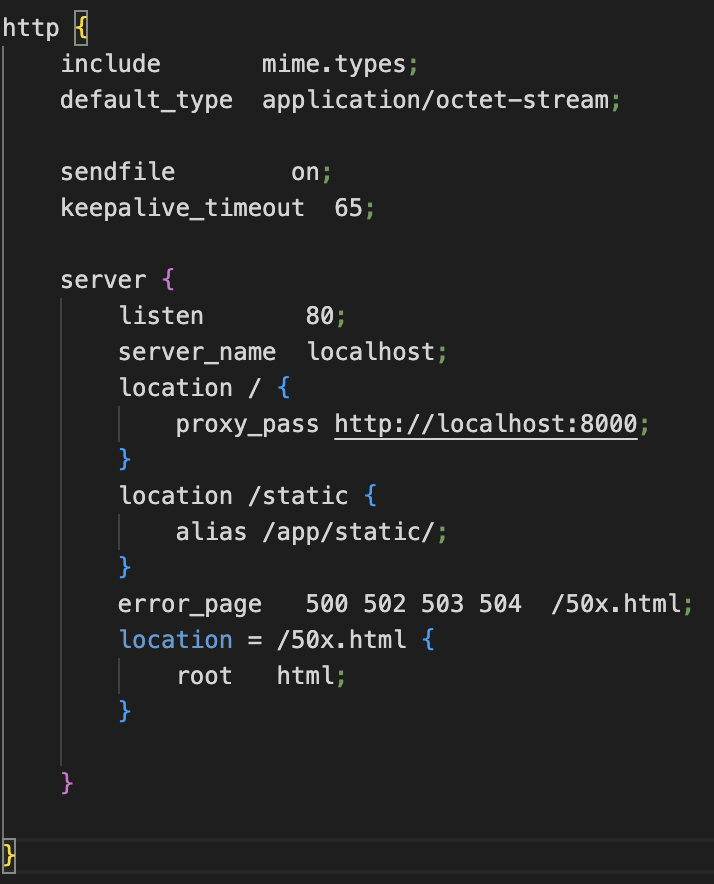
}

…

}

Cách duy nhất để điều chỉnh access\_token: lấy secret\_key

Trong cấu hình của nginx:



Chúng ta thấy /static dẫn đến /app/static/, không hạn chế tập tin chúng ta có thể truy cập (bao gồm jwt.secret). Điều này cho phép chúng ta chuyển hướng đến /static../jwt.secret dẫn đến /app/static/../jwt.secret, là secret key kí cho access token.

Giờ chúng ta có thể tạo JWT token hợp lệ

Truy cập /admin/dashboard để lấy flag

**Flag: CSCTF{flag1}**