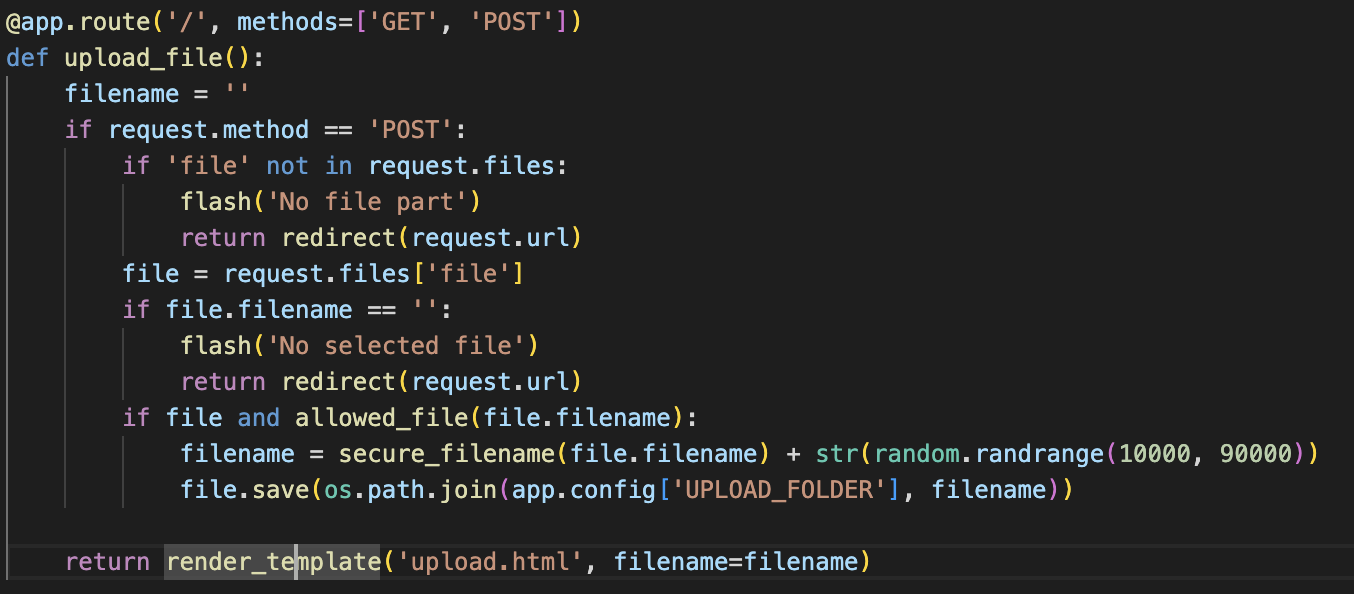
**Web Enumeration**

Tìm hiểu luồng code hoạt động:

* /



Trang chủ cho phép ta upload file nhưng phần mở rộng phải là png hoặc jpg.

Tại sao trang web phải giới hạn các tiện ích mở rộng được phép? Bạn không muốn người dùng có thể tải lên mọi thứ ở đây vì máy chủ sẽ gửi lại dữ liệu cho máy khách. Trang web có thể đảm bảo người dùng không thể tải lên tập tin html sẽ gây ra lỗ hổng XSS. Cũng không cho phép các tập tin .php nếu máy chủ thực thi nó, nhưng có ai cài đặt PHP trên máy chủ đâu ☺

Hàm secure\_filename() làm gì? Có 1 nguyên tắc là không được tin đầu vào của người dùng. Điều này cũng đúng với tên tập tin được tải lên. Tất cả dữ liệu gửi đi có thể bị làm giả, và tên tập tin có thể rất nguy hiểm. Hãy nhớ dùng hàm này để bảo mật tên tập tin trước khi lưu nó trên máy chủ.

filename = "../../../../home/username/.bashrc"

secure\_filename('../../../../home/username/.bashrc')

🡪 'home\_username\_.bashrc'

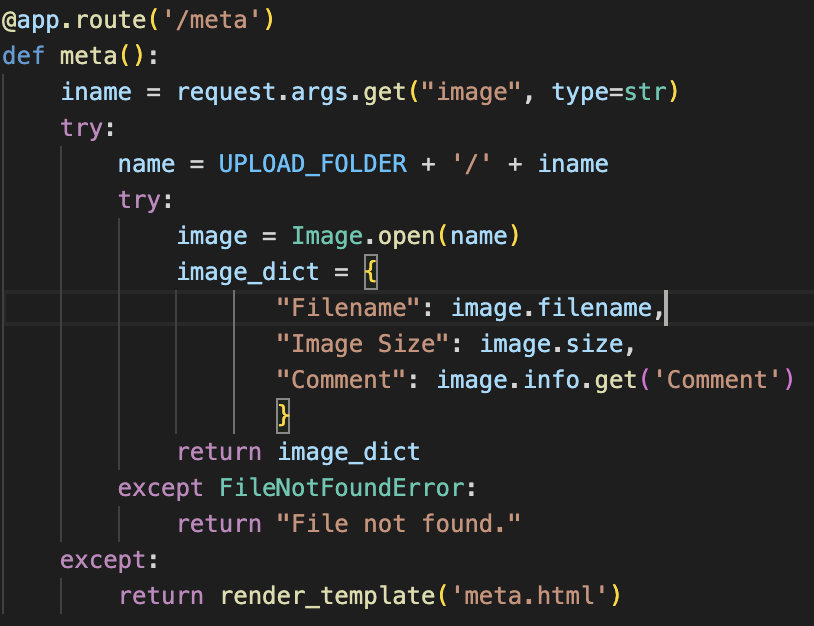
Cách gửi file bằng curl:

curl -X POST -F "file=@rabbit.png" http://chal.competitivecyber.club:9090

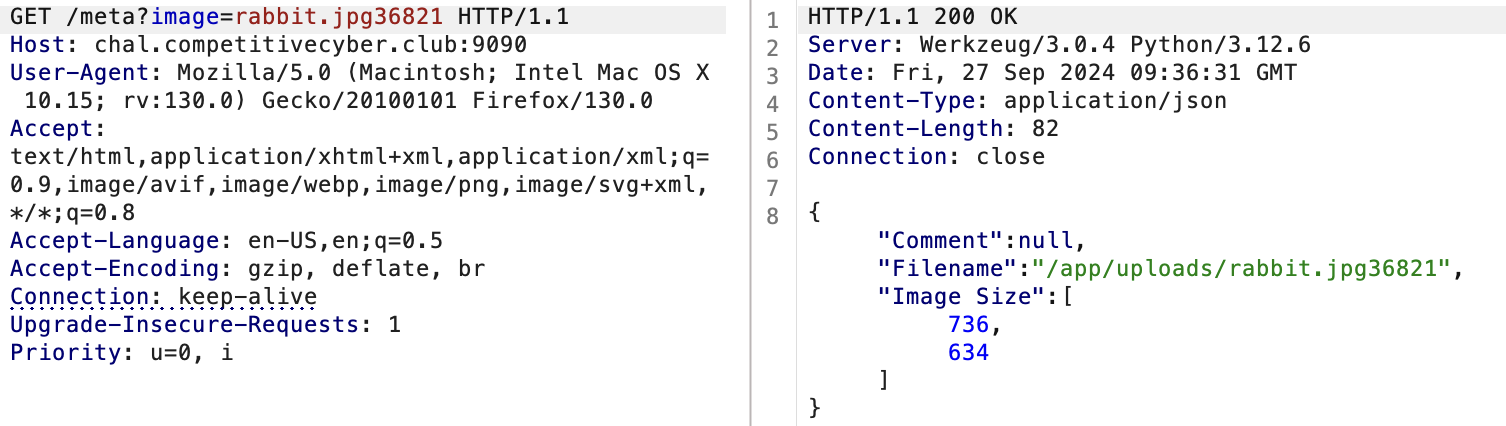
Trang web sẽ trả về tên hình ảnh:



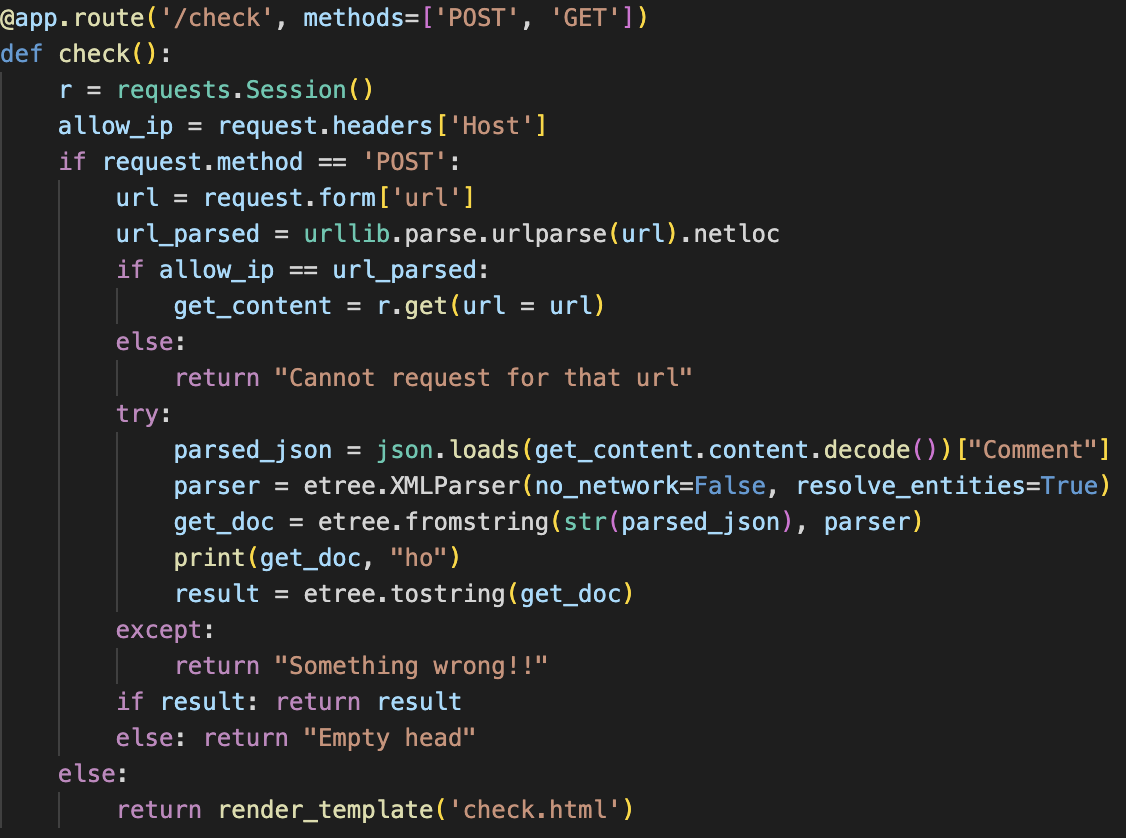
* /meta



Sau khi tải hình ảnh lên, bạn có thể vào /meta để đọc filename, size và comment



* /check



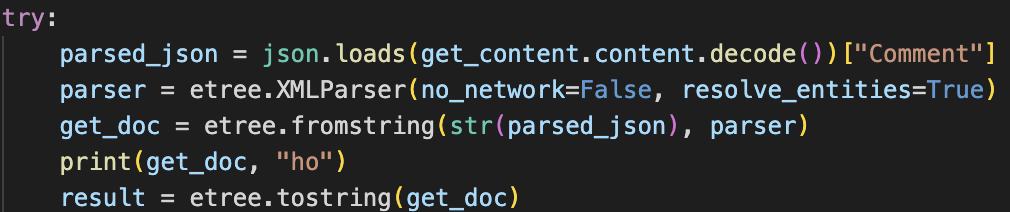
Nó nhận đầu vào là 1 url, kiểm tra url trùng với Host header.

Nếu url của bạn là http://chal.competitivecyber.club:9090/meta?img=rabbit.jpg19347 thì nó sẽ xử lí phần comment của hình ảnh.

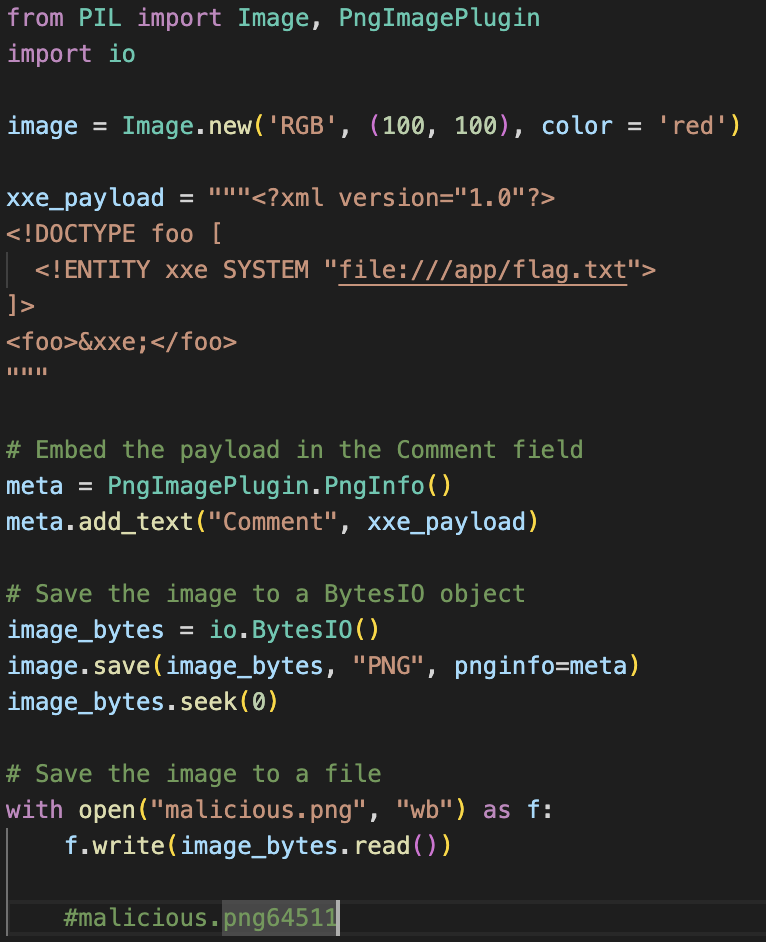
**Exploitation**

Chìa khóa để làm được thử thách này là: Comment không bị sàng lọc và thực thi XML





**Cách 1:** Viết 1 code tạo 1 hình ảnh và thay thế comment với payload XML độc hại để đọc /app/flag.txt. Sau đó kiểm tra nó:

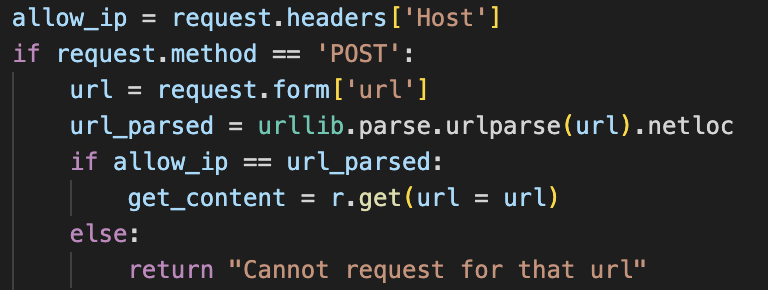




**Cách 2:**

Có hai biến so sánh với nhau trước khi xử lí dữ liệu: allow\_ip và url\_parsed

Giá trị allow\_ip được đặt qua Host header và url\_parsed được đặt bằng tham số url. Nếu allow\_ip == url\_parsed, chúng ta có thể lấy nội dung từ url chúng ta cung cấp và dữ liệu này sẽ được xử lý.



Logic ở đây là url phải được đặt giống với Host header, luôn là chal.competitivecyber.club:9090. Chuyện gì sẽ xảy ra nếu chúng ta có thể chỉnh sửa Host header? Chúng ta có thể đặt nó là webhook, sau đó yêu cầu sẽ được gửi đến webhook của chúng ta.

Chúng ta có thể dễ dàng chỉnh sửa phản hồi của webhook, chúng ta có thể gửi nội dung XML cho máy chủ:

XML và tập tin flag.txt. Đúng vậy là XXE!

{

"Comment ": "<?xml version= '1.0'?><!DOCTYPE foo [ <!ENTITY xxe SYSTEM 'file:///app/flag.txt'> ]><foo>&xxe;</foo>"

}

