**1 SSRF (Server Side Request Forgery)**

* ***Khái niệm:***

SSRF là một loại lỗ hổng bảo mật xảy ra khi kẻ tấn công thao túng ứng dụng web hoặc API để đưa ra yêu cầu tới các tài nguyên nội bộ, có khả năng dẫn đến truy cập trái phép, lộ dữ liệu, xâm phạm hệ thống và thực thi mã từ xa. Kẻ tấn công bỏ qua xác thực đầu vào và buộc các ứng dụng truy cập vào các đích đến web độc hại ngay cả khi được bảo vệ bằng tường lửa hoặc giải pháp Mạng riêng ảo (VPN) .

* ***Các cuộc tấn công SSRF phổ biến***:

Các cuộc tấn công SSRF thường khai thác các mối quan hệ tin cậy để leo thang một cuộc tấn công từ ứng dụng dễ bị tấn công và thực hiện các hành động trái phép. Các mối quan hệ tin cậy này có thể tồn tại liên quan đến server hoặc liên quan đến back-end trong cùng một tổ chức.

* *Server SSRF Attacks*: kẻ tấn công khiến ứng dụng thực hiện một yêu cầu HTTP trở lại máy chủ đang lưu trữ ứng dụng, thông qua giao diện mạng vòng lặp của nó
* *Back-End systems SSRF attacks*: Trong một số trường hợp, server có thể tương tác với back-end mà người dùng không thể truy cập trực tiếp. Các hệ thống này thường có địa chỉ IP riêng không thể định tuyến. Trong nhiều trường hợp back-end chứa chức năng nhạy cảm có thể được truy cập mà không cần xác thực bởi bất kỳ ai có thể tương tác với các hệ thống.
* ***Phân loại***:
* *Standard SSRF*: kẻ tấn công chèn một URL độc hại như một phần của dữ liệu đầu vào của người dùng, kích hoạt máy chủ thực hiện yêu cầu đến tài nguyên đã chỉ định hay trực tiếp quan sát phản hồi từ máy chủ và thu thập thông tin về mạng nội bộ, chẳng hạn như cách truy xuất dữ liệu hoặc xác định các dịch vụ có thể truy cập được.
* *Blind SSRF*: kẻ tấn công không trực tiếp nhận được phản hồi từ máy chủ. Thay vào đó, kẻ tấn công gián tiếp xác nhận sự thành công hay thất bại của cuộc tấn công SSRF bằng cách quan sát những thay đổi trong hành vi của ứng dụng.
* ***Cách vượt qua các biện pháp phòng thủ***
* Lọc trên blacklist-based:

+ Sử dụng biểu diễn IP thay thế

+ Đăng ký tên miền của riêng mình

+ Làm tối nghĩa các chuỗi bị chặn bằng cách sử dụng mã hóa URL hoặc thay đổi chữ hoa chữ thường.

* Lọc trên whitelist-based:

+ Nhúng thông tin xác thực vào URL trước tên máy chủ bằng cách sử dụng ký tự @

+ Sử dụng ký tự # để chỉ ra một đoạn URL

+ Tận dụng hệ thống phân cấp đặt tên DNS để đưa dữ liệu đầu vào cần thiết vào tên DNS đủ điều kiện mà bạn kiểm soát

+ Mã hóa ký tự URL để gây nhầm lẫn cho mã phân tích cú pháp URL

* Bỏ lọc thông qua chuyển hướng mở
* ***Cách tìm kiếm lỗ hổng SSRF***

Lỗ hổng SSRF thường xuất hiện tại:

* Partial URLs in requests
* URLs data formats
* Referer header

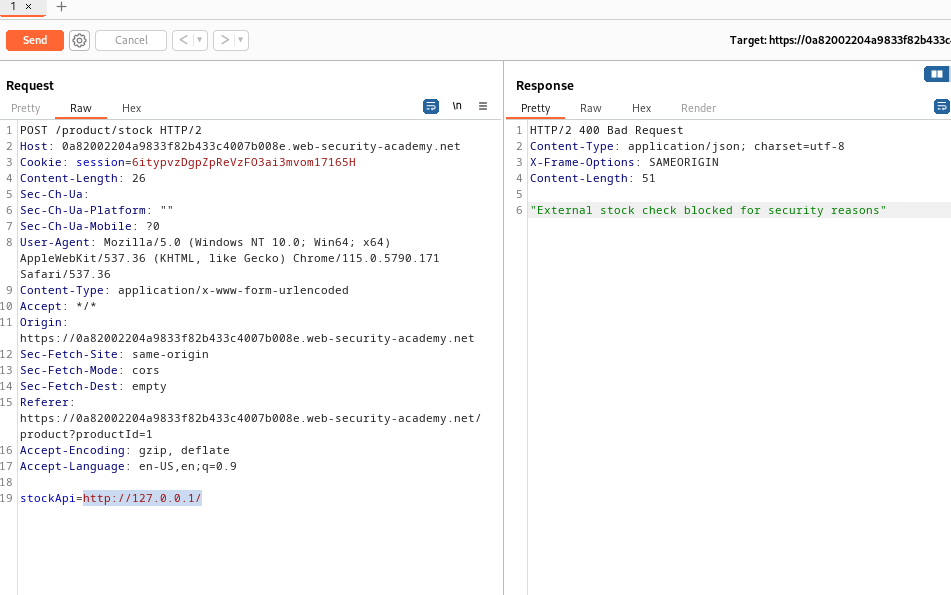
**VD:**

Trong ví dụ này, phần stockApi thực hiện chuyển hướng Url để kiểm tra số sản phẩm còn tồn tại. Điều này khiến máy chủ gửi yêu cầu tới URL đã chỉ định, lấy trạng thái kho và trả về cho người dùng. Ta có thể tận dụng nó để thực hiện SSRF:

A screenshot of a computer

Description automatically generated

Thay đổi URL trong stockApi tham số thành <http://127.0.0.1> và thấy yêu cầu bị chặn có thể do đã bị trùng vào blacklist:



Sử dụng biểu diễn IP thay thế của 127.0.0.1 như 127.1 :

A screenshot of a computer

Description automatically generated

Thay đổi URL trong stockApi tham số để chuyển hướng đến trang admin và thấy yêu cầu bị chặn có thể do admin bị trùng vào blacklist :

A screenshot of a computer

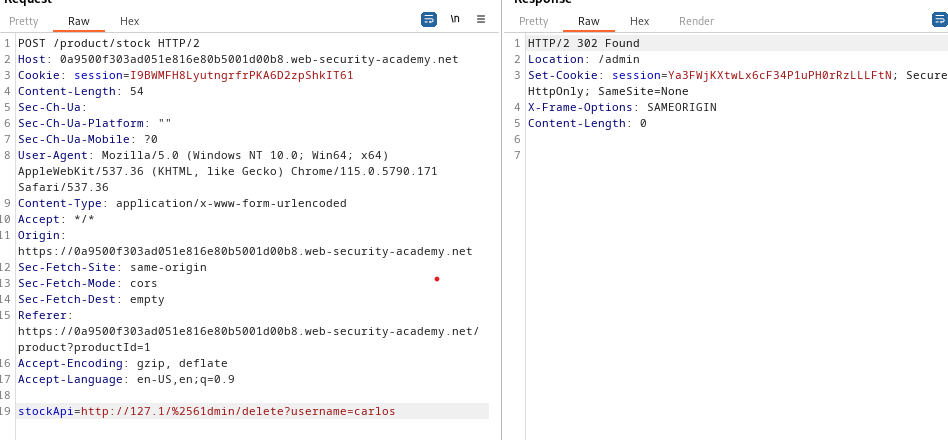
Description automatically generated

Làm tối nghĩa "a" bằng cách mã hóa URL kép thành %2561 để truy cập giao diện admin:

A screenshot of a computer screen

Description automatically generated

Như vậy ta đã truy cập được vào trang admin mà không cần đăng nhập với tài khoản admin có thực thi các hành động với quyền admin như xóa tài khoản :



A screenshot of a computer

Description automatically generated