1. A06:2021 – Vulnerable and Outdated Components

* YAML(Yet another markup language): Là ngôn ngữ tuần tự hóa dữ liệu mà con người có thể đọc được, thường được sử dụng cho các tệp cấu hình và trong các ứng dụng nơi dữ liệu đang được lưu trữ hoặc truyền đi

Ảnh có chứa văn bản

Mô tả được tạo tự động

* Khi truy cập vào challenge ta được 1 trang web để chuyển đổi file Yaml thành File Json

Ảnh có chứa Website

Mô tả được tạo tự động

* Ta thử tạo 1 file Yaml đơn giản và upload lên

Ảnh có chứa văn bản

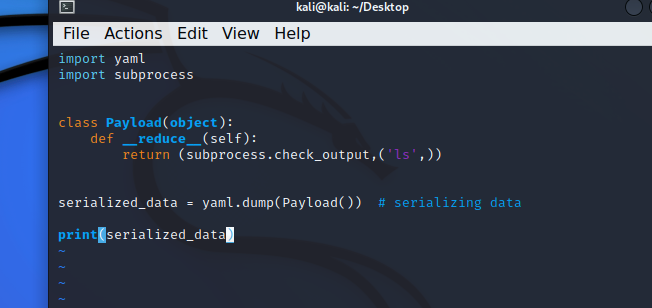
Mô tả được tạo tự động

* Ta thấy cụng không có gì lạ

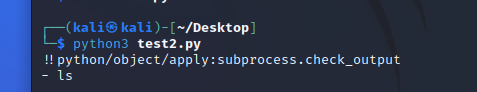
Ảnh có chứa văn bản

Mô tả được tạo tự động

* Khi kiểm tra version thì ta biết được đây là trang web sử dụng phiên bản pyyaml v5.1
* Khi tìm hiểu thì biết ở phiên bản này có 1 lỗ hổng
* Tìm hiểu về lỗ hổng này thì ta biết là ở phiên bản này Yaml dễ bị vulnerable bởi vì nó có khả năng serialize các đối tượng của các lớp và các phương thức
* Lỗ hổng này xảy ra khi attacker có thể chèn mã Yaml độc hại vào 1 tệp Yaml hoặc chuỗi Yaml, sau đó khi tệp hoặc chuỗi này sẽ được chuyển đổi bởi PyYAML, dẫn đến việc thực thi mã tùy ý
* Nó xảy ra khi PyYAML được sử dụng để chuyển đổi một tệp YAML được tạo ra đặc biệt chứa một Python object constructor, có thể được sử dụng để thực thi mã tùy ý trên hệ thống máy chủ.



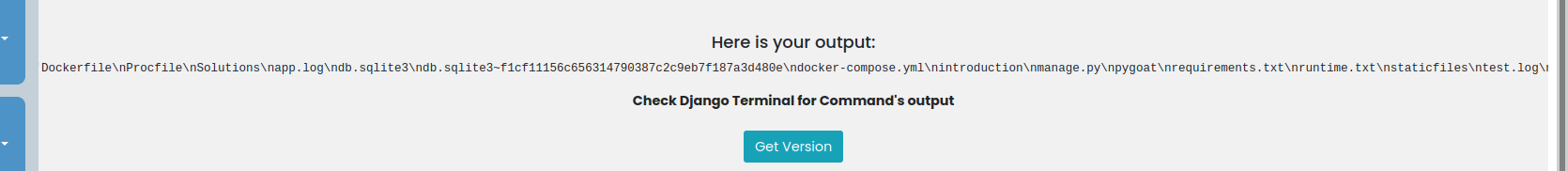
* Ta viết 1 code python để tạo 1 đối tượng payload để trả về 1 tuple chứa hàm “subprocess.check\_output” và đối số là “ls”, sau đó đối tượng được chuyển đổi thành chuỗi YAML



* Ta dùng chuỗi này tạo thành 1 file YAML sau đó upload lên trang web

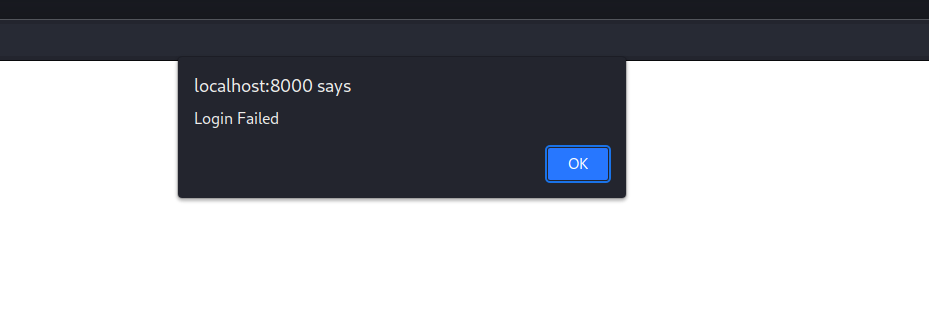
Ảnh có chứa văn bản

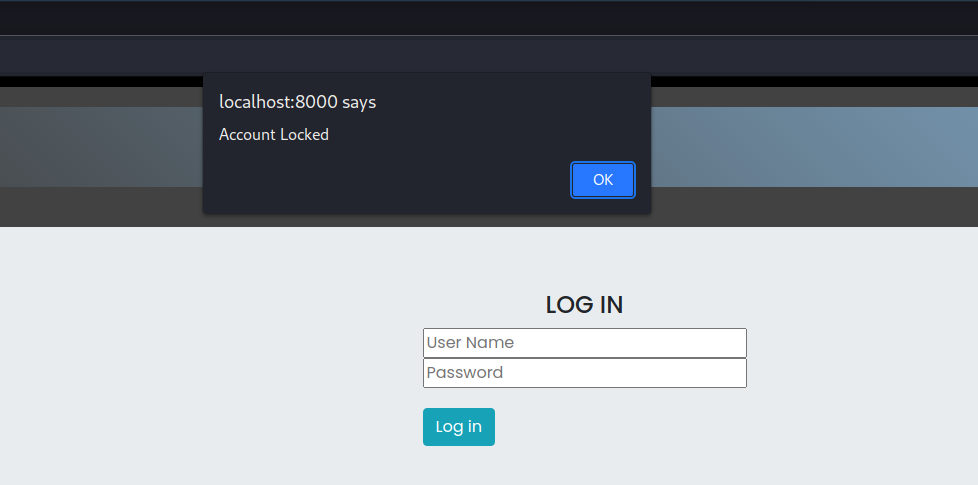
Mô tả được tạo tự động



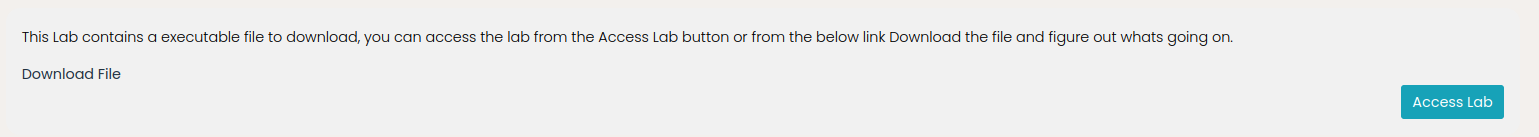
* Thế là ta đã có thể chạy được lệnh “ls” để khai thác lỗ hổng

1. A07:2021 – Identification and Authentication Failures





A08:2021 – Software and Data Integrity Failures

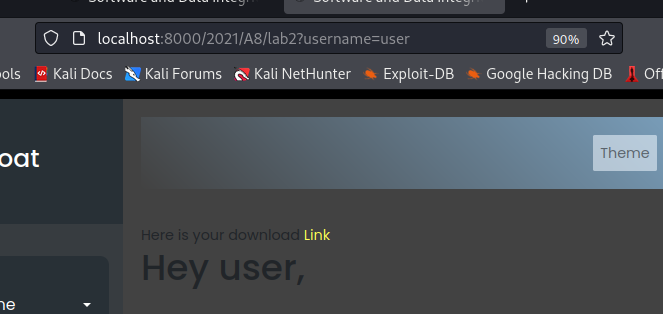


* Truy cập vào challenge thì ta được bảo là trang web này sẽ có chức năng là hiển thị trang download cho người dùng
* Để mở được trang và truy cập thì ta có thể ấn vào nút “Download File” hoặc là nút “Access Lab”

Ảnh có chứa văn bản, Website

Mô tả được tạo tự động

* Khi ấn vào nút acces lab thì ta được hiển thị 1 trang để nhập tên sau đó có nút để download



* Ta sẽ có 1 đường link để khi ấn vào thì sẽ download được tài liệu

Ảnh có chứa văn bản, ảnh chụp màn hình, màn hình, điện thoại

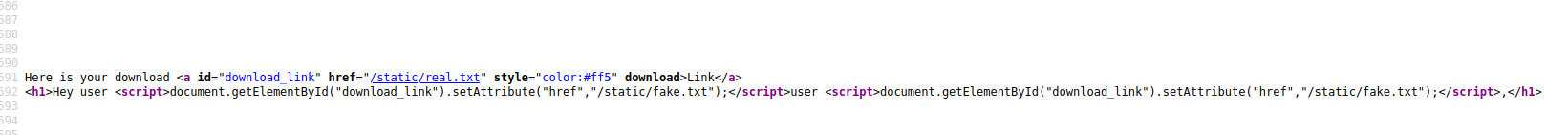
Mô tả được tạo tự động

* Tài liệu ở đây sẽ là real.txt

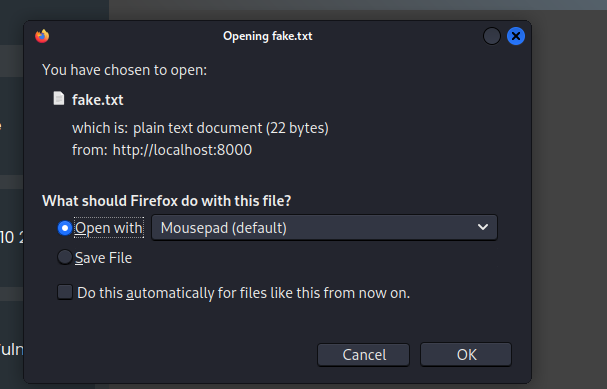
Ảnh có chứa văn bản

Mô tả được tạo tự động

* Còn khi ấn vào nút “Download File” thì sẽ được dẫn thẳng tới trang web để lấy link download mà không cần nhập tên nhưng khi nhìn vào đường dẫn thì nó lại khác so với đường dẫn bình thường
* Nó được chèn 1 đoạn script



* Ý nghĩa của đoạn scipt đó là sẽ tạo 1 liên kết để tải xuống cho tệp “real.txt” nhưng sau đó sử dụng JavaScript để thay đổi đường dẫn của liên kết từ “static/real.txt” sang “static/fake.txt”
* Do đó nếu ta ấn vào liên kết đó, nó sẽ tải xuống 1 tập tin “fake.txt”



A09:2021 – Security Logging and Monitoring Failures

Ảnh có chứa văn bản

Mô tả được tạo tự động

* Truy cập vào challenge ta được cho 1 trang web để nhập user name và password
* Ta nhập thử 1 giá trị và nó bảo ta nhập sai
* Và trang web cũng cho ta thông tin là logs đã bị rò rỉ
* Để kiểm tra logs ta nhập vào URL các subdomain để xem thử có gì xảy ra không

Ảnh có chứa văn bản

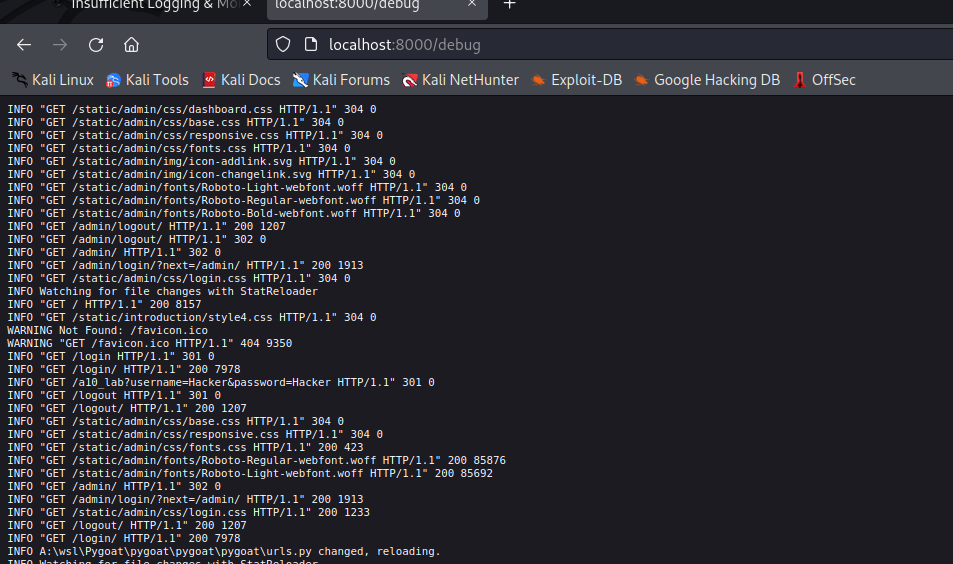
Mô tả được tạo tự động

* Ta nhập vào thử /robots.txt thì trang web bảo là không tìm thấy nhưng lại cho ta các subdomain khác hợp lệ và ta có thể dùng những domain này để kiểm thử hoặc sử dụng để xâm nhập cũng như làm những điều bất hợp pháp

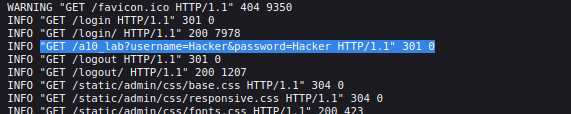
Ảnh có chứa văn bản

Mô tả được tạo tự động

* Khi tìm kiếm subdomain của /10\_lab thì ta thấy có 1 subdomain tên là debug
* Ta truy cập thử

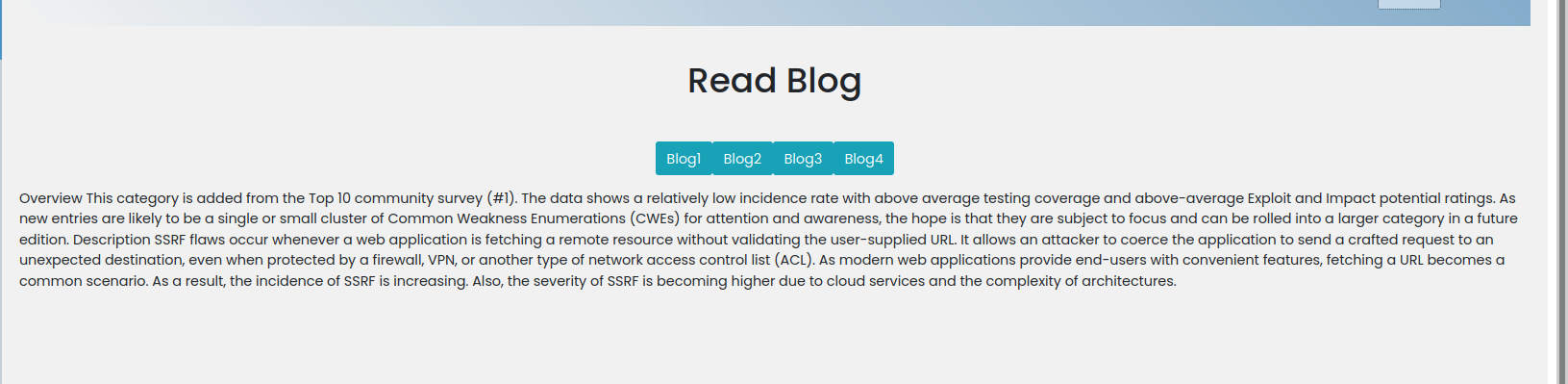


* Ta được 1 trang ghi lại các HTTP REQUEST của người dùng



* Ta kiếm các domain liên quan đến challenge và thấy là có ghi lại log với username=”Hacker” và Passwrd=”Hacker”

e) A10:2021 – Server-Side Request Forgery (SSRF)



* Challenge cung cấp cho ta 1 trang để thể đọc blog

Ảnh có chứa văn bản

Mô tả được tạo tự động

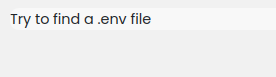
* Khi đọc code của trang web thì ta có thể yêu chức năng của trang web như sau

+ Cho phép người dùng đăng nhập và sau đó truy cập vào 1 tập tin trên máy chủ thông qua một đường dẫn được cung cấp trong yêu cầu POST

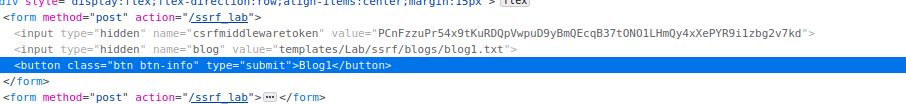
+ Không kiểm tra tính hợp lệ của đường dẫn: View này không kiểm tra tính hợp lệ của đường dẫn truy cập được cung cấp bởi người dùng

+ View này cho phép người dùng truy cập vào bất kỳ tập tin nào trên máy chủ mà nó đang chạy trên thông qua một đường dẫn được cung cấp trong yêu cầu POST. Điều này có thể dẫn đến lỗ hổng SSRF nếu kẻ tấn công có thể điều khiển đường dẫn được cung cấp và truy cập vào các tài nguyên mà họ không được phép truy cập, bao gồm các tài nguyên trong mạng nội bộ.

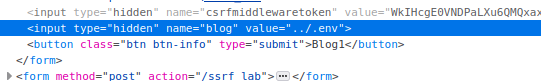
+ không kiểm tra quyền truy cập của người dùng đến tệp được yêu cầu. Điều này có thể dẫn đến việc người dùng truy cập vào các tệp mà họ không được phép truy cập.



* Yêu cầu của challenge này là ta tìm được file .env



* Inspect button 1 thì ta thấy có 1 button hidden nó chứa đường dẫn đến thư mục mà ta đọc blog
* Ta có thể thay đổi đường dẫn này tới bất kỳ đâu hoặc bất kỳ file nào



* Ta sửa nó thành ../.env để truy câp vào file .env

Ảnh có chứa Website

Mô tả được tạo tự động

* Sau khu sửa ta ấn lại nút Blog1 1 lần nữa để xem kết quả