

2

Lab

PHỤC VỤ MỤC ĐÍCH GIÁO DỤC
FOR EDUCATIONAL PURPOSE ONLY

Tổng quan các lỗ hổng bảo mật web thường gặp (phần 2)

Thực hành môn Bảo mật web và ứng dụng

Tháng 3/2024

Lưu hành nội bộ

<Ng nghiêm cấm đăng tải trên internet dưới mọi hình thức>

A. TỔNG QUAN

1. Mục tiêu

- Giúp sinh viên có cái nhìn tổng quan hơn về các lỗ hổng web thường gặp thông qua tìm hiểu các lỗ hổng thuộc top 10 oswap 2021.
- Ở bài thực hành 2, sẽ tìm hiểu top từ 6 đến 10 của OSWAP. Sinh viên cần hiểu rõ cách thức lỗ hổng này xảy ra và cách tiếp cận để khai thác nó, đồng thời có giải pháp để khắc phục các lỗ hổng.

2. Thời gian thực hành

- Thực hành tại lớp: 5 tiết tại phòng thực hành.
- Hoàn thành báo cáo kết quả thực hành: tối đa 7 ngày.

B. CHUẨN BỊ MÔI TRƯỜNG

1. Docker

- Môi trường thực hành sử dụng docker được cung cấp

```
docker load -i oswap.tar
```

- Kiểm tra image được load vào.

```
docker images
```

- Chạy môi trường bài thực hành

```
docker run --rm -p 8000:8000 oswap
```

2. Phần mềm yêu cầu

- Phần mềm Burp Suite được cung cấp hoặc bất kỳ một phần mềm proxy nào mà sinh viên sử dụng quen thuộc.

C. THỰC HÀNH

1. Danh mục các lỗ hổng thuộc top 10 OSWAP 2021

Top 10 oswap là một tài liệu giành cho việc nâng cao nhận thức cho các nhà phát triển và nhà bảo mật ứng dụng web.

a) A06:2021 – Vulnerable and Outdated Components

i. Mô tả

- Thành phần phần mềm là một phần của hệ thống hoặc ứng dụng giúp mở rộng chức năng của nó, chẳng hạn như mô đun, gói phần mềm hoặc API. Các lỗ hổng dựa trên thành phần xảy ra khi một thành phần không được hỗ trợ, lỗi thời hoặc có lỗ hổng bị khai thác đã biết.

ii. Kích bản tấn công

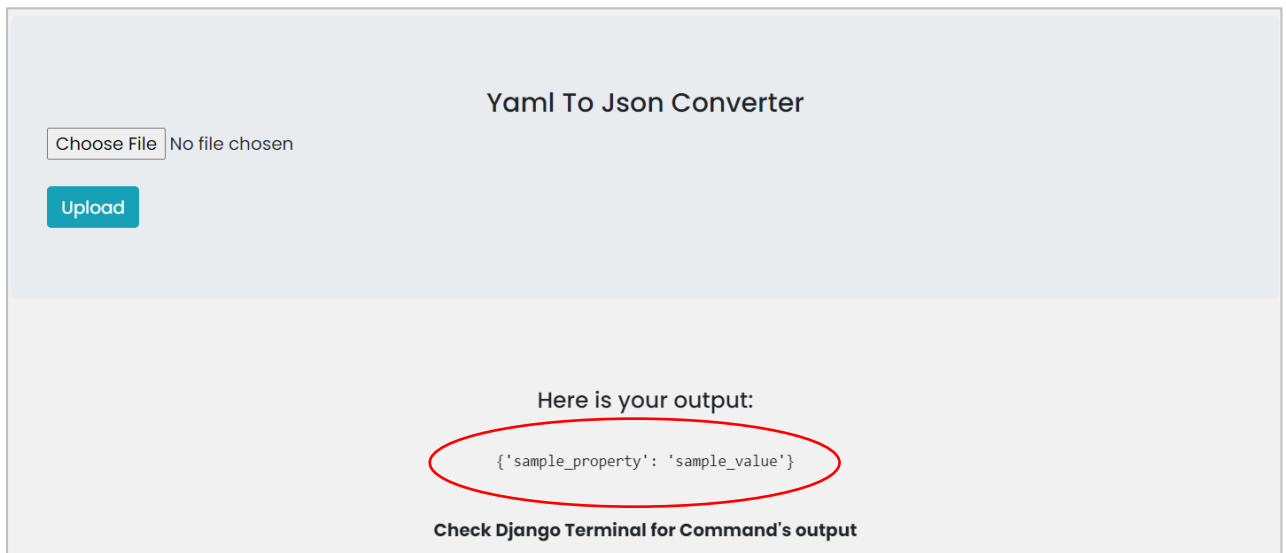
- Bài thực hành giúp cho sinh viên hiểu được tại sao các thành phần có lỗ hổng đã biết trở thành một vấn đề. Người dùng truy cập vào bài lab được cung cấp với tính năng chuyển đổi tập tin yaml thành định dạng json.
- Truy cập bài tập thực hành tại: <http://localhost:8000/a9>

iii. Nội dung

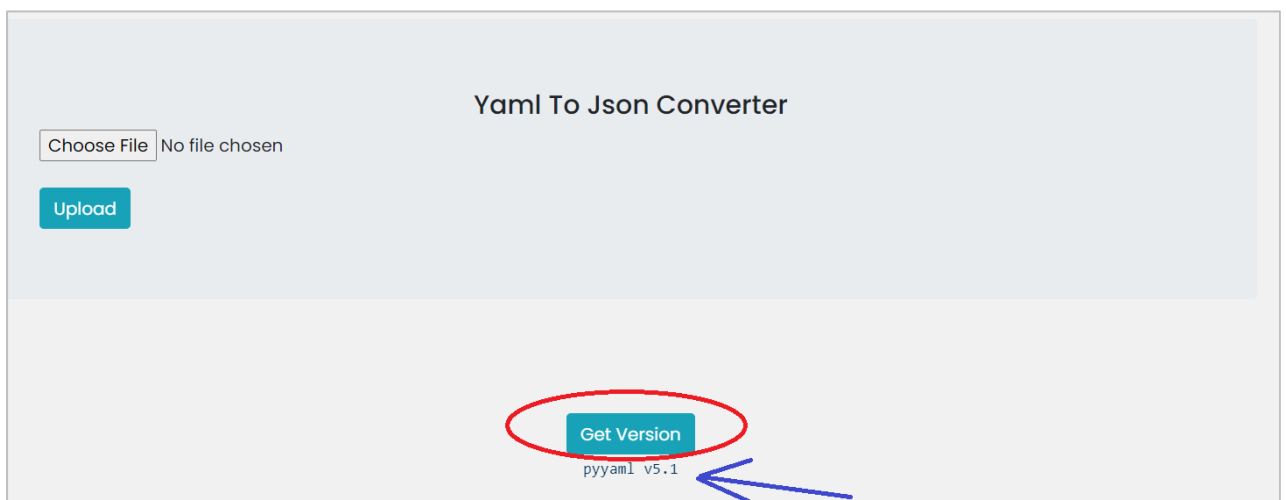
- Đây là một trang dùng để chuyển dữ liệu với định dạng YAML sang JSON.
 - YAML và JSON đều là 2 định dạng lưu dữ liệu dưới dạng text.
- Thực hiện tạo 1 file **sample.yaml** với nội dung như sau:

```
sample_propery: sample_value
```

- Thực hiện upload và nhận được kết quả:



- Ta nhận thấy, có một nút là **Get Version**, kiểm tra và có được thư viện **pyyaml** với phiên bản 5.1:



- Pyyaml có thể thực hiện serialize class object thành nội dung dưới dạng YAML và deserialize ngược lại thành Python. Điều này đôi khi có thể xảy ra việc thực thi code không mong muốn.
- Khi thực hiện tra cứu thì nhận thấy rằng phiên bản pyyaml 5.1 này có thể thực hiện deserialize và thực thi code nếu đó là đoạn code được serialize là class.

Description

PyYAML 5.1 through 5.1.2 has insufficient restrictions on the load and load_all functions because of a class deserialization issue, e.g., Popen is a class in the subprocess module. NOTE: this issue exists because of an incomplete fix for CVE-2017-18342.

Severity

CVSS Version 3.x

CVSS Version 2.0

CVSS 3.x Severity and Metrics:



NIST: NVD

Base Score: 9.8 CRITICAL

Vector: CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

- Ta sẽ thực hiện tạo 1 class với phương thức `__reduce__`. `__reduce__` giúp pyyaml biết làm sao để xử lý một loại dữ liệu nào đó, ở đây là class do ta tự định nghĩa.

```
import yaml
import subprocess

class Payload(object):
    def __reduce__(self):
        return (subprocess.Popen, ('ls',))

deserialized_data = yaml.dump(Payload())
print(deserialized_data)
```

Ở phương thức `__reduce__`, ta sẽ thực hiện trả về 1 tuple với phần tử thứ nhất là hàm ta cần gọi, ở đây là `subprocess.Popen`, và phần tử tiếp theo là tham số. Thực hiện đoạn code và có được payload:

```
!!python/object/apply:subprocess.Popen
- ls
```

- Thực hiện tạo nội dung file payload.yaml với nội dung như trên và thực hiện upload, có được kết quả:

Yaml To Json Converter

Choose File

No file chosen

Upload

Here is your output:

```
<Popen: returncode: None args: 'ls'>
```

Bài tập 1: Thực hiện việc khai thác lỗ hổng với một ứng dụng render Markdown thành HTML. Sử dụng format sau mẫu để trình bày

#Tiêu đề: Tiêu đề rõ ràng và ngắn gọn bao gồm loại lỗ hổng và tài sản bị ảnh hưởng.

#Mô tả lỗ hổng: Lỗ hổng này là gì? Trong các bước rõ ràng, làm thế nào để bạn tái tạo nó?.

Tóm tắt:

Các bước để thực hiện lại và bằng chứng:

1. bước 1

2. bước 2

3. bước 3

Tài liệu hỗ trợ và tham khảo:

* Liệt kê bất kỳ tài liệu bổ sung nào (ví dụ: ảnh chụp màn hình, nhật ký, v.v.)

#Mức độ ảnh hưởng của lỗ hổng: Tác động bảo mật nào mà kẻ tấn công có thể đạt được?

#Khuyến cáo khắc phục: Làm thế nào để vá lỗ hổng này?

Chạy bài tập

```
docker run --rm -p 10001:10001 lab2-vuln-outdated-components-exercise
```

PinaMarkdown - A Markdown renderer that supports front-matter using [react-markdown](#) and [gray-matter](#).

In case you don't know what Markdown is: [MarkdownGuide](#)

Your markdown here

```
# Exploiting this is ez? Right?
```js
const ezzz = true;
```
```

RENDER 🍷

Output 🍷

Content

Exploiting this is ez? Right?

```
const ezzz = true;
```

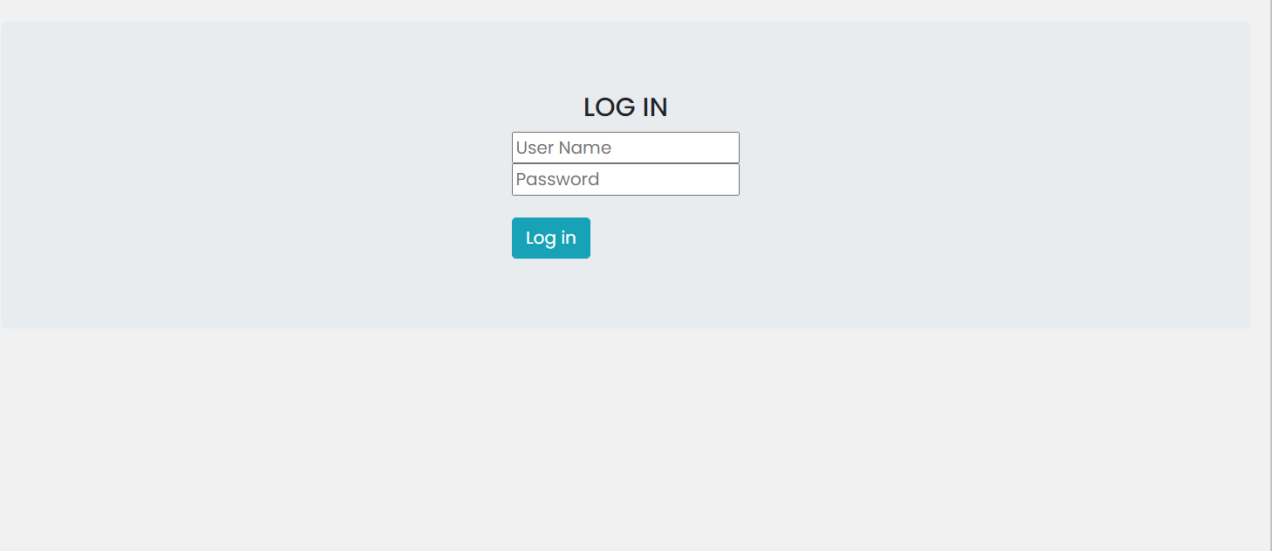
b) A07:2021 - Identification and Authentication Failures

i. Mô tả

- Lỗi nhận dạng và xác thực có thể xảy ra khi các chức năng liên quan đến danh tính, xác thực hoặc quản lý phiên của người dùng không được triển khai đúng cách hoặc không được bảo vệ đầy đủ.
- Kẻ tấn công có thể khai thác các lỗi nhận dạng và xác thực bằng cách xâm phạm mật khẩu, khoá, mã thông báo phiên hoặc khai thác các lỗi triển khai khác để giả định danh tính của người dùng khác, tạm thời hoặc vĩnh viễn.

ii. Kịch bản tấn công thực hành

- Có 1 trang đăng nhập của admin, kiểm tra thử xem có tồn tại lỗi gì trên trang này không.
- Truy cập bài thực hành tại:
http://localhost:8000/auth_failure/lab2/admin12983gfugef81e8yeryepanel



Chậm lại và suy nghĩ 1: Dựa vào thông tin recon được, có khai thác được gì không, ngoài ra còn có lỗi nào khác không, có thể đọc mã nguồn ứng dụng để tìm hiểu?

Bài tập 2: Báo cáo lỗi hỏng đang được thực hành. Sử dụng format theo mẫu.

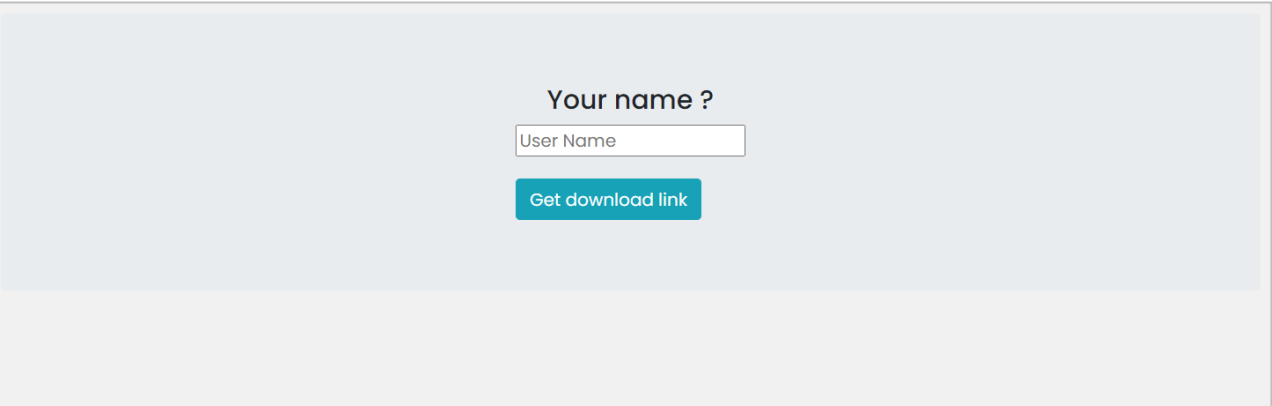
c) A08:2021 - Software and Data Integrity Failures

i. Mô tả

- Các lỗi về tính toàn vẹn của phần mềm và dữ liệu liên quan đến mã và cơ sở hạ tầng không bảo vệ chống lại các vi phạm về tính toàn vẹn.
- Điều này có thể xảy ra khi sử dụng phần mềm từ các nguồn và kho lưu trữ không đáng tin cậy hoặc thậm chí là phần mềm bị can thiệp tại nguồn, trong quá trình chuyển tiếp hoặc thậm chí là trong bộ đệm của endpoint.

ii. Kịch bản tấn công thực hành

- Kịch bản này bao gồm một trang chức năng là hiển thị trang download tài liệu cho người dùng.
- Truy cập kịch bản tại: <http://localhost:8000/2021/A8/lab2>



Chậm lại và suy nghĩ 2: Lỗi ở đây là gì, gây nên vấn đề gì đối với chức năng của web thực tế ảnh hưởng đến sự toàn vẹn của phần mềm?

Bài tập 3: Báo cáo lỗ hổng đang được thực hành. Sử dụng format theo mẫu.

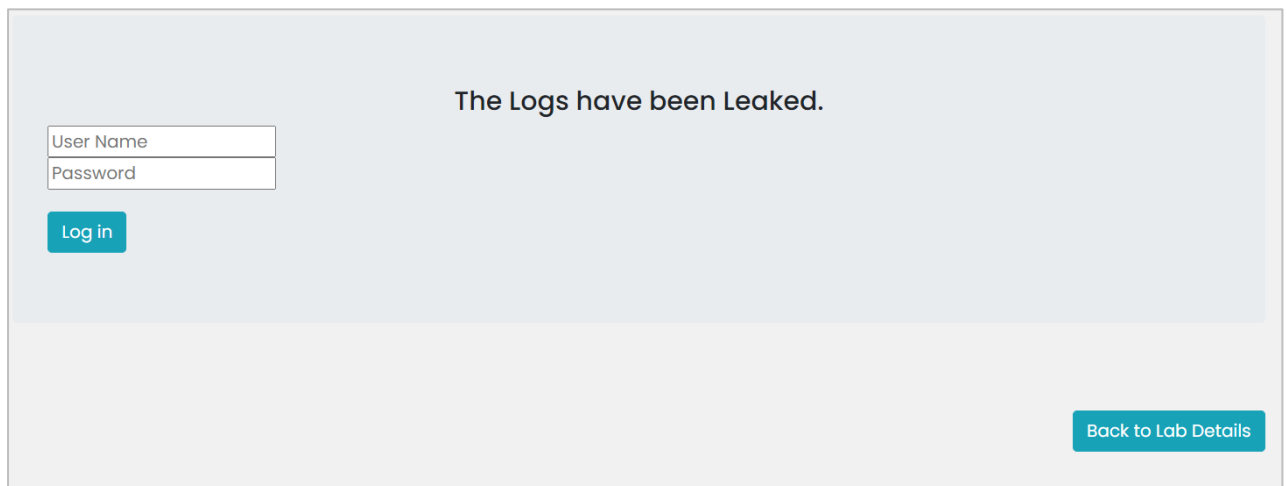
d) A09:2021 – Security Logging and Monitoring Failures

i. Mô tả

- Việc không ghi nhật ký, giám sát hoặc báo cáo đầy đủ các sự kiện bảo mật, chẳng hạn như các lần thử đăng nhập, khiến hành vi đáng ngờ khó bị phát hiện và làm tăng đáng kể khả năng kẻ tấn công có thể khai thác thành công ứng dụng.

ii. Kịch bản tấn công thực hành

- Bài thực hành này giúp hiểu được rằng đôi khi việc ghi nhật ký không đúng cách có thể dẫn đến tiết lộ thông tin. Khi truy cập vào bài thực hành, người dùng sẽ được cho một trang đăng nhập, cho biết rằng nhật ký đã bị rò rỉ. Người dùng cần tìm ra lỗ hổng và cố gắng thu thập thông tin đăng nhập đã bị rò rỉ trong các nhật ký.
- Truy cập kịch bản tại: http://localhost:8000/a10_lab



The screenshot shows a web application interface. At the top, a light blue banner contains the text "The Logs have been Leaked." Below this, there is a login form with two input fields: "User Name" and "Password". A blue "Log in" button is positioned below the password field. In the bottom right corner of the page, there is a blue button labeled "Back to Lab Details".

Chậm lại và suy nghĩ 3: Bài thực hành ghi log ở đâu, thông tin nhạy cảm có thể được tiết lộ từ vị trí nào của log?

Bài tập 4: Báo cáo lỗ hổng đang được thực hành. Sử dụng format theo mẫu.

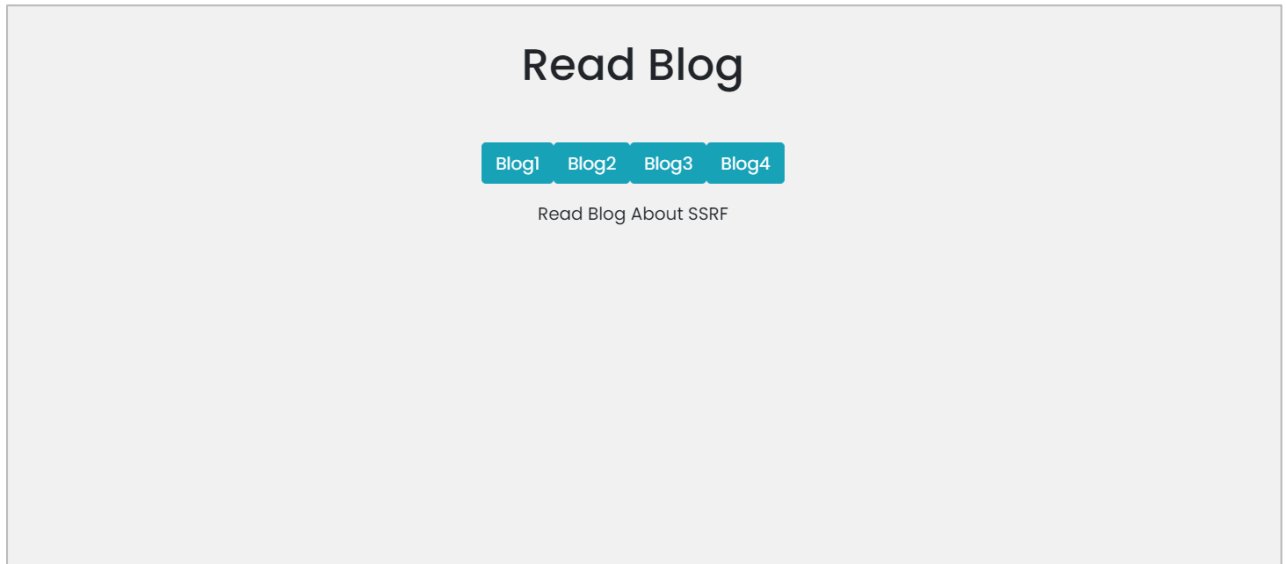
e) A10:2021 – Server-Side Request Forgery (SSRF)

i. Mô tả

- Lỗ hổng SSRF xảy ra bất cứ khi nào ứng dụng web tìm nạp tài nguyên từ xa mà không xác thực URL do người dùng cung cấp. Nó cho phép kẻ tấn công ép buộc ứng dụng gửi yêu cầu được tạo thủ công đến đích không mong muốn. Ứng dụng web dễ bị tấn công thường sẽ có đặc quyền đọc, ghi hoặc nhập dữ liệu bằng URL.
- Để thực hiện một cuộc tấn công SSRF, kẻ tấn công lạm dụng chức năng trên máy chủ để đọc hoặc cập nhật tài nguyên nội bộ. Sau đó kẻ tấn công có thể buộc ứng dụng gửi yêu cầu truy cập các tài nguyên ngoài ý muốn.

ii. Kịch bản tấn công thực hành

- Kịch bản này giúp cho sinh viên hiểu được SSRF có thể gây ra lỗi bảo mật như thế nào. Bài lab cung cấp giao diện một trang web cung cấp các bài blog cho mọi người xem.
- Truy cập kịch bản tại: http://localhost:8000/ssrf_lab



Chậm lại và suy nghĩ 4: Vị trí lỗ hổng ở đâu, khai thác lỗi này như thế nào?

Bài tập 5: Báo cáo lỗ hổng đang được thực hành. Sử dụng format theo mẫu.

2. Bài tập thực hành

1. http://localhost:8000/a9_lab2
2. http://localhost:8000/insec_des_lab
3. http://localhost:8000/ssrf_lab2
4. http://localhost:8000/ssrf_discussion
5. <https://portswigger.net/web-security/authentication/other-mechanisms/lab-password-brute-force-via-password-change>
6. <https://portswigger.net/web-security/authentication/password-based/lab-username-enumeration-via-different-responses>
7. <https://portswigger.net/web-security/authentication/password-based/lab-username-enumeration-via-response-timing>
8. <https://portswigger.net/web-security/authentication/multi-factor/lab-2fa-simple-bypass>
9. <https://portswigger.net/web-security/clickjacking/lab-exploiting-to-trigger-dom-based-xss>
10. <https://portswigger.net/web-security/request-smuggling/exploiting/lab-deliver-reflected-xss>
11. <https://portswigger.net/web-security/server-side-template-injection/exploiting/lab-server-side-template-injection-basic>
12. <https://portswigger.net/web-security/deserialization/exploiting/lab-deserialization-modifying-serialized-objects>
13. <https://portswigger.net/web-security/information-disclosure/exploiting/lab-infoleak-via-backup-files>

D. YÊU CẦU & ĐÁNH GIÁ

- Sinh viên tìm hiểu và thực hành theo hướng dẫn, thực hiện **theo nhóm đã đăng ký**.
- Nộp báo cáo kết quả gồm **Code, CSDL được export** và chi tiết những việc (**Report**) mà nhóm đã thực hiện, quan sát thấy và kèm ảnh chụp màn hình kết quả (nếu có); giải thích cho quan sát (nếu có).

Báo cáo:

- File **.PDF**. Tập trung vào nội dung, không mô tả lý thuyết.
- Đặt tên theo định dạng: **[Mã lớp]-LabX_MSSV1-MSSV2-MSSV3**.
Ví dụ: *[NT213.K11.ANTN.1]-Lab1_1852xxxx-1852yyyy-1852zzzz*.
- Nếu báo cáo có nhiều file, nén tất cả file vào file .ZIP với cùng tên file báo cáo.
- Nộp file báo cáo trên theo thời gian đã thống nhất tại courses.uit.edu.vn.

Bài sao chép, trễ, ... sẽ được xử lý tùy mức độ vi phạm.

E. GIẢI ĐÁP MẪU CÁC CÂU HỎI CHẬM LẠI VÀ SUY NGHĨ

HẾT

Chúc các bạn hoàn thành tốt!