# BÁO CÁO ĐỒ ÁN

## Môn học: Bảo mật web và ứng dụng

## Đề tài: Tìm hiểu công cụ Acunetix

**GVHD: ThS. Nguyễn Công Danh**

Nhóm4

21520911 - Bùi Quốc Huy

21520942 - Võ Quốc Huy

21520836 - Phạm Phúc Hậu

21521277 - Nguyễn Minh Phú

# NỘI DUNG BÁO CÁO

1. Giới thiệu đề tài
2. Tổng quan về Acunetix
3. Thực nghiệm
4. Kết luận và đánh giá

# 1. Giới thiệu đề tài

**Đặt vấn đề**

Bảo mật trang web là vấn đề nên là ưu tiên hàng đầu trong bất kỳ tổ chức nào. Việc có thể truy cập trang web mọi lúc và từ mọi nơi giúp tin tặc thực hiện các cuộc tấn công tập trung vào các ứng dụng web như giỏ hàng, biểu mẫu, trang đăng nhập, nội dung động, ...

**=>** Việc sử dụng Acunetix cho kiểm thử bảo mật web để vá các lỗ hổng sẽ giảm thiểu được tình trạng trên.

# NỘI DUNG BÁO CÁO

1. Giới thiệu đề tài
2. Tổng quan về Acunetix
3. Thực nghiệm
4. Kết luận và đánh giá

# 2. Tổng quan về Acunetix

**Acunetix là gì?**

Acunetix là một công cụ kiểm tra bảo mật ứng dụng web tự động giúp kiểm tra các ứng dụng web của ta bằng cách kiểm tra các lỗ hổng như SQL Insert, Cross-site scripting và các lỗ hổng có thể khai thác khác. Nói chung, Acunetix quét bất kỳ trang web hoặc ứng dụng web nào có thể truy cập được qua trình duyệt web và sử dụng giao thức HTTP/HTTPS.

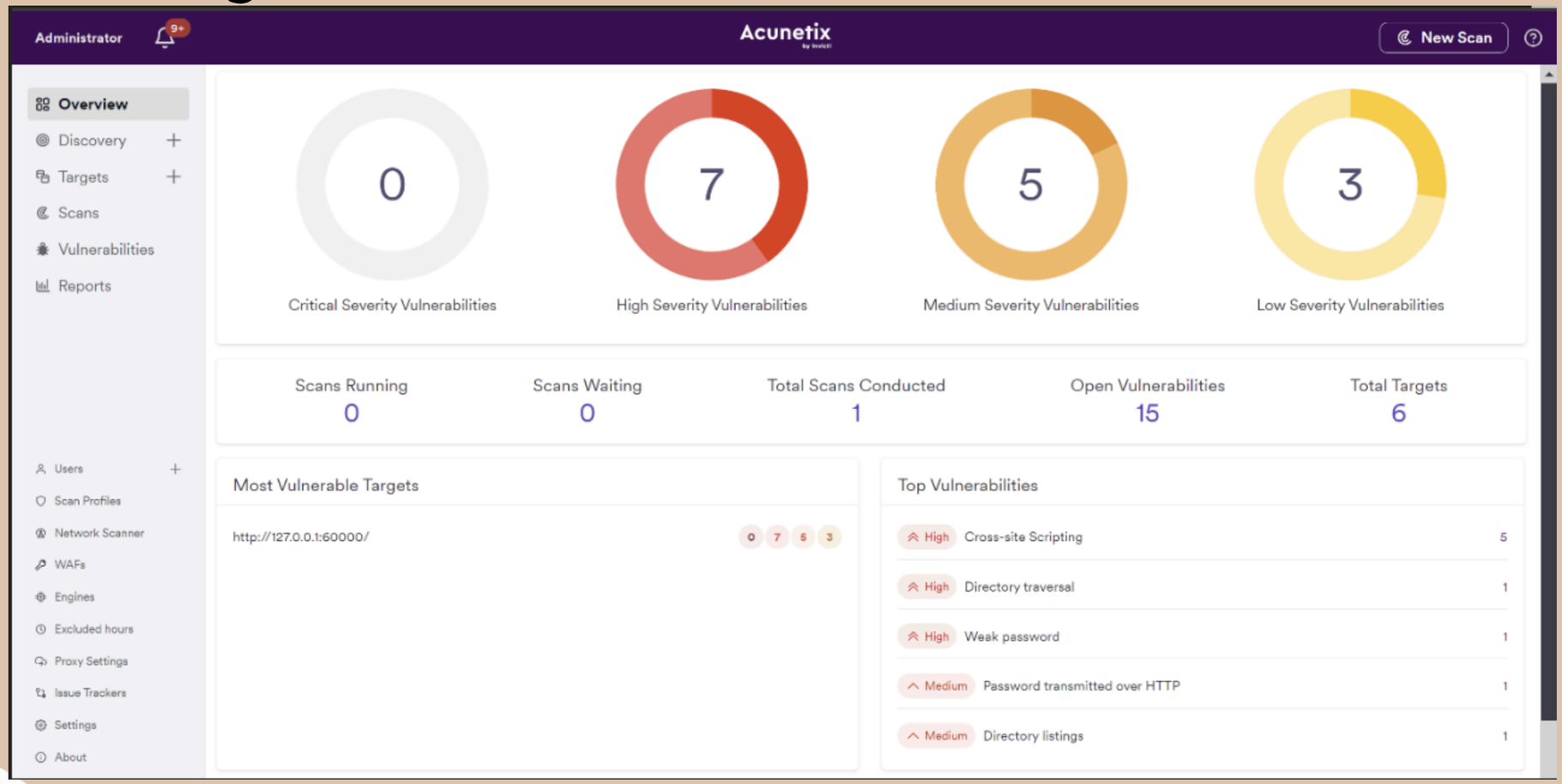
# 2. Tổng quan về Acunetix

**Các chức năng của Acunetix**

Dựa trên giao diện web của Acunetix, nhóm đưa ra các nhóm chức năng: Tổng quan, Khám phá, Mục tiêu, Quét, Lỗ hổng, Báo cáo, Người dùng, Hồ sơ quét, WAFs, Engines, Excluded Hours, Agents, Proxy Settings, Issue Trackers, Cài đặt.

# 2. Tổng quan về Acunetix

## Các chức năng của Acunetix



# 2. Tổng quan về Acunetix

**Acunetix Manual Tools là gì?**

Acunetix Manual Tools (AMT) không phải là ứng dụng kiểm thử ứng dụng web tự động mà là một bộ công cụ kiểm thử xâm nhập thủ công miễn phí được cấp bởi Invicti.

# 2. Tổng quan về Acunetix

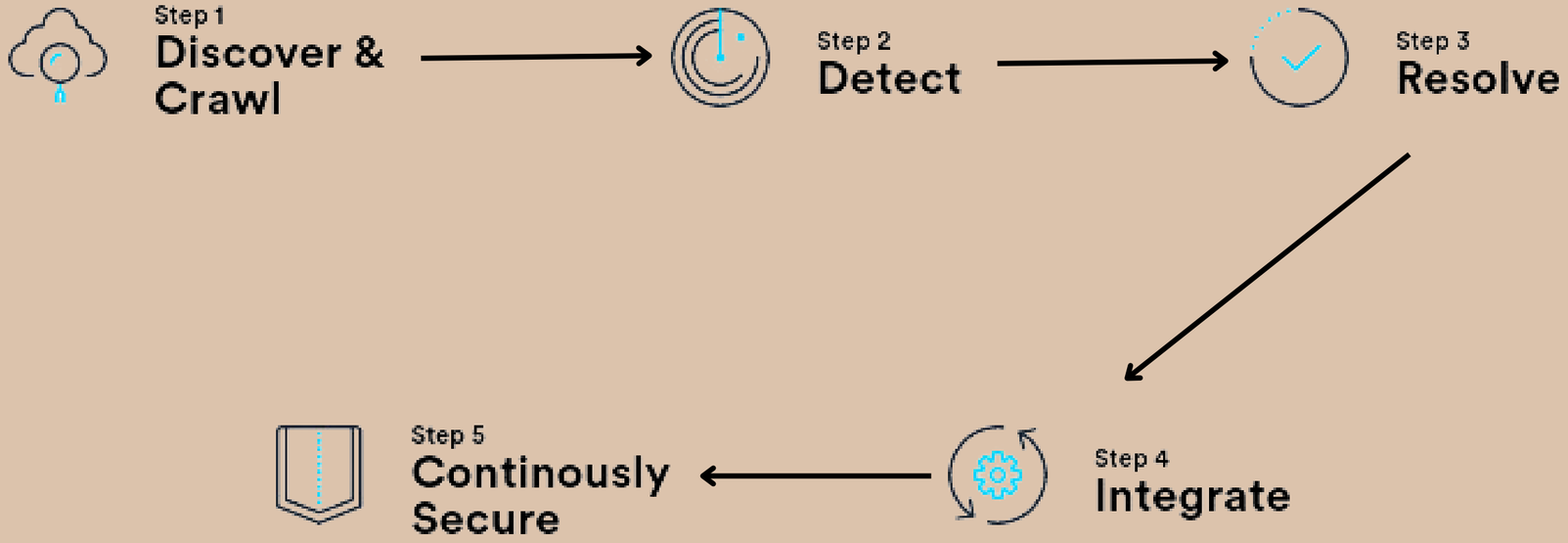
**Các chức năng của Acunextix Manual Tools**

Acunetix Manual Tools gồm 8 mô-đun:

- HTTP Editor
- Subdomain Scanner
- Target Finder
- Blind SQL Injector
- HTTP Fuzzer
- Authentication Tester
- Web Services Editor
- HTTP Sniffer.

# 2. Tổng quan về Acunetix

## Cách Acunetix thực hiện quét trang web



# NỘI DUNG BÁO CÁO

1. Giới thiệu đề tài
2. Tổng quan về Acunetix
3. Thực nghiệm
4. Kết luận và đánh giá

# 3. Thực nghiệm

**Công cụ thực hiện**

Nhóm sử dụng Acunetix Premium v24.4.240514098 bản CRACK và Acunetix Manual Tools để thực hiện các kịch bản mà nhóm đề ra.

# 3. Thực nghiệm

## Các kịch bản thực hiện

**Kịch bản 1:** Thực hiện Quét toàn phần trang web http://testphp.vulnweb.com/

**Kịch bản 2:** Thực hiện Quét toàn phần ứng dụng web juice-shop với tính năng Site Login.

**Kịch bản 3:** Sử dụng AMT để giải tìm flag CTF của challenge HTTP – Cookies của Root Me. Link challenge: HTTP – Cookies.

**Kịch bản 4:** Sử dụng AMT để giải tìm flag CTF của challenge HTTP – POST của Root Me. Link challenge: HTTP – POST.

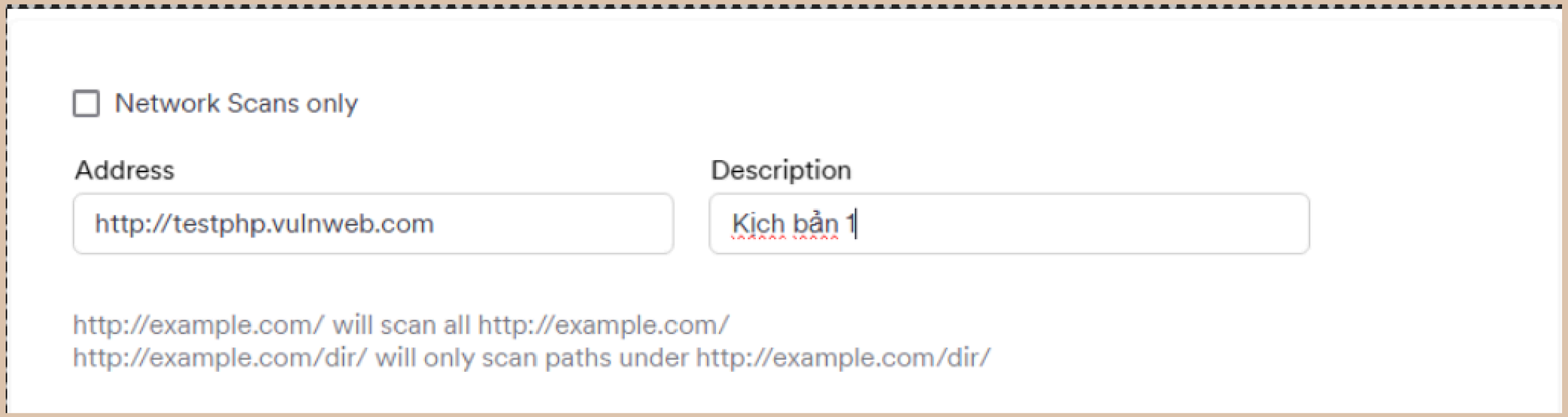
**Kịch bản 5:** Cài đặt AcuSensor và quét một trang web nội bộ được xây dựng bằng PHP.

**Kịch bản 6:** Thử nghiệm tính năng Issue Tracker.

# 3. Thực nghiệm

**Kịch bản 1**

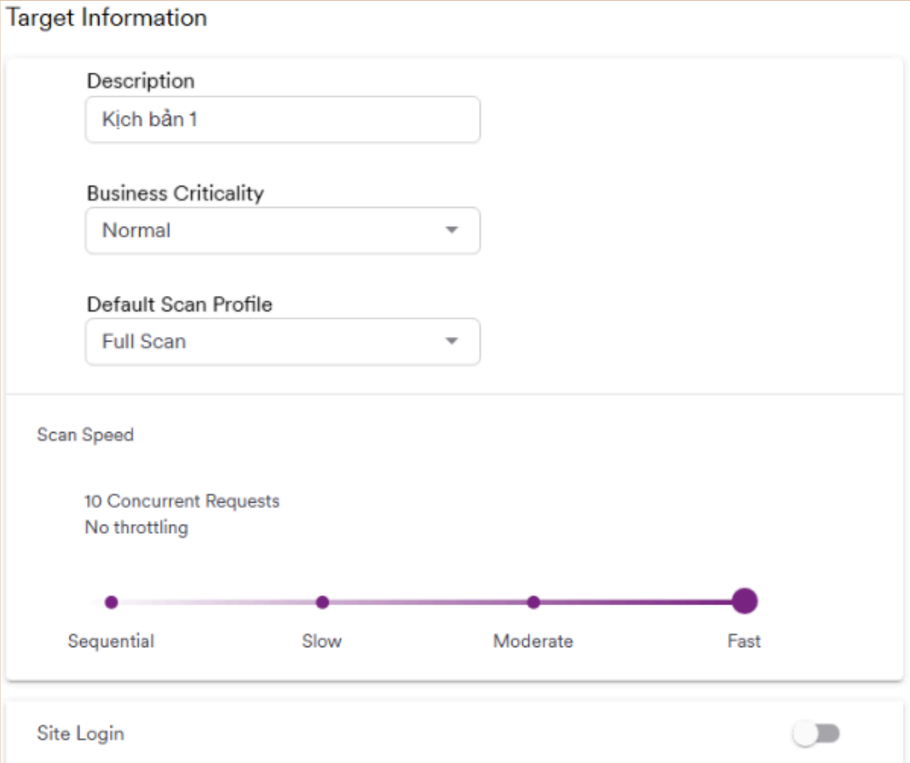
Ta tiến hành tạo Mục tiêu, ở trường Address ta nhập url của web.



# 3. Thực nghiệm

**Kịch bản 1**

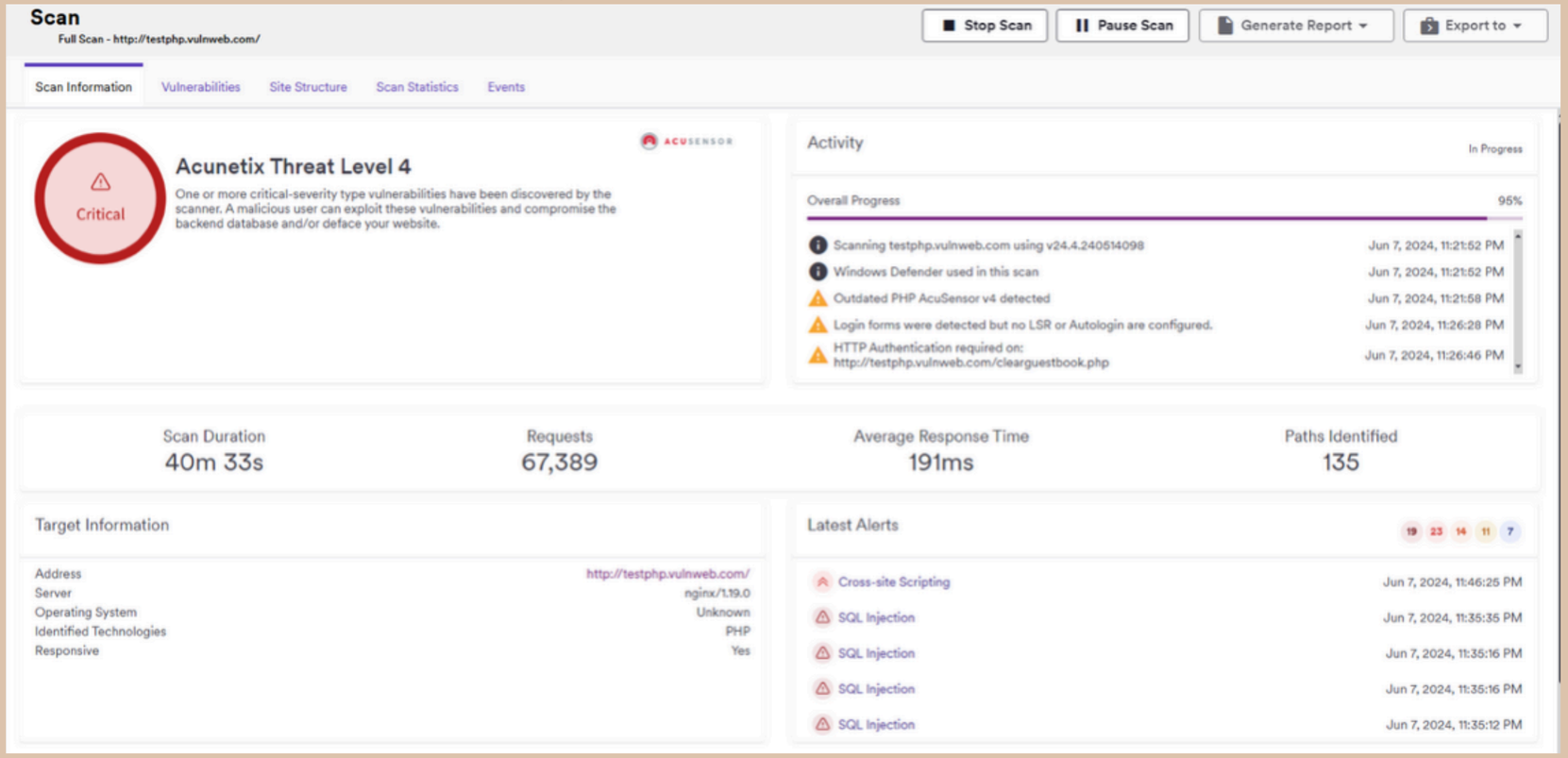
Ta để cấu hình Mục tiêu theo mặc định và tiến hành quét.



# 3. Thực nghiệm

## Kịch bản 1

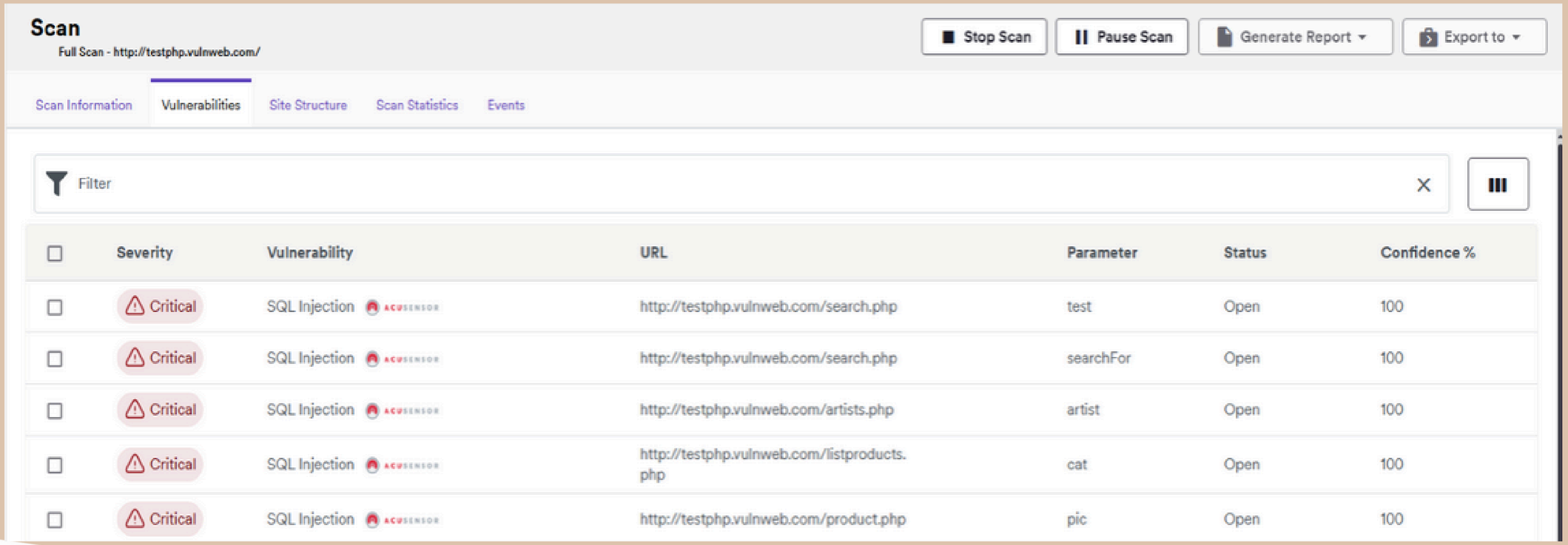
Kết quả thu được sau khi quét.



# 3. Thực nghiệm

## Kịch bản 1

Danh sách các lỗ hổng quét được của mục tiêu.

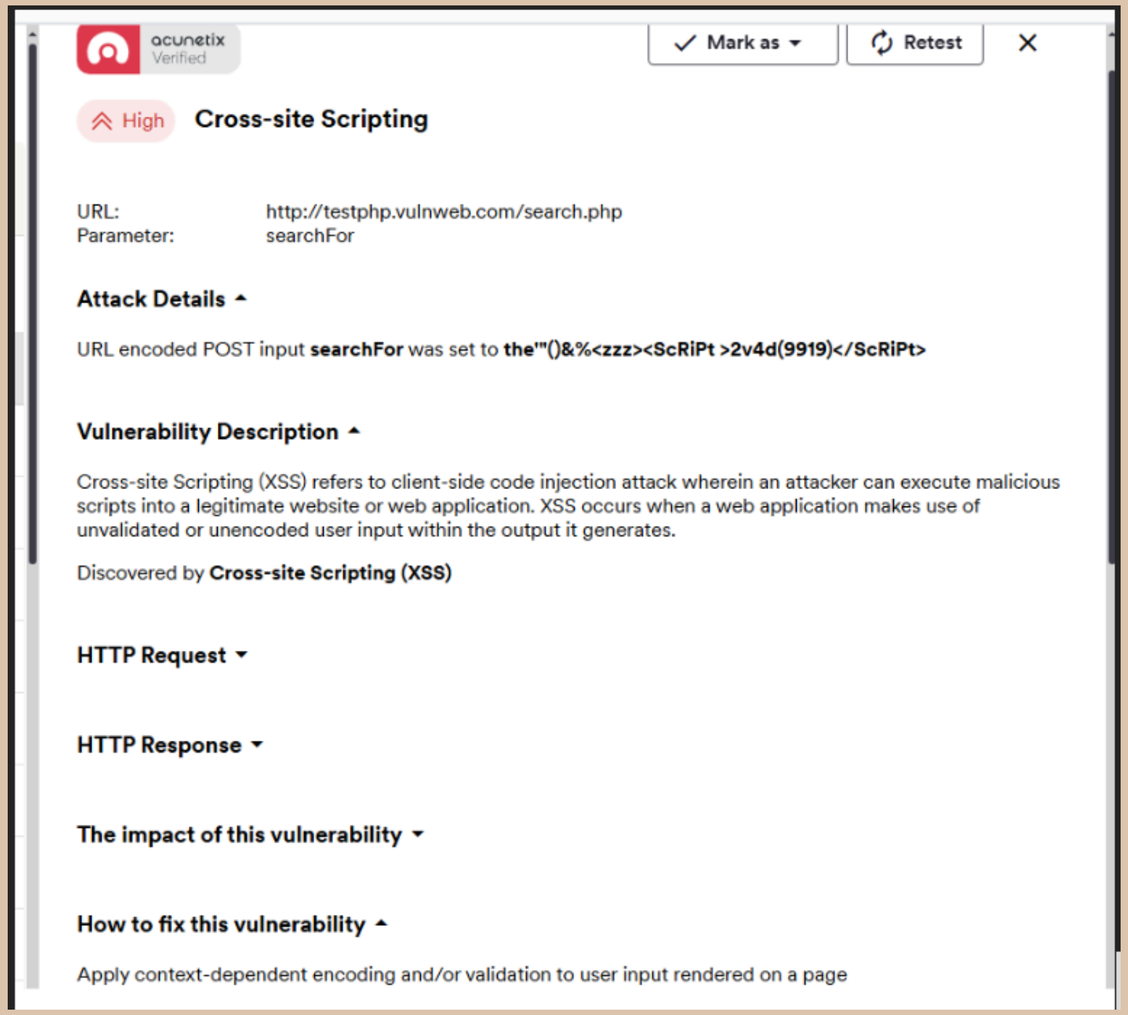



| Severity | Vulnerability | URL | Parameter | Status | Confidence % |
|---|---|---|---|---|---|
| Critical | SQL Injection | http://testphp.vulnweb.com/search.php | test | Open | 100 |
| Critical | SQL Injection | http://testphp.vulnweb.com/search.php | searchFor | Open | 100 |
| Critical | SQL Injection | http://testphp.vulnweb.com/artists.php | artist | Open | 100 |
| Critical | SQL Injection | http://testphp.vulnweb.com/listproducts.php | cat | Open | 100 |
| Critical | SQL Injection | http://testphp.vulnweb.com/product.php | pic | Open | 100 |

# 3. Thực nghiệm

**Kịch bản 1**

Chi tiết lỗ hổng.



# 3. Thực nghiệm

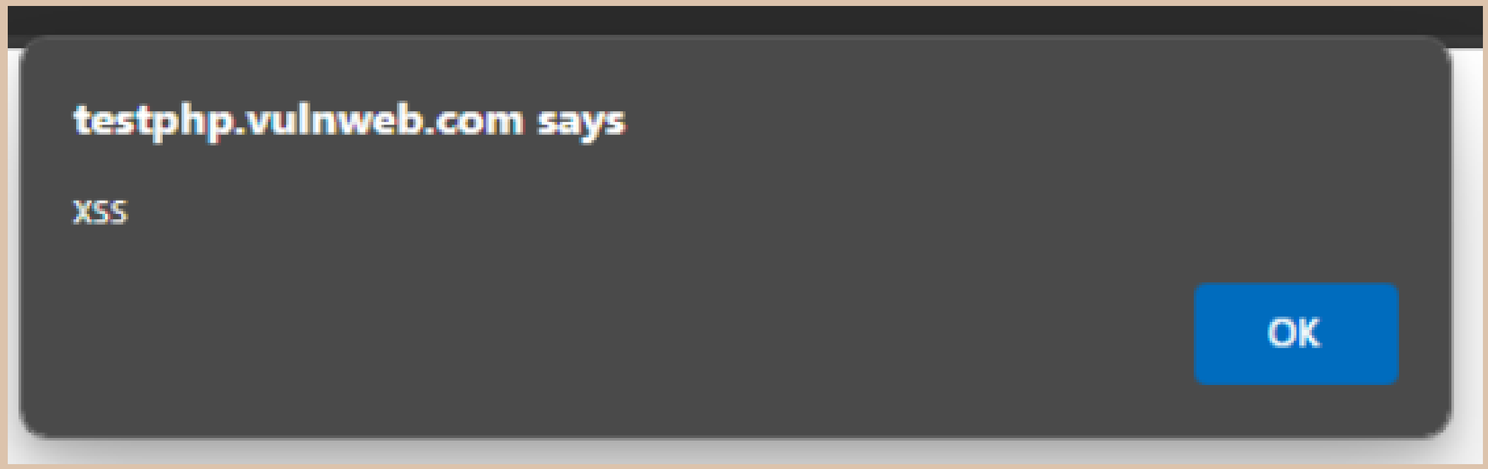
**Kịch bản 1**

Thử tiến hành khai thác lỗ hổng.

# 3. Thực nghiệm

**Kịch bản 1**

Kết quả khai thác.



# 3. Thực nghiệm

**Kịch bản 2**

Trước khi thực hiện kịch bản, cần cài đặt OWASP Juice Shop trước.

- Cài đặt đơn giản: Sử dụng docker, chạy 2 lệnh sau:
- Để tải image về: docker pull bkimminich/juice-shop
- Để chạy container: docker run --rm -p 3000:3000 bkimminich/juice-shop
- Sau khi chạy container xong thì ta có thể truy cập juice-shop qua đường dẫn: http://localhost:3000/

# 3. Thực nghiệm

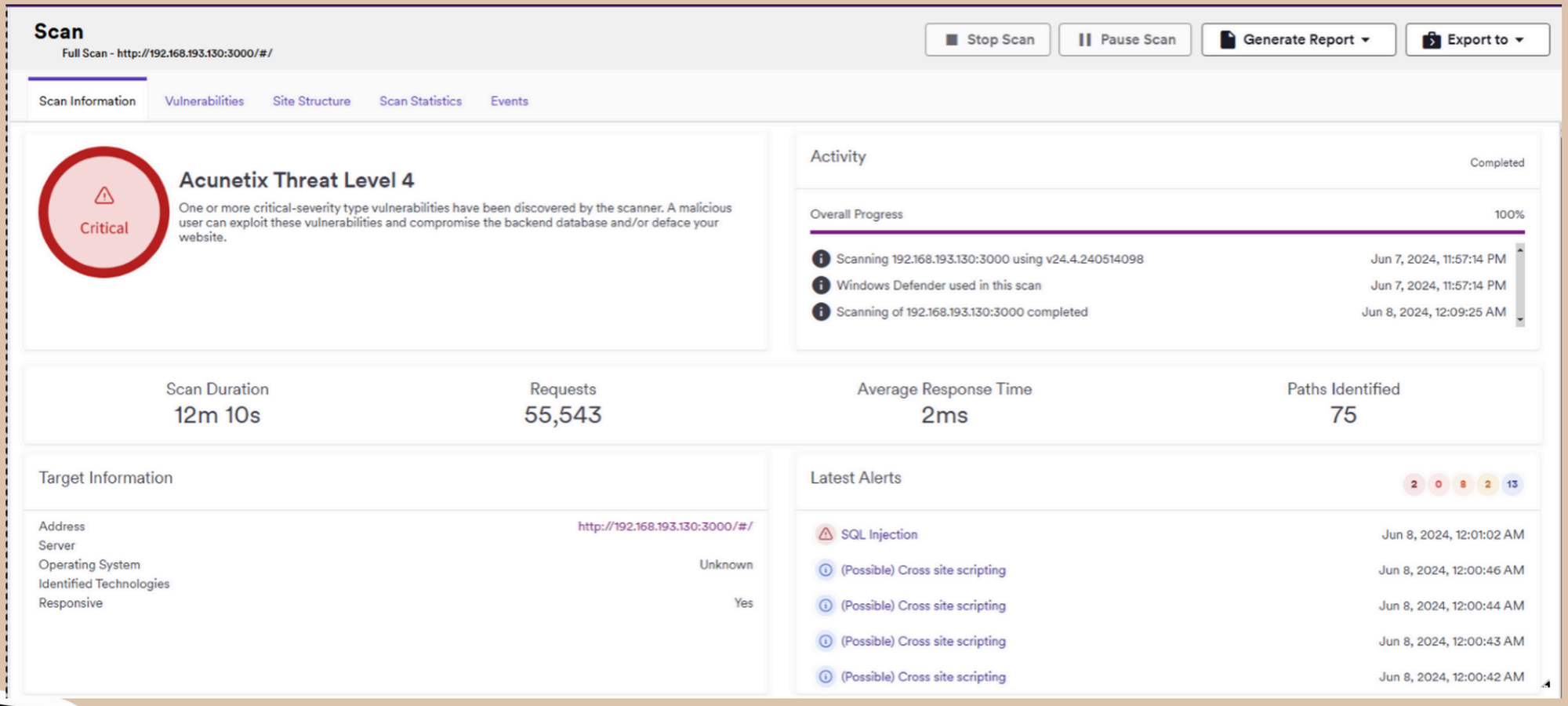
**Kịch bản 2**

Khi thực hiện kịch bản này, nhóm thực hiện 2 lần quét để so sánh hiệu quả của việc quét có sử dụng Site Login.

**Lần quét đầu tiên**, nhóm không dùng Site Login và tiến hành quét juice shop. Acunetix không phát hiện được lỗ hổng nào ở mức nghiêm trọng High.

# 3. Thực nghiệm

## Kịch bản 2



# 3. Thực nghiệm

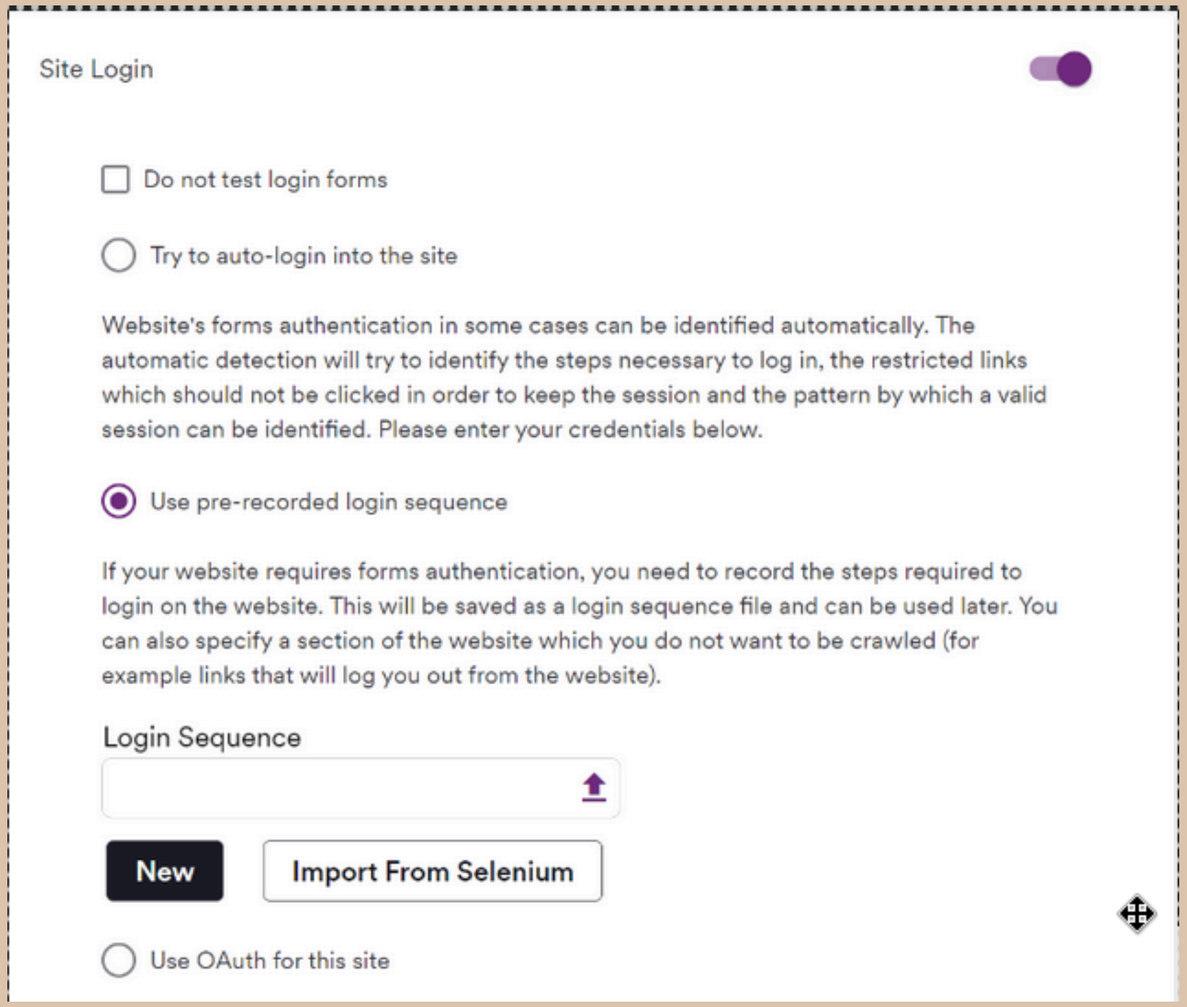
**Kịch bản 2**

**Lần quét thứ 2**, nhóm sử dụng Site Login và tiến hành scan lại. Ở Site login, nhóm dùng “trình tự đăng nhập được ghi trước” (Use pre-recorded login sequence) để đảm bảo Acunetix có thể quét được các vùng cần đăng nhập của juice shop.

- Ở “Use pre-recorded login sequence” Chọn "New" để tạo trình tự đăng nhập mới cho Site Login trên Acunetix.

# 3. Thực nghiệm

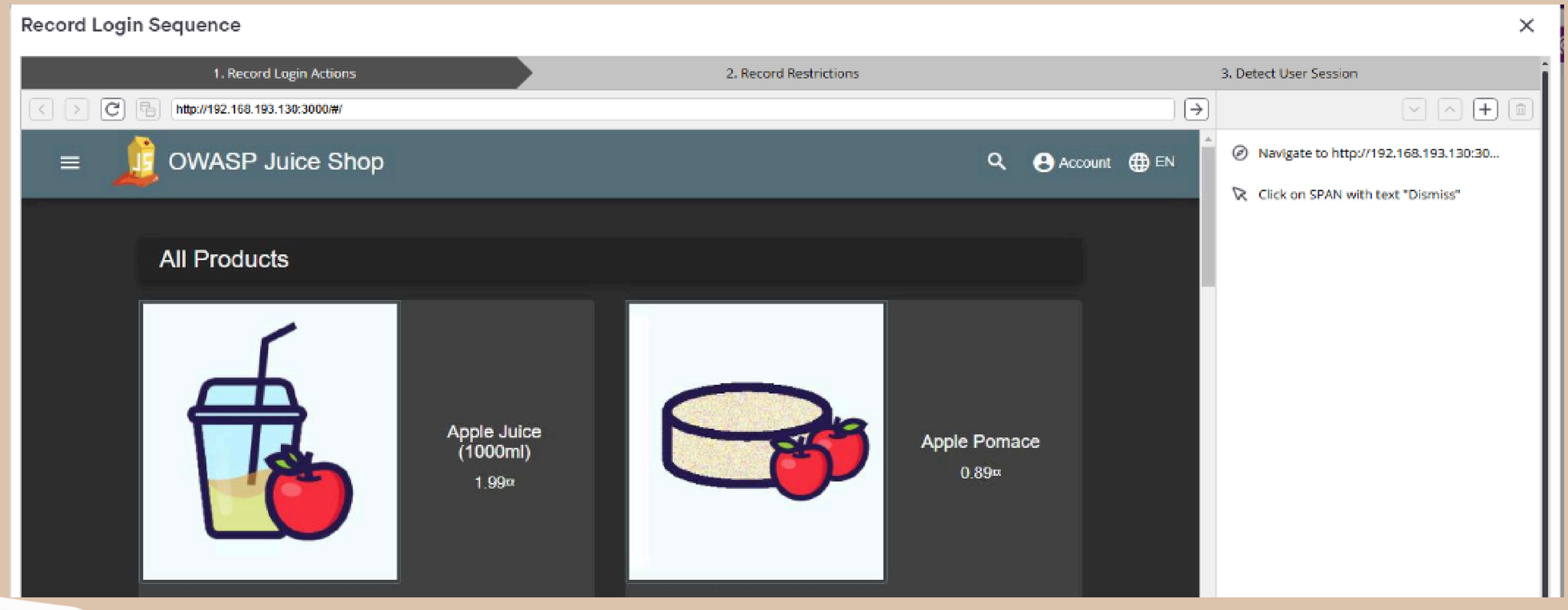
## Kịch bản 2



# 3. Thực nghiệm

## Kịch bản 2

- Ghi lại trình tự đăng nhập cho Site Login trên Acunetix.



# 3. Thực nghiệm

**Kịch bản 2**

Kết quả: Acunetix phát hiện được thêm nhiều lỗ hổng ở mức nghiêm trọng High ở các vùng cần đăng nhập.

Scan Information | Vulnerabilities | Site Structure | Scan Statistics | Events

Filter

| Severity | Vulnerability | URL | Parameter | Status | Confidence % |
|---|---|---|---|---|---|
| Critical | SQL Injection | http://192.168.193.130:3000/rest/products/search | q | Open | 95 |
| Critical | SQL Injection | http://192.168.193.130:3000/rest/user/login | email | Open | 95 |
| High | API Sensitive Info(PII) accessible without authentication | http://192.168.193.130:3000/rest/products/1/reviews | | Open | 80 |
| High | API Sensitive Info(PII) accessible without authentication | http://192.168.193.130:3000/rest/products/37/reviews | | Open | 80 |
| High | API Sensitive Info(PII) accessible without authentication | http://192.168.193.130:3000/rest/products/43/reviews | | Open | 80 |

# 3. Thực nghiệm

**Kịch bản 2**

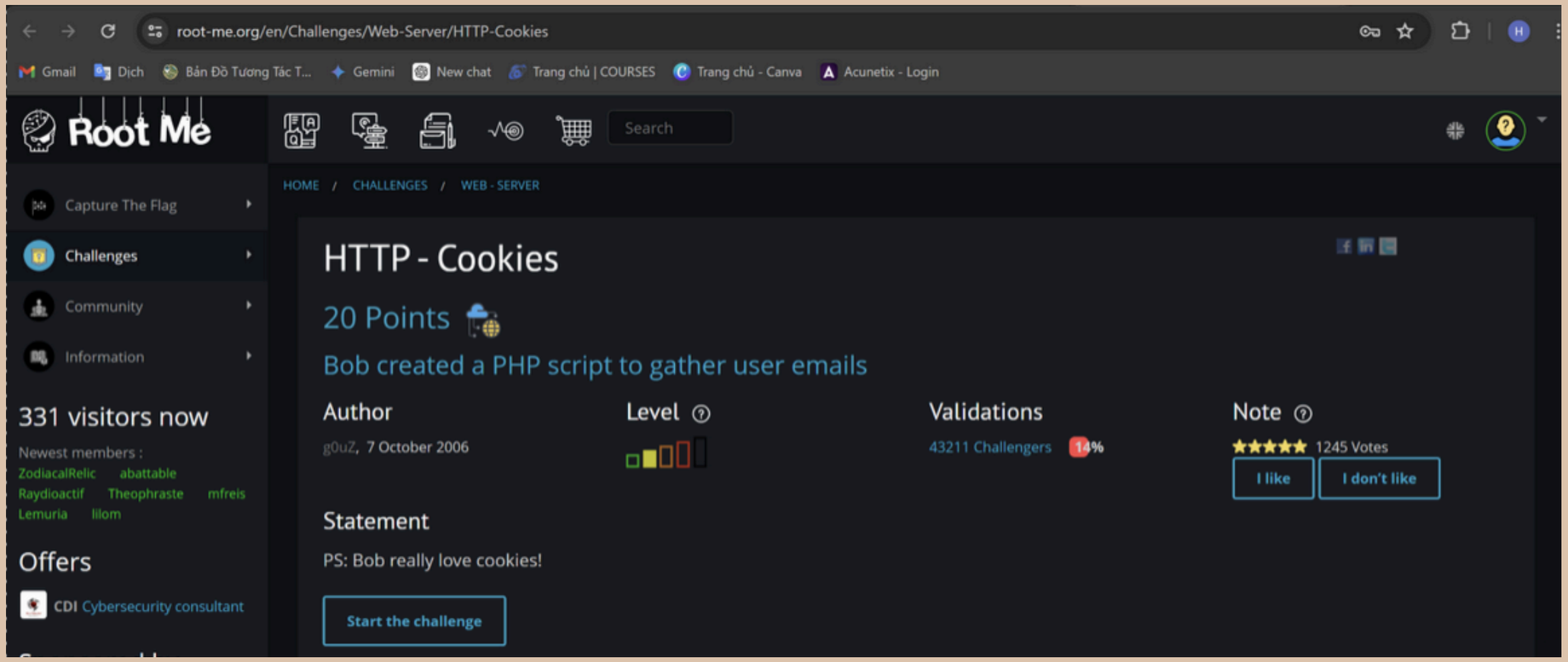
So sánh tổng quan giữa có và không dùng Site Login để quét juice shop.

| Target Description | Scan Profile | Schedule | Vulnerabilities |
|---|---|---|---|
| juice-shop with login sequence | Full Scan | Last run on Jun 8, 2024, 12:20:01 PM | 2 12 8 3 15 |
| juice-shop | Full Scan | Last run on Jun 7, 2024, 11:57:09 PM | 2 0 8 2 13 |

# 3. Thực nghiệm

## Kịch bản 3

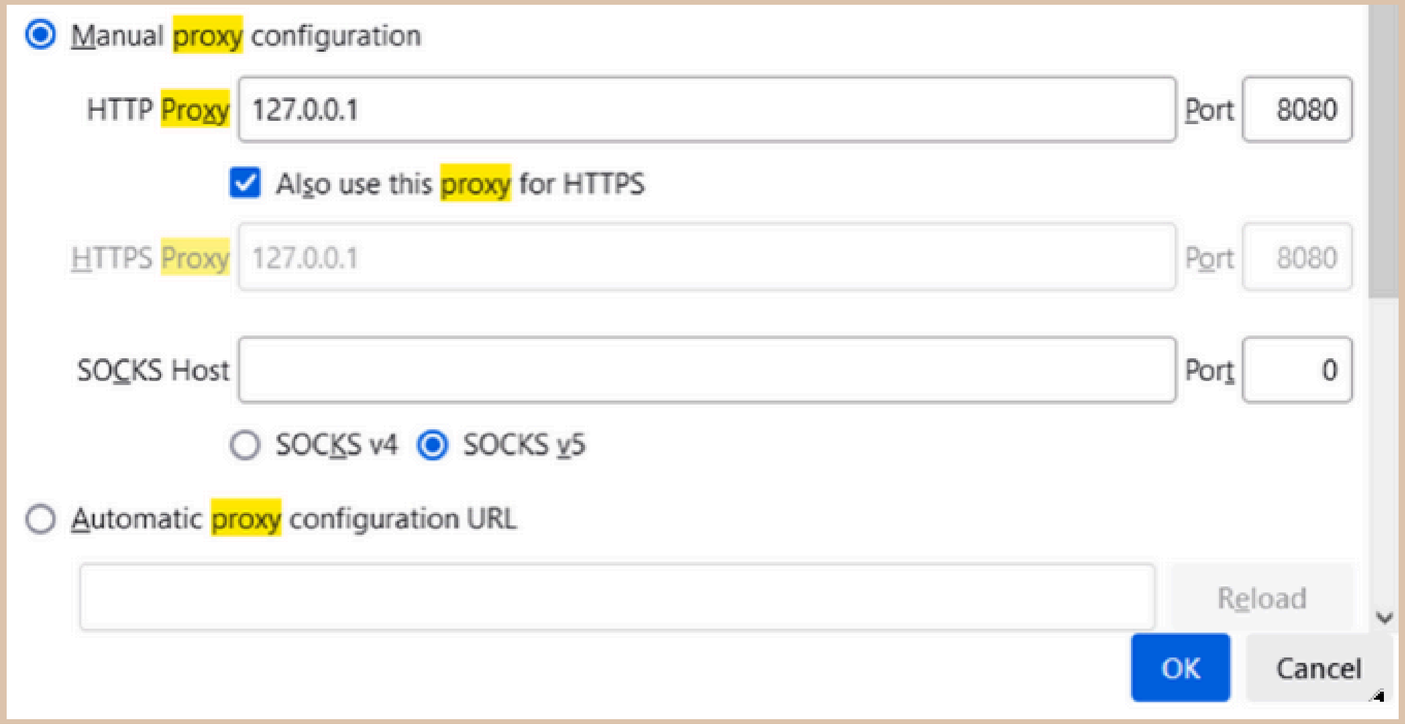
Challenge HTTP – Cookies của Root Me.



# 3. Thực nghiệm

**Kịch bản 3**

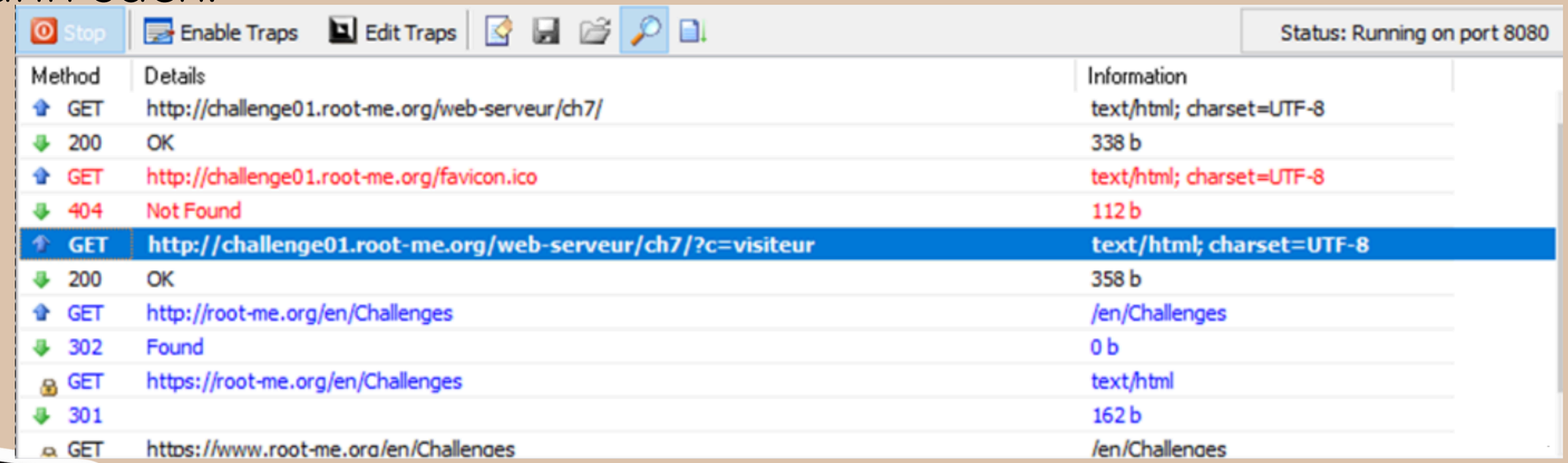
Thực hiện cấu hình proxy như sau trên trình duyệt để tạo điều kiện cho HTTP Sniffer của AMT làm việc.



# 3. Thực nghiệm

**Kịch bản 3**

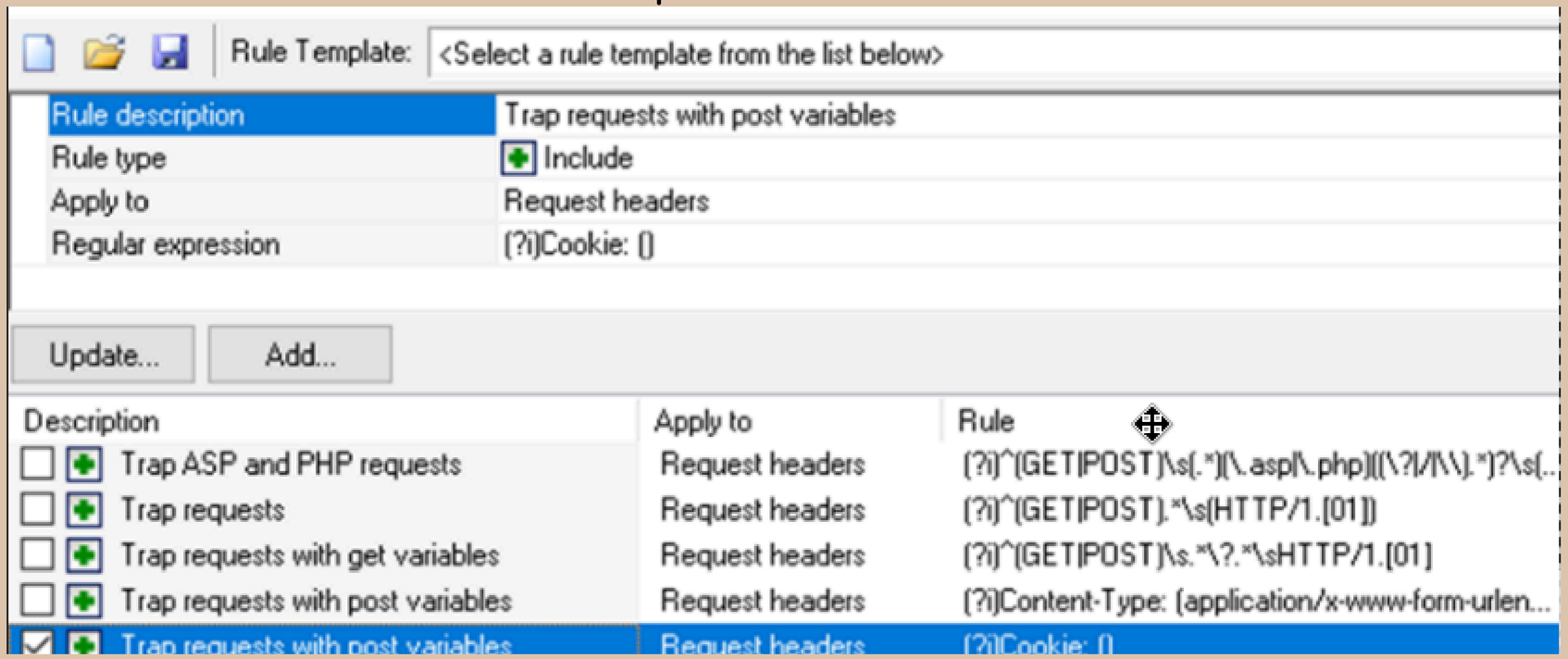
Click “Start” ở trang HTTP Sniffer của AMT.Các gói tin yêu cầu và phản hồi trên trình duyệt sẽ được “đánh hơi” và ghi nhận thành một danh sách.



# 3. Thực nghiệm

**Kịch bản 3**

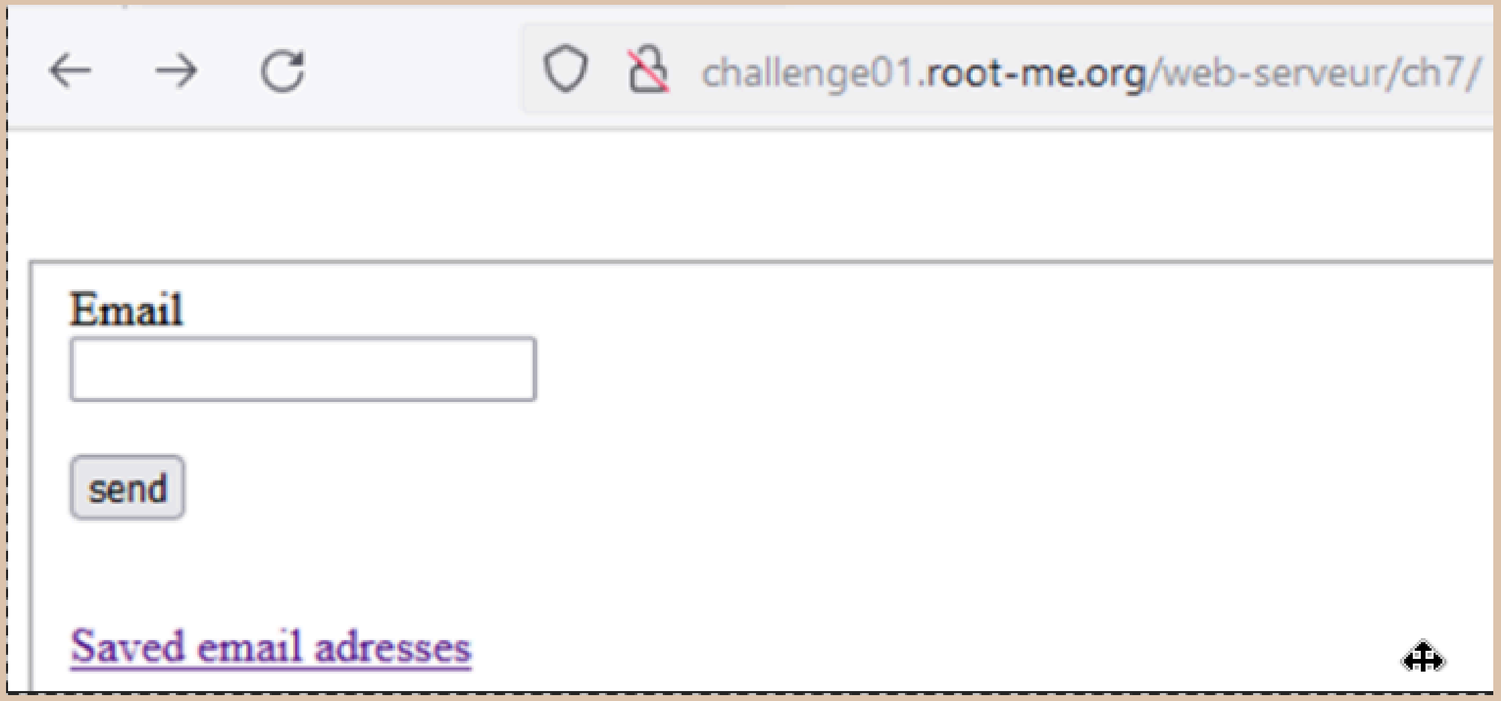
Thực hiện chặn các gói tin POST có bao gồm trường Cookie bằng cách thêm rule vào HTTP Traps của HTTP Sniffer.



# 3. Thực nghiệm

**Kịch bản 3**

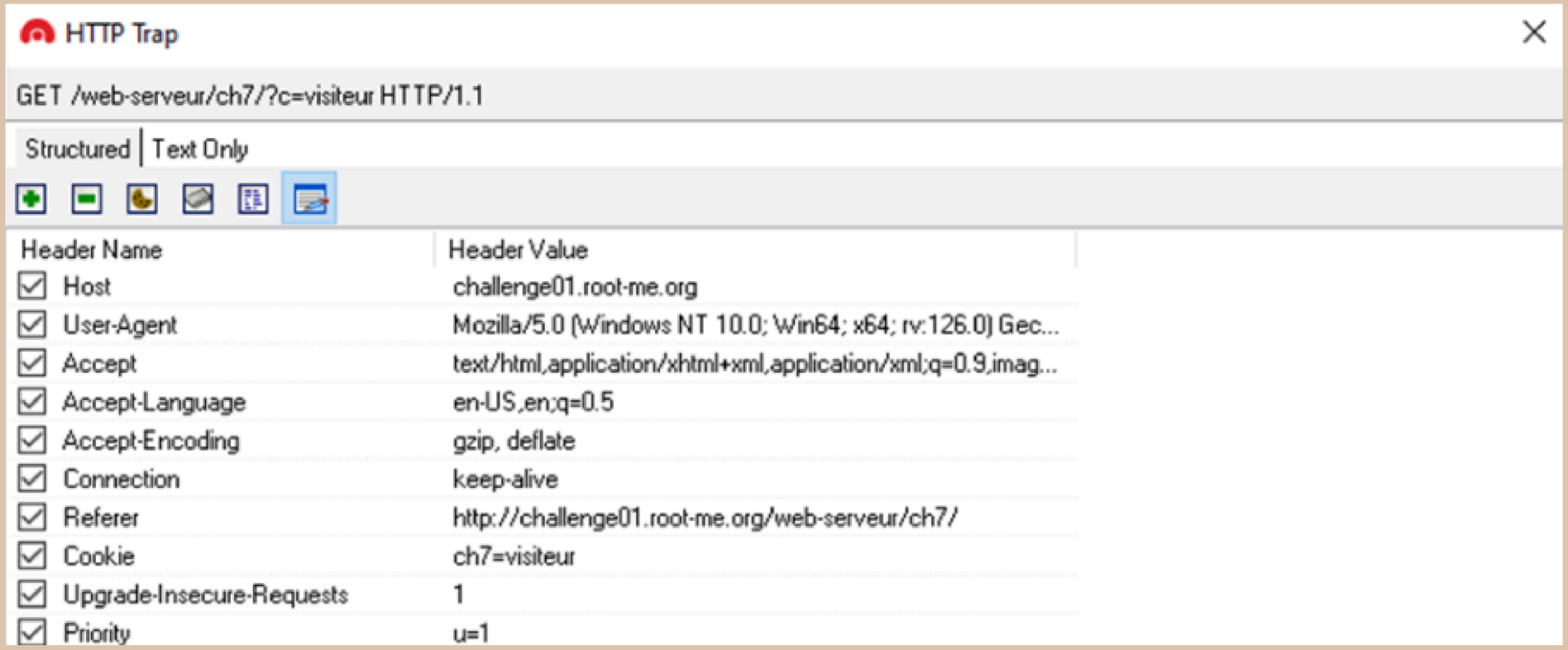
Start challenge và phân tích yêu cầu POST được gửi đi khi click "Saved email adresses".



# 3. Thực nghiệm

**Kịch bản 3**

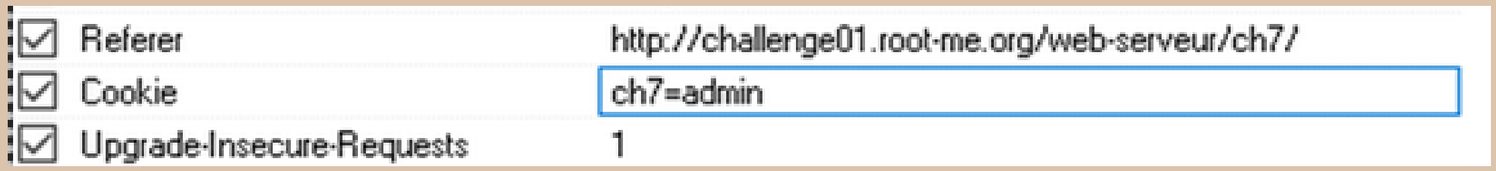
HTTP Sniffer chặn được yêu cầu POST này, yêu cầu này có trường Cookie.



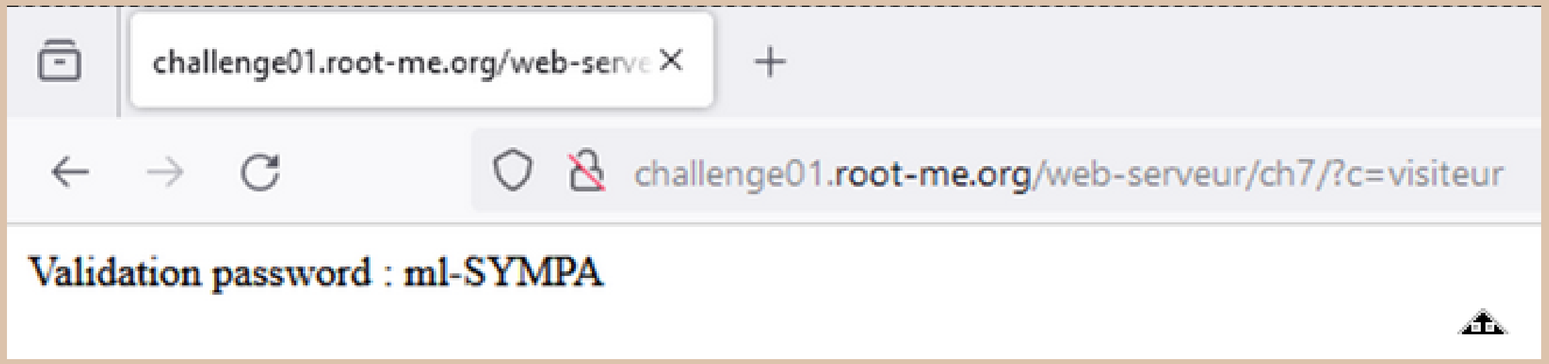
# 3. Thực nghiệm

**Kịch bản 3**

Chỉnh sửa trường Cookie “ch7=visiteur” thành “ch7=admin” và chuyển đi.



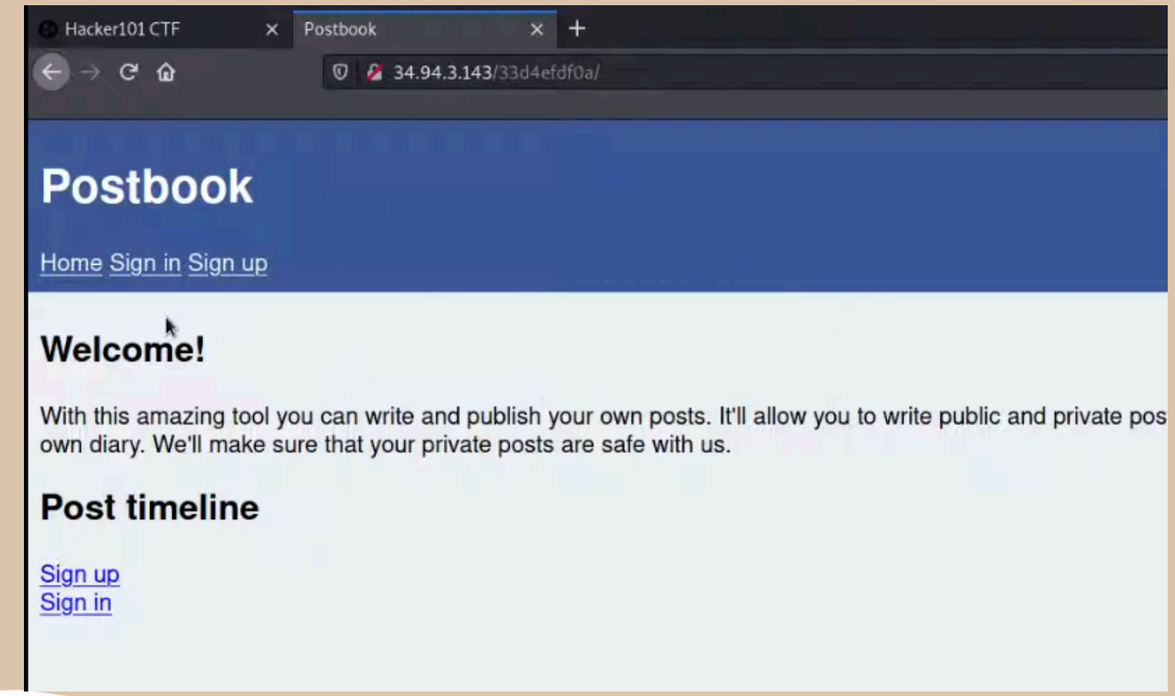

Sau khi chuyển yêu cầu, trang web của challenge trên trình duyệt trả về nội dung của flag .

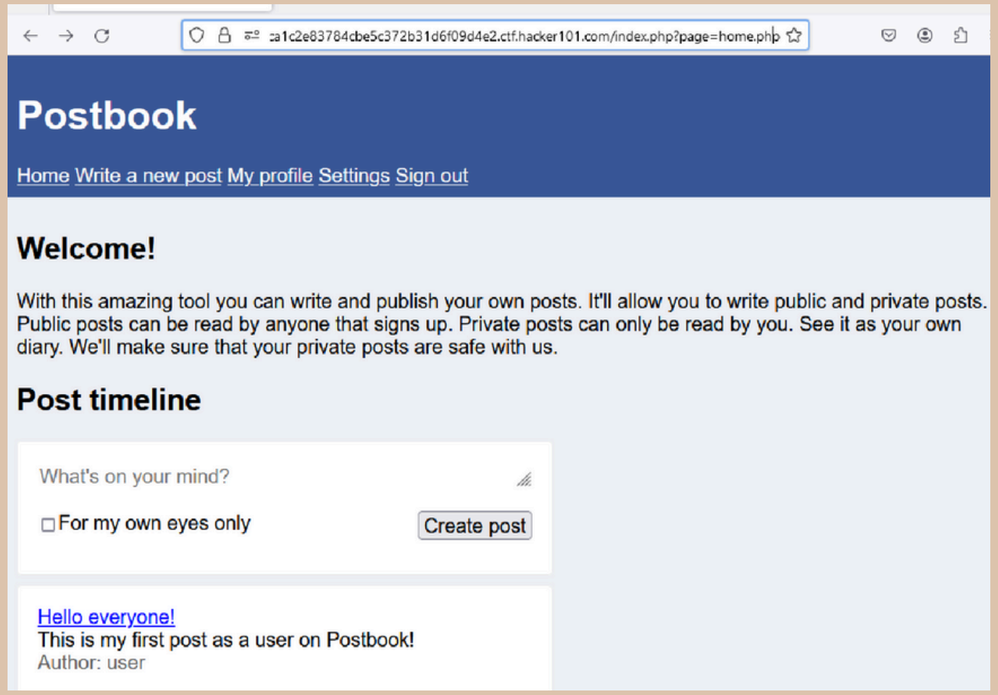


# 3. Thực nghiệm

## Kịch bản 4

Challenge ctf.hack101.com – Postbook

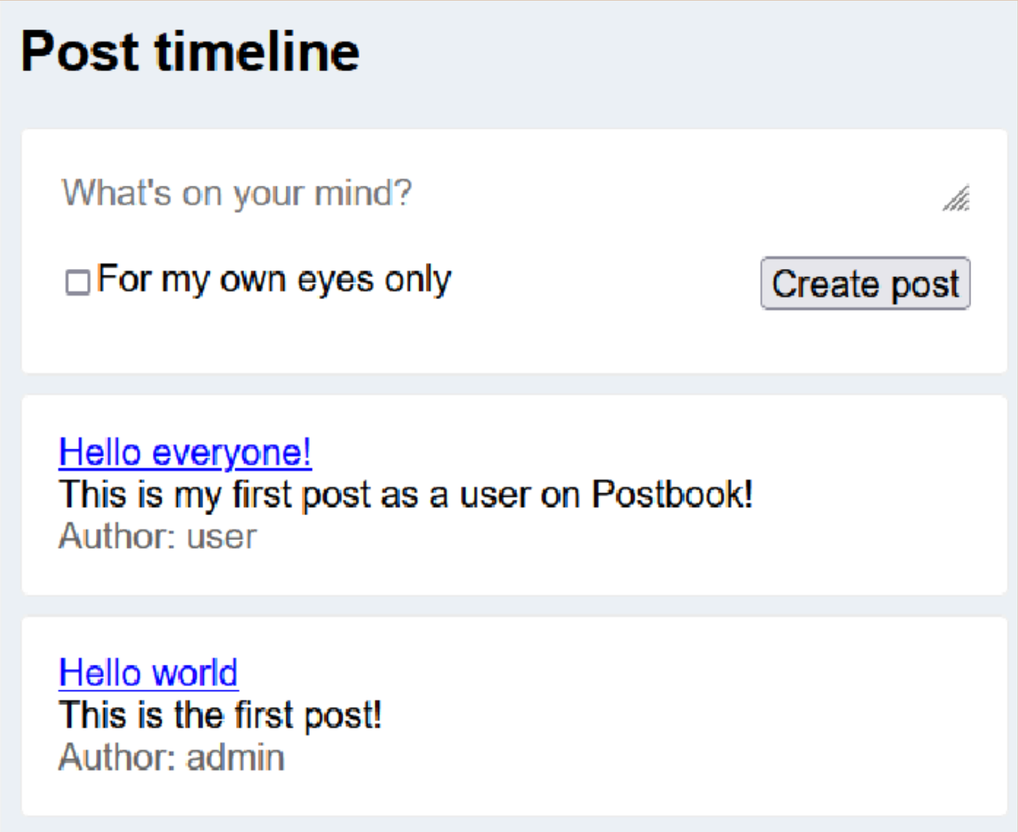


# 3. Thực nghiệm

**Kịch bản 4**

Challenge ctf.hack101.com – Postbook

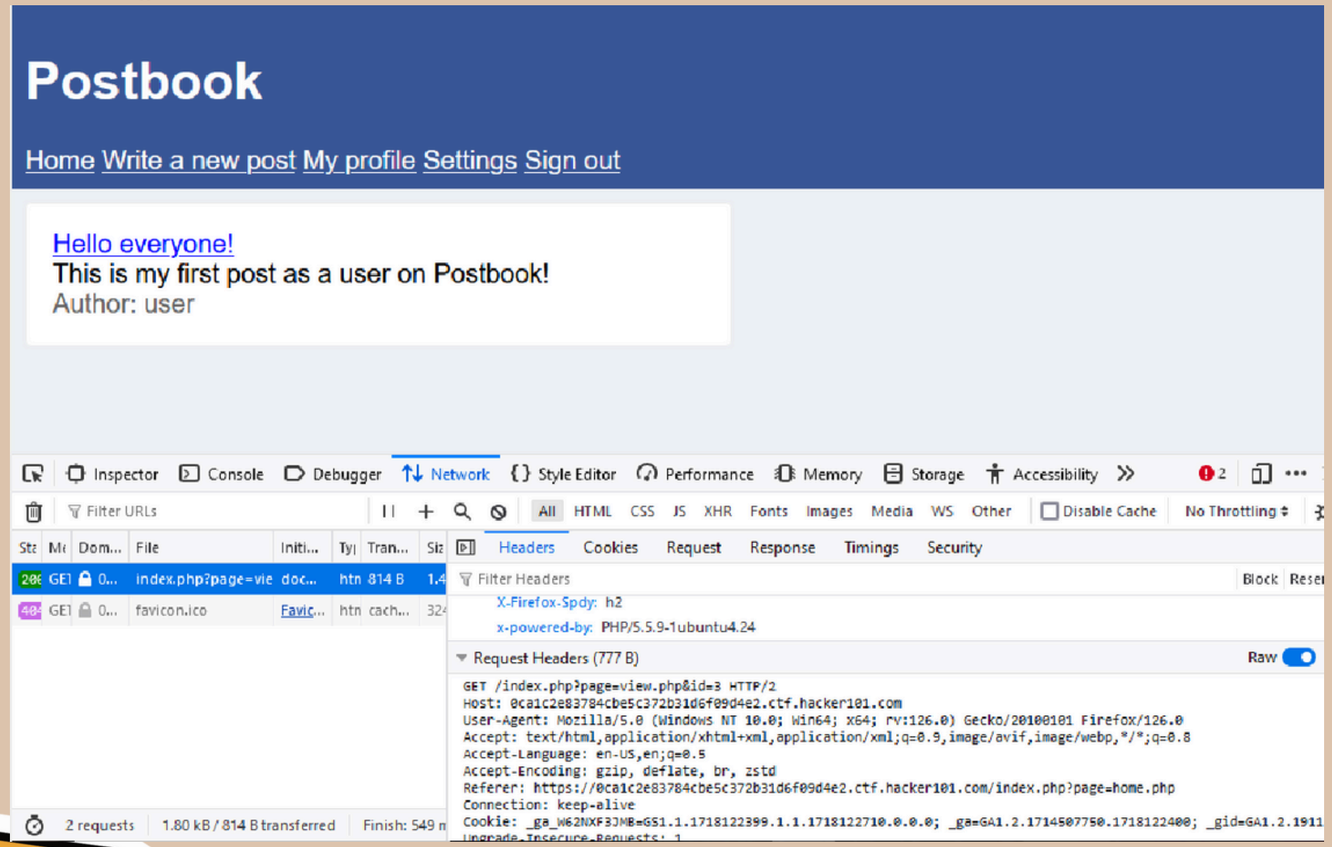



Danh sách các bài post của các người dùng

# 3. Thực nghiệm

## Kịch bản 4

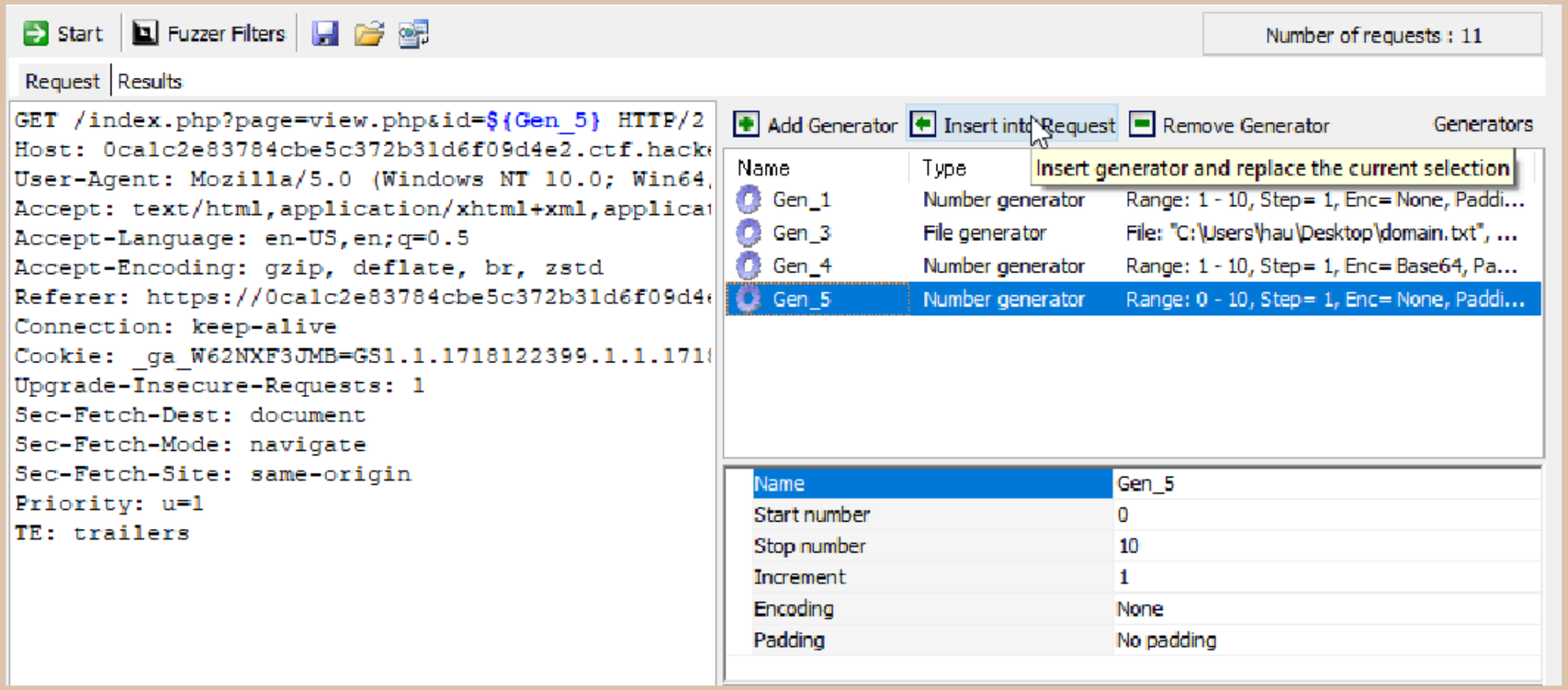
Challenge ctf.hack101.com – Postbook




Click vào một post bất kỳ để xem bài post của user đồng thời mở Dev tool đề xem request

# 3. Thực nghiệm

## Kịch bản 4




Copy nội dung request GET từ Dev Tool của Microsoft Edge vào HTTP Fuzzer của AMT và thêm tham số $Gen_5 (sau khi đã thêm Number_ Generator, click “Insert into Request”).

# 3. Thực nghiệm

**Kịch bản 4**

Click “start” để bắt đầu tấn công Brute Force bằng một loạt yêu cầu POST với id tăng dần.

Request | Results

| # | Status code | URL | Generators | Response tim |
|---|---|---|---|---|
| 0 | OK | https://003748d7418202b3ac2a965a3dc33222.ctf.hacker101.com/... | Gen_1 = 1 | 4391 |
| 1 | OK | https://003748d7418202b3ac2a965a3dc33222.ctf.hacker101.com/... | Gen_1 = 2 | 4375 |
| 2 | OK | https://003748d7418202b3ac2a965a3dc33222.ctf.hacker101.com/... | Gen_1 = 3 | 1000 |
| 3 | OK | https://003748d7418202b3ac2a965a3dc33222.ctf.hacker101.com/... | Gen_1 = 4 | 1047 |
| 4 | OK | https://003748d7418202b3ac2a965a3dc33222.ctf.hacker101.com/... | Gen_1 = 5 | 4375 |
| 5 | OK | https://003748d7418202b3ac2a965a3dc33222.ctf.hacker101.com/... | Gen_1 = 6 | 1047 |
| 6 | OK | https://003748d7418202b3ac2a965a3dc33222.ctf.hacker101.com/... | Gen_1 = 7 | 4391 |
| 7 | OK | https://003748d7418202b3ac2a965a3dc33222.ctf.hacker101.com/... | Gen_1 = 8 | 1094 |
| 8 | OK | https://003748d7418202b3ac2a965a3dc33222.ctf.hacker101.com/... | Gen_1 = 9 | 1093 |
| 9 | OK | https://003748d7418202b3ac2a965a3dc33222.ctf.hacker101.com/... | Gen_1 = 10 | 4375 |

Request | Response | View page | Raw Text

Look for: HTML

```
67        <a href="index.php?page=profile.php&id=d">My profile</a>
68        <a href="index.php?page=account.php">Settings</a>
```

# 3. Thực nghiệm

**Kịch bản 4**

Kiểm tra danh sách kết quả phát hiện phản hồi khi id=2 khác với các kết quả khác.

Request | Response | View page | Raw Text

Look for: HTML

```
71      </div>
72
73      <div id="container">
74        <div class="post">
^FLAG^cf277c0e18f38f9116e01e3782acbc464645b5ae705d396e9686410d838b2446$FLAG$</div><div class=
"post"><span style="color: red;">SECRET:</span> <a href="index.php?page=view.php&id=2">Dear
diary...</a><br />I am so glad that I am on Postbook. I can finally write down my thoughts and
no one can see them. See you tomorrow. Yours truly, admin<br /><span style="color: #666">
Author: admin</span></div>    </div>
75    </body>
76  </html>
77
```

# 3. Thực nghiệm

## Kịch bản 4

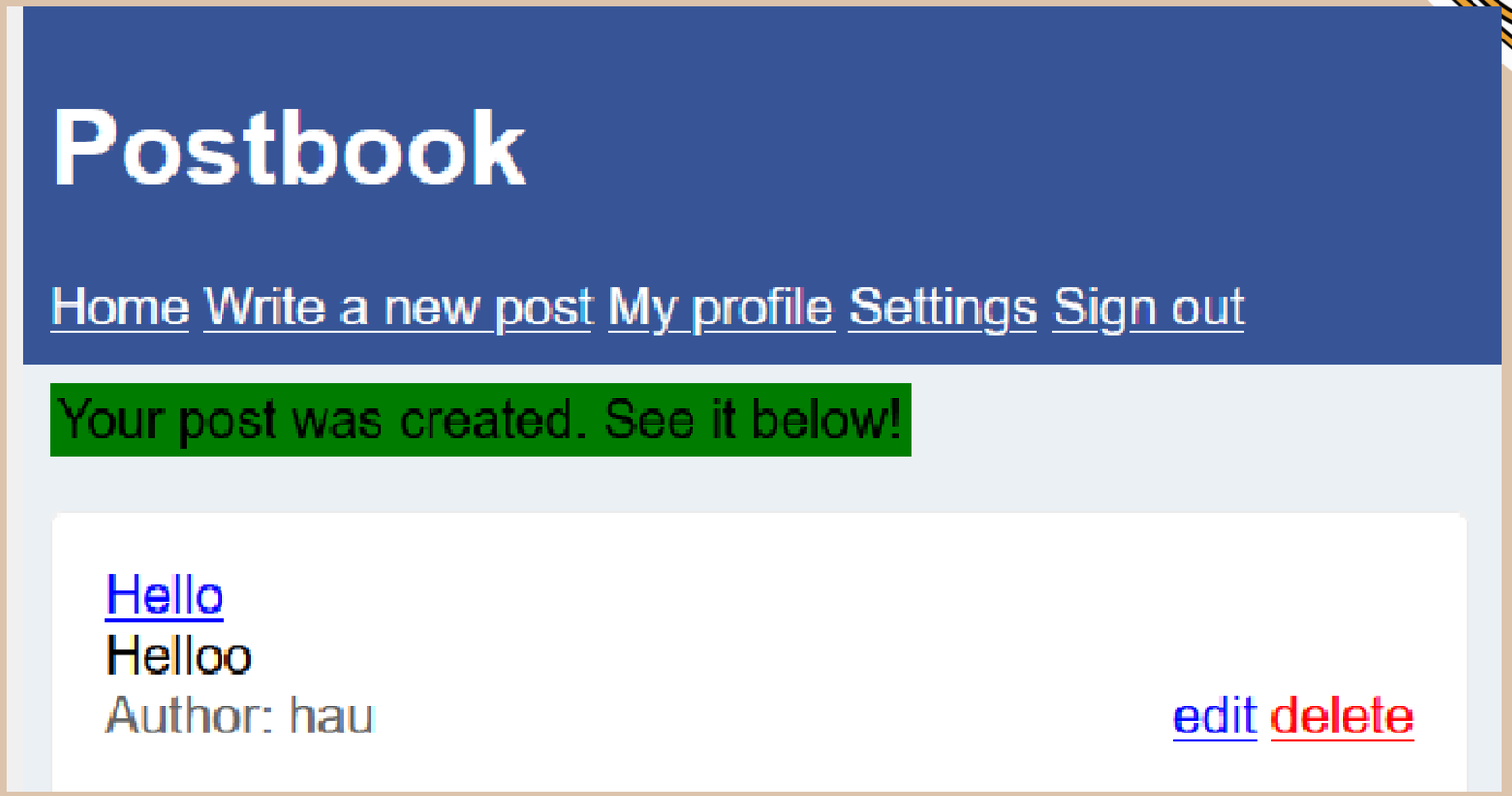
Tạo một post có nội dụng như hình bên sau đó chọn “Create post” để hiển thị bài post nay lên trang “Home”

**Tạo 1 post**

# 3. Thực nghiệm

**Kịch bản 4**

Sau khi tạo post thành công, có một trường “edit”. .Có thể trường này có lỗ hổng (có thể sửa cả bài post của user khác)

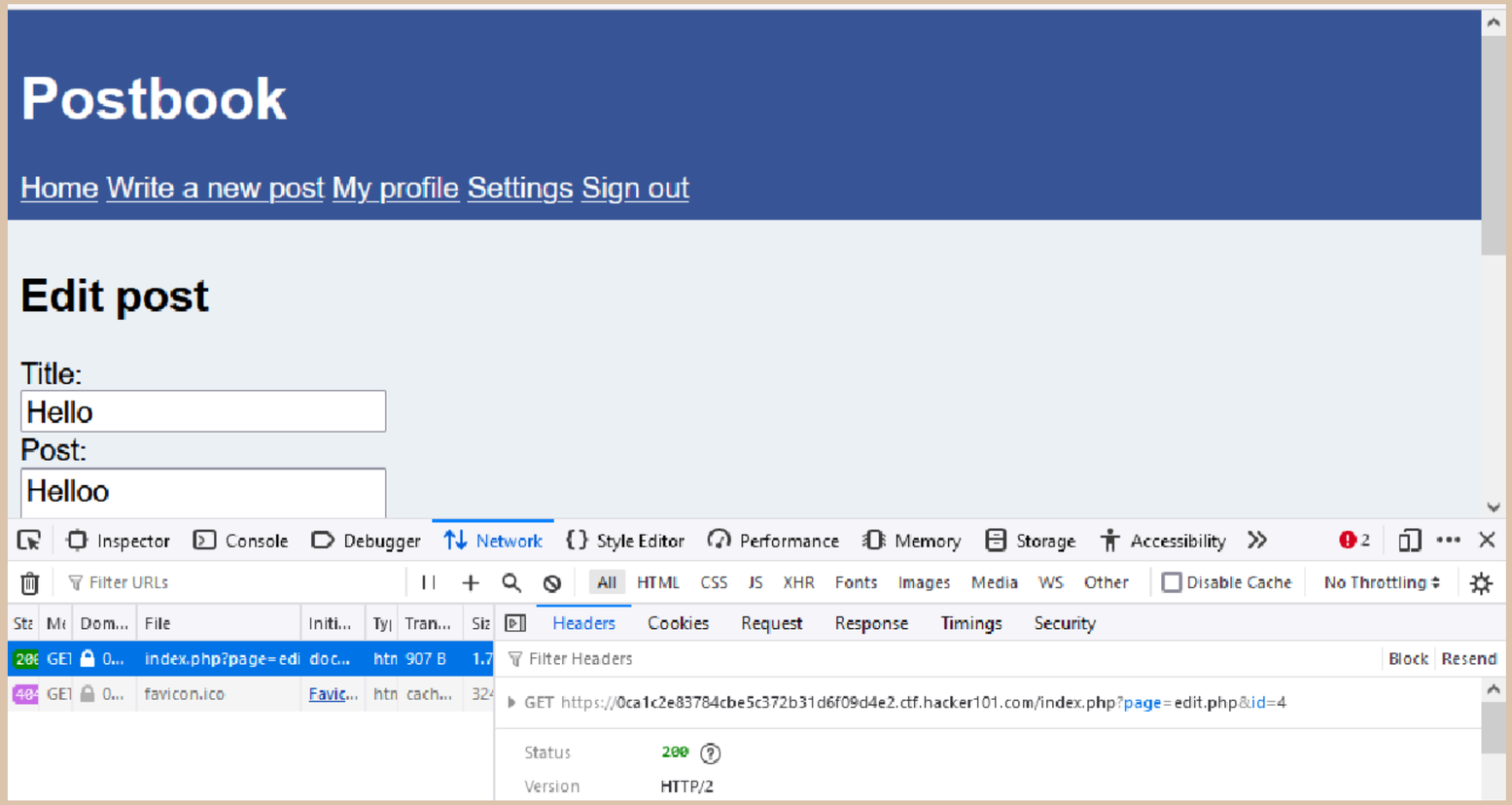



**Tạo post thành công**

# 3. Thực nghiệm

## Kịch bản 4

Thực hiện chọn "edit" đồng thời mở "Dev Tool"

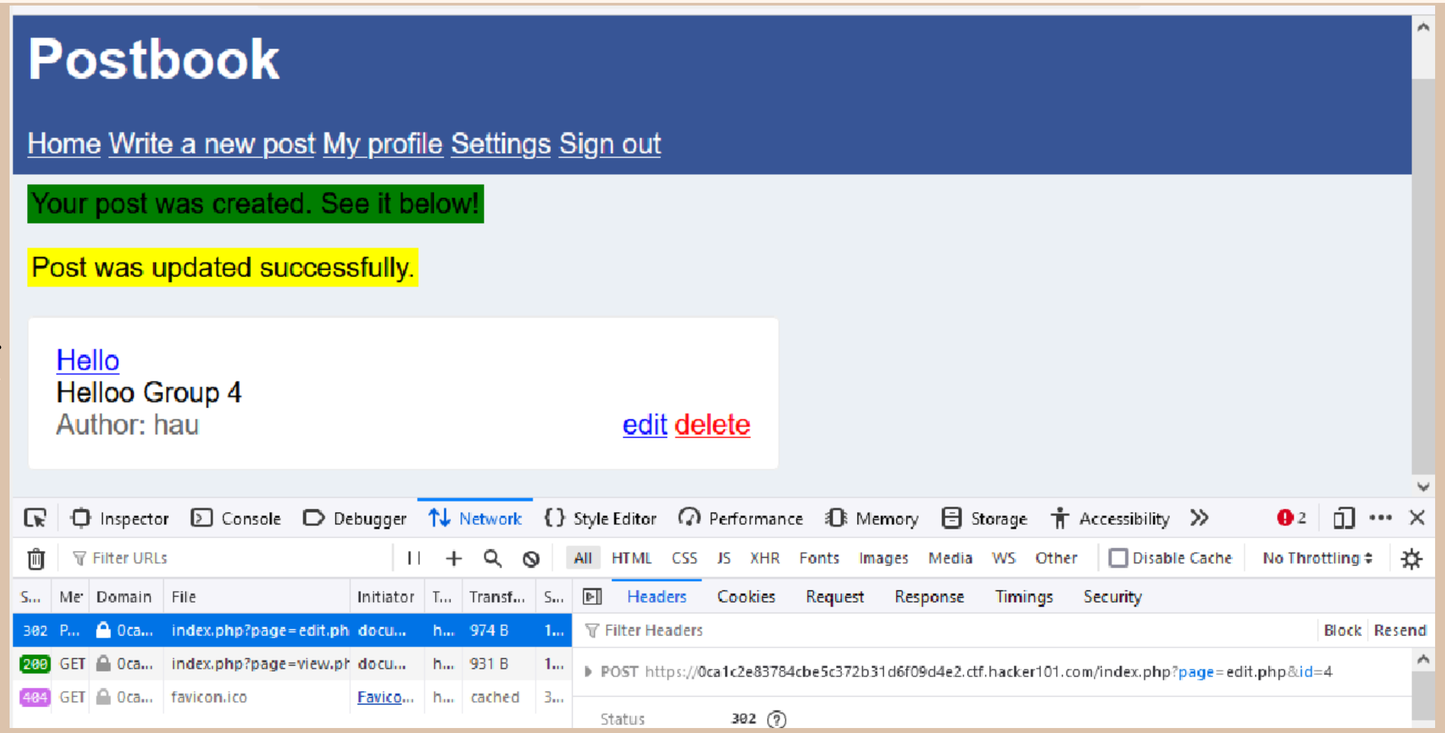



**Chọn Edit và mở Dev Tool**

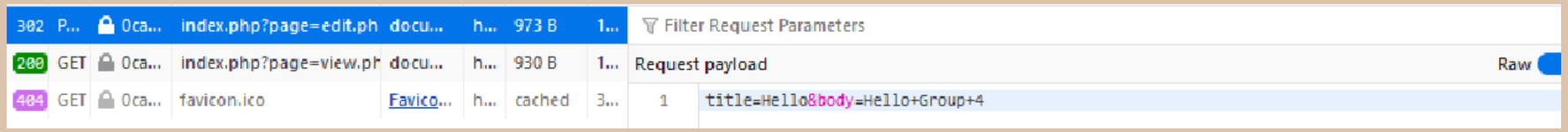
# 3. Thực nghiệm

## Kịch bản 4

Chỉnh sửa post với nội dụng bất kỳ và chọn "Save post"




Một yêu cầu POST với yêu cầu sửa đổi nội dung post

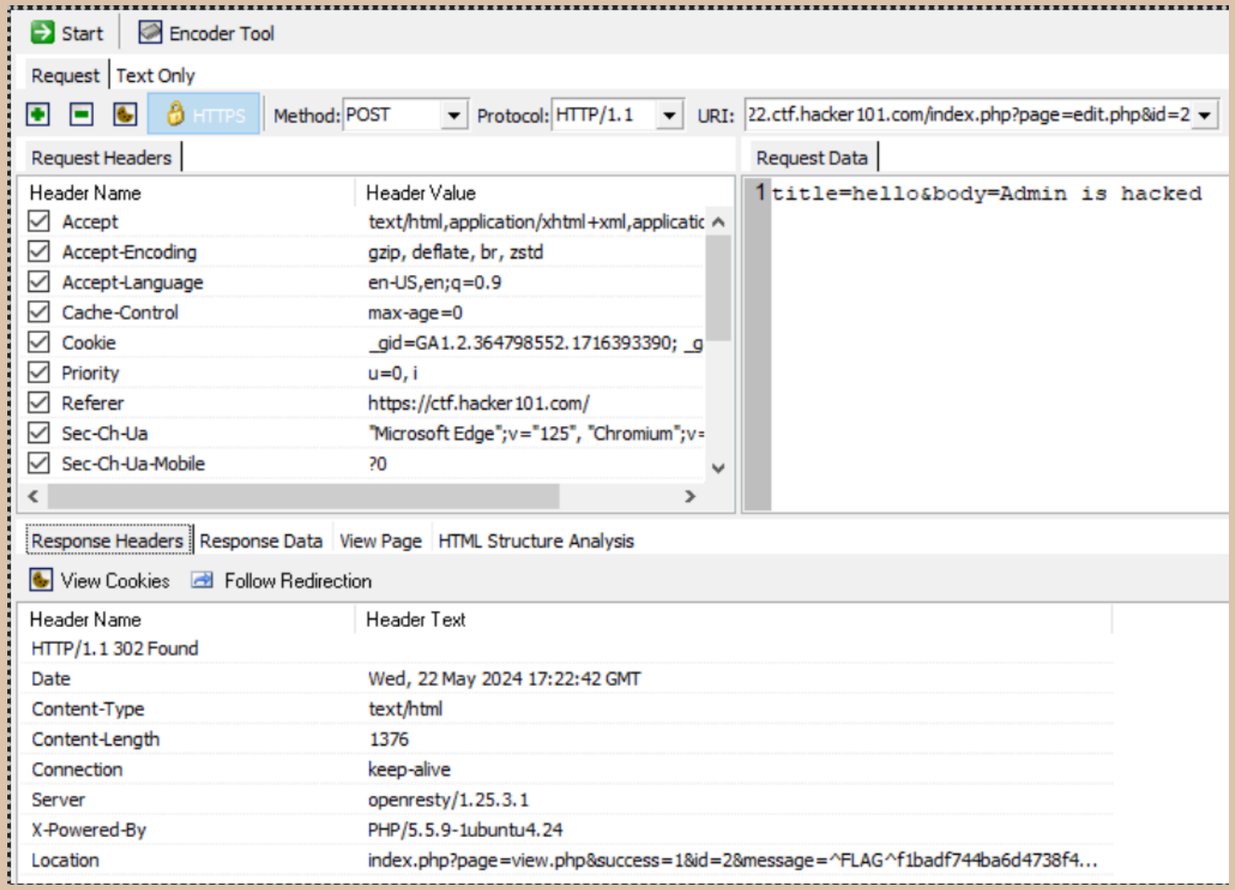



Payload của request POST

# 3. Thực nghiệm

## Kịch bản 4

Chuyển nội dung yêu cầu POST vào HTTP Editor và chỉnh sửa trường URI là id=2 và chỉnh sửa nội dung bài post của admin thành nội dung bất kỳ. Sau đó chọn “Start”



# 3. Thực nghiệm

**Kịch bản 4**

Nhận có thể chuyển hướng nên click “Follow Redirection” để chuyển hướng và nhóm nhận được phản hồi chứa flag.

sponse Headers | Response Data | View Page | HTML Structure Analysis

k for: HTML

```
    </div>

    <div id="container">
      <span style="background-color: green; padding: 3px;">Your post was created. See it below!
</span><br /><br /><span style="background-color: yellow; padding: 3px;">
^FLAG^flbadf744ba6d4738f435dca6e4153383cledbfb95f85efdc2d906feaed3d68a$FLAG$</span><br /><br />
<div class="post"><a href="index.php?page=view.php&id=2">hello</a><br />Hacked<br /><span style=
"color: #666">Author: admin</span></div>    </div>
  </body>
</html>
```

# 3. Thực nghiệm

**Kịch bản 5**

Kịch bản được thực hiện trên máy ảo Kali Linux và sử dụng Acunetix v24.1 CRACK.

Các bước thực hiện kịch bản:

- Xây dựng web PHP
- Cài đặt AcuSensor cho web
- Tiến hành quét trang web có dùng AcuSensor
- Phân tích và vá các lỗ hổng phát hiện nhờ AcuSensor.

# 3. Thực nghiệm

**Kịch bản 5**

Xây dựng web PHP

Bước 1: Tạo mới hoặc sửa đổi tệp index.html trong thư mục /var/www/html như sau

```
1 <html>
2         <head></head>
3         <body>
4                 <h1>Internal Resources</h1><br><br>
5                 <a href="catfile.php">Document Library</a>
6         </body>
7 </html>
```

# 3. Thực nghiệm

**Kịch bản 5**

Xây dựng web PHP

Bước 2: Tạo mới một tệp catfile.php trong thư mục /var/www/html với nội dụng sau:

```php
<?php
$base_folder = "/var/www/html";
$file=$_POST["filename"];
if (!empty($file)) {
  echo "<h1>This is the content of the file using the file_get_contents instruction</h1><br>";
  $filetext=file_get_contents($base_folder . "/library/" . $file);
  echo $filetext;
}
else {
  $filelist = array_diff(scandir($base_folder . "/library"), array('..','.'));
  $i=1;
  echo "<h1>List of available files</h1><br><br>";
  echo "<table border=\"1\" style=\"padding:5px\">";
  foreach ($filelist as $value) {
    if ($i==1) echo "<tr>";
    echo "<td style=\"padding:5px\">".$value."</td>";
    if ($i==5) {
      echo "</tr>";
      $i=0;
    }
  $i++;
  }
echo "</table><br><br>";
echo "<form action=\"catfile.php\" method=\"post\">";
echo "Enter desired filename: <input type=\"text\" name=\"filename\"><br>";
echo "<input type=\"submit\">";
echo "</form>";
}
?>
```

# 3. Thực nghiệm

**Kịch bản 5**

Xây dựng web PHP

Bước 3: Trang web đang tạo có thể hiển thị nội dung các tệp dựa vào filename nhận từ phía khách. Tạo các tệp txt mẫu để có đối tượng hiển thị và chạy dịch vụ apache2. Lần lượt chạy các câu lệnh sau:

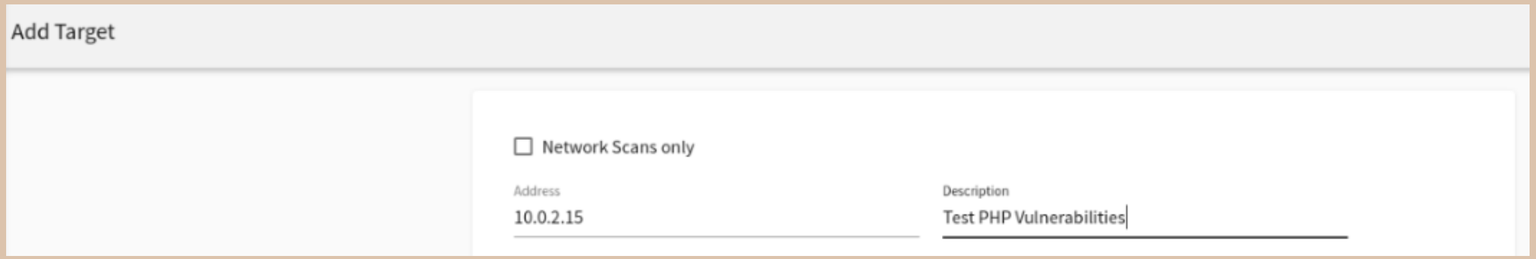
- echo $'example contents\n' > /var/www/html/library/Chapter_01.txt
- tee </var/www/html/library/Chapter_01.txt /var/www/html/library/Chapter_{02..40}.txt >/dev/null
- sudo systemctl start apache2

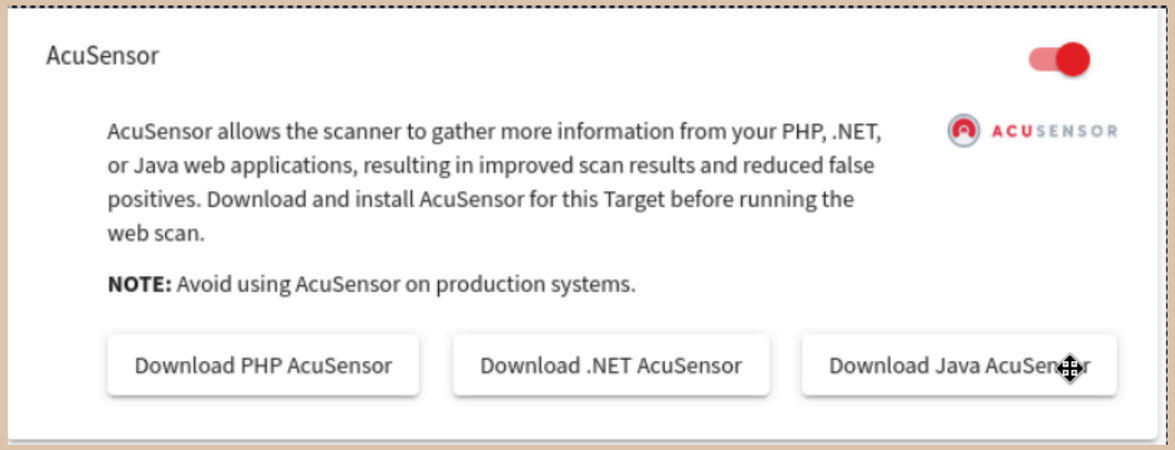
# 3. Thực nghiệm

**Kịch bản 5**

Cài đặt AcuSensor cho trang web:

Bước 1: Tạo Mục tiêu trên Acunetix và trong phần cấu hình Mục tiêu bật AcuSensor.



# 3. Thực nghiệm

**Kịch bản 5**

Cài đặt AcuSensor cho trang web:

Bước 2: Sau khi bật AcuSensor, click “Download PHP AcuSensor”. Tệp “acu_phpaspect.php” sẽ được tải về. Các trường thông tin khác trong phần cấu hình Mục tiêu giữ nguyên, click “Save”.

Bước 3: Tạo thư mục “acusensor” trong thư mục “/var/www/html/” và chuyển tệp “acu_phpaspect.php” vào.

# 3. Thực nghiệm

**Kịch bản 5**

Cài đặt AcuSensor cho trang web:

Bước 4: Chỉnh sửa tệp php.ini (thường có đường dẫn là /etc/php/8.2/apache2/php.ini) bằng lệnh sudo nano /etc/php/8.2/apache2/php.ini . Tìm dòng có chuỗi "auto_prepend_file" và sửa thành "auto_prepend_file = /acusensor/acu_phpaspect.php". Dòng mới này sẽ chạy một tệp PHP trước khi các tệp PHP khác được chạy.

Bước 5: Khởi động lại dịch vụ Apache2: sudo systemctl restart apache2

# 3. Thực nghiệm

**Kịch bản 5**

Tiến hành quét trang web có dùng AcuSensor: Phát hiện lỗ hổng Directory traversal ở file catfile.php dòng 6.

**Attack Details**

URL encoded POST input **filename** was set to **1ACUSTARTFILE/../../xxx\..\..\ACUENDFILE**

Source file: **/var/www/html/catfile.php** line: **6**

Additional details:

```
File: /var/www/html/library/1ACUSTARTFILE/../../xxx\..\..\ACUENDFILE
"file_get_contents" was called.
```

# 3. Thực nghiệm

**Kịch bản 5**

Sửa lỗ hổng bằng cách thực hiện lọc đầu vào có chứa kí tự đặc biệt mà trong ngữ cảnh hiện tại chính là "../".

```php
<?php
$base_folder = "/var/www/html";
$file=$_POST["filename"];
$file_validated = str_replace ("../","",$file);
if (!strcmp($file,$file_validated) == 0){
        header("Location: /");
        exit;
}
if (!empty($file)) {
  echo "<h1>This is the content of the file using the file_get_contents instruction</h1><br>";
  $filetext=file_get_contents($base_folder . "/library/" . $file);
  echo $filetext;
```

# 3. Thực nghiệm

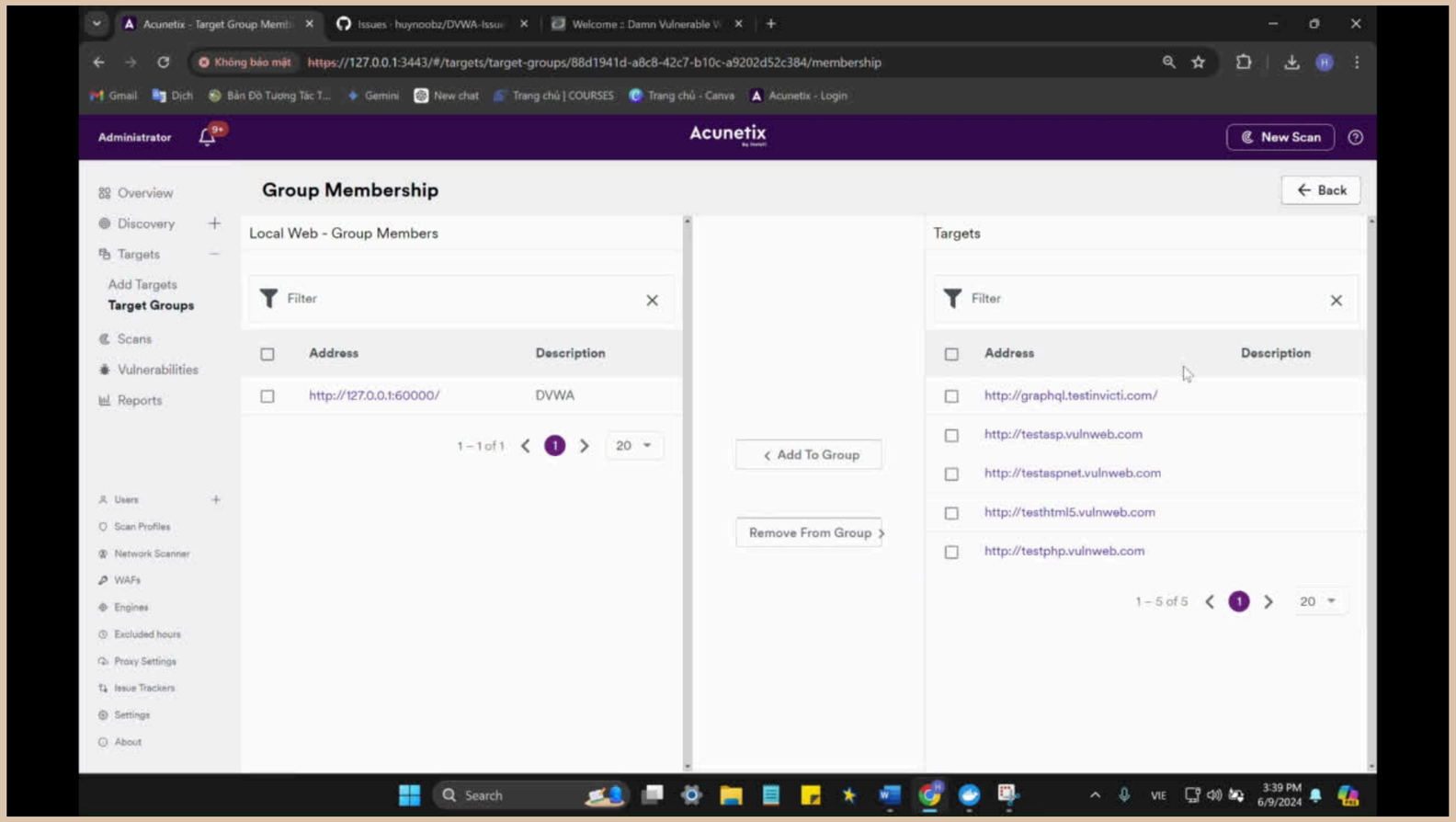
## Kịch bản 5

Sau khi vá lỗ hổng và thực hiện quét lại trang web, có thể thấy không còn phát hiện các lỗ hổng bảo mật trên nữa.

| | | | | |
|---|---|---|---|---|
| http://10.0.2.15 | High Risk Vulnerabilities | Last run on Jun 5, 2024, 6:04:57 AM | 0 0 0 0 | Completed |
| http://10.0.2.15 | High Risk Vulnerabilities | Last run on Jun 5, 2024, 4:36:48 AM | 2 1 0 0 | Completed |

# 3. Thực nghiệm

## Kịch bản 6



# NỘI DUNG BÁO CÁO

1. Giới thiệu đề tài
2. Tổng quan về Acunetix
3. Thực nghiệm
4. Kết luận và đánh giá

# 4. Kết luận và đánh giá

**Kết luận**

Acunetix là một công cụ mạnh mẽ và toàn diện trong lĩnh vực kiểm thử bảo mật ứng dụng web, giúp phát hiện và xử lý các lỗ hổng bảo mật một cách hiệu quả.

Acunetix còn hỗ trợ phân tích thủ công bằng cách cung cấp các bằng chứng khai thác và nhà phát triển cũng cung thêm bộ công cụ thủ công – AMT.

**=>** Acunetix là một giải pháp đáng tin cậy và hiệu quả, đóng góp quan trọng vào việc bảo vệ ứng dụng web khỏi các mối đe dọa ngày càng tinh vi trên không gian mạng.

# 4. Kết luận và đánh giá

**Đánh giá**

Điểm mạnh:

- Khả năng quét toàn diện nhiều loại lỗ hổng cũng như nhiều loại ngôn ngữ lập trình.
- Tính năng tự động hóa cao.
- Cung cấp các báo cáo chi tiết và dễ hiểu.
- Giao diện thân thiện và dễ sử dụng.
- Hỗ trợ phân tích thủ công

# 4. Kết luận và đánh giá

**Đánh giá**

Điểm yếu:

- Chi phí cao giao động từ 70 - 170 triệu đồng tuỳ phiên bản.

# THANK YOU SO MUCH!