

Report on Remote code execution via web shell upload

**Presented By
Tanuj Sood**

Table of Contents:-

Sr. No.	Contents	Page No.
1	Introduction	3
2	Objectives	3
3	Requirements	3
4	Scope	4
5	Tools Used	4
6	Methodology Used	4-12
7	Flow Diagram	13
8	Recommendation	14
9	Conclusion	15
10	References	15

1-Introduction:

This scenario involves a website with a vulnerability in its image upload feature. This vulnerability allows users to upload files without proper validation, meaning they can upload files that are not images, such as PHP scripts disguised as images.

The objective is to exploit this vulnerability by uploading a PHP web shell. A web shell is a script that can be uploaded to a web server to gain remote access and control. Once uploaded, the attacker can execute commands on the server remotely.

In this scenario, the goal is to upload a PHP web shell and then use it to access and retrieve the contents of a specific file located at `"/home/carlos/secret"`. This file contains sensitive information that needs to be retrieved.

2-Objectives:

The primary objective is to exploit a vulnerability within a web application's image upload function. This vulnerability enables users to upload files without adequate validation, potentially allowing for the uploading of malicious files, such as PHP scripts disguised as images.

The aim is to take advantage of this vulnerability by uploading a PHP web shell. This web shell serves as a means to gain remote access and control over the server. Once the web shell is successfully uploaded, the next step is to utilize it to access and retrieve the contents of a specific file located at `"/home/carlos/secret"`. The ultimate goal is to retrieve the sensitive information contained within this file.

3- Requirements:

1. VirtualBox
2. Kali Linux virtual machine
3. Foxy proxy
4. Burp suite
5. Knowledge about cyber security fundamentals
6. Knowledge about web security.
7. Know about networking Fundamentals
8. Basic Knowledge of PHP

4- Scope:

The scope entails exploiting a vulnerability in a web application's image upload, lacking proper validation. Users can upload files, including malicious PHP scripts. The aim is to upload a PHP web shell to gain remote server access. Post-exploitation, the objective is to retrieve sensitive information from a specified file. This encompasses identifying, exploiting, and post-exploitation activities. The vulnerability enables unauthorized access and control over the server. The web shell facilitates remote execution of commands. Attacker gains access to the server environment upon successful upload. Accessing "/home/carlos/secret" file marks completion of the task. The focus is on the impact of the vulnerability and the severity of exploitation. The scenario underscores the importance of robust security measures. It highlights risks associated with inadequate validation in web applications. The objective is to demonstrate the potential consequences of such vulnerabilities. The scope emphasizes the need for thorough security assessments and measures in web development

5-Tools Used:

1. Virtual Box
2. Kali Linux Virtual machine
3. Burp suite
4. Foxy proxy

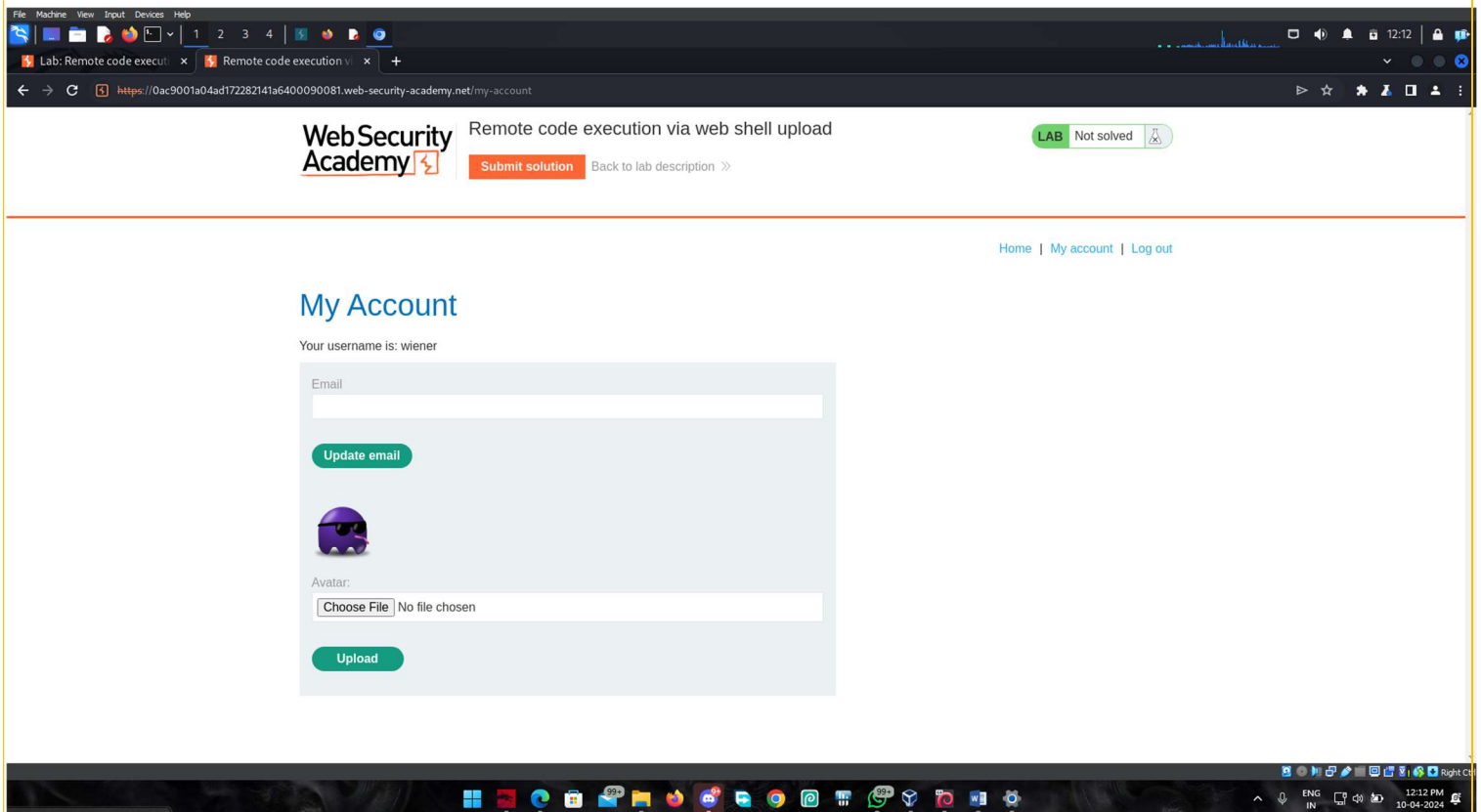
6- Methodology Used:

- i. **Initial Assessment:** Analyze the web application to identify potential vulnerabilities, focusing on the image upload feature.
- ii. **Vulnerability Identification:** Determine if the application lacks proper validation, allowing for the upload of non-image files.
- iii. **Exploitation:** Develop or obtain a PHP web shell and disguise it as an image file. Upload the web shell using the vulnerable upload function.
- iv. **Remote Access:** Once the web shell is uploaded successfully, use it to establish remote access to the server.
- v. **Command Execution:** Utilize the web shell's capabilities to execute commands on the server remotely.
- vi. **File Retrieval:** Access the "/home/carlos/secret" file using the web shell's functionality to retrieve its contents.

- vii. **Data Exfiltration:** Extract the sensitive information from the retrieved file.
- viii. **Submission:** Submit the extracted secret using the provided interface to complete the task.
- ix. **Cleanup and Mitigation:** Remove any traces of the web shell and address the vulnerability to prevent further exploitation.
- x. **Documentation:** Document the steps taken, findings, and remediation measures for future reference and improvement of security practices

8th April 2024

Step 1: log in to your account and notice the option for uploading an avatar image. Upload an arbitrary image, then return to your account page. Notice that a preview of your avatar is now displayed on the page.



8th April 2024

Step 2: In Burp, go to **Proxy > HTTP history**. Click the filter bar to open the **HTTP history filter window**. Under **Filter by MIME type**, enable the **Images** checkbox, then apply your changes.

The screenshot shows the Burp Suite Community Edition v2024.1.15 interface. The 'HTTP history' tab is active, displaying a list of intercepted HTTP requests. A 'Configure filter' dialog box is open, showing the 'Settings mode' and 'Bambda mode' tabs. The 'Filter by MIME type' section is expanded, and the 'Images' checkbox is checked. The 'Filter by request type' section is also expanded, showing 'Show only in-scope items' and 'Show only parameterized requests' options. The 'Filter by search term' section is expanded, showing 'Regex', 'Case sensitive', and 'Negative search' options. The 'Filter by file extension' section is expanded, showing 'Show only: asp,aspx,jsp,php' and 'Hide: js,gif,jpg,png,css' options. The 'Filter by status code' section is expanded, showing '2xx [success]', '3xx [redirection]', '4xx [request error]', and '5xx [server error]' options. The 'Filter by annotation' section is expanded, showing 'Show only items with notes' and 'Show only highlighted items' options. The 'Filter by listener' section is expanded, showing 'Port' options. The 'Filter by request type' section is expanded, showing 'Show only in-scope items' and 'Show only parameterized requests' options. The 'Filter by search term' section is expanded, showing 'Regex', 'Case sensitive', and 'Negative search' options. The 'Filter by file extension' section is expanded, showing 'Show only: asp,aspx,jsp,php' and 'Hide: js,gif,jpg,png,css' options. The 'Filter by status code' section is expanded, showing '2xx [success]', '3xx [redirection]', '4xx [request error]', and '5xx [server error]' options. The 'Filter by annotation' section is expanded, showing 'Show only items with notes' and 'Show only highlighted items' options. The 'Filter by listener' section is expanded, showing 'Port' options. The 'Filter by request type' section is expanded, showing 'Show only in-scope items' and 'Show only parameterized requests' options. The 'Filter by search term' section is expanded, showing 'Regex', 'Case sensitive', and 'Negative search' options. The 'Filter by file extension' section is expanded, showing 'Show only: asp,aspx,jsp,php' and 'Hide: js,gif,jpg,png,css' options. The 'Filter by status code' section is expanded, showing '2xx [success]', '3xx [redirection]', '4xx [request error]', and '5xx [server error]' options. The 'Filter by annotation' section is expanded, showing 'Show only items with notes' and 'Show only highlighted items' options. The 'Filter by listener' section is expanded, showing 'Port' options.

#	Host	Method	URL	Params	Status code	Length	MIME type	Extension	Title	Notes	TLS	IP	Cookies	Time	Listener port
1	https://chrome.google.com	POST	/v2/items/_storeMetadataBatchGet		403	1263	HTML		Sorry...		✓	142.250.77.234		11:58:22 10 Apr 20...	8080
2	https://www.google.com	GET	/search?q=port&oeq=port&gs_l=crpEgZjAHUWUy...		200	514390	HTML		port - Google Search		✓	216.58.196.100	1P_JAR=2024-04-10-06...	12:03:13 10 Apr 2...	8080
3	https://portswigger.net	GET	/		200	48645	HTML		Web Application Security, Testin...		✓	34.240.117.4	SessionId=CTDJBPIYIFOF...	12:03:14 10 Apr 2...	8080
7	https://portswigger.net	GET	/content/images/svg/icon/enterprise.svg		200	2094	XML	svg			✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:15 10 Apr 2...	8080
8	https://portswigger.net	GET	/content/images/svg/icon/professional.svg		200	1938	XML	svg			✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:15 10 Apr 2...	8080
9	https://portswigger.net	GET	/content/images/svg/icon/community.svg		200	2094	XML	svg			✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
10	https://portswigger.net	GET	/mega-nav/images/dastandly.svg		200	1914	XML	svg			✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
11	https://portswigger.net	GET	/mega-nav/images/burp-suite-scanner.jpg		200	16903	JPEG	jpg			✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
12	https://portswigger.net	GET	/mega-nav/images/latest-burp-suite-software-d...		200	8380	JPEG	jpg			✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
13	https://portswigger.net	GET	/images/portswigger-homepage-hero.jpg		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
14	https://portswigger.net	GET	/bundles/public/statics/js/comp2/ANAMM...		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
15	https://portswigger.net	GET	/images/company-logos/amazon.svg		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
16	https://portswigger.net	GET	/images/logo/g2-spring-leader-23.png		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
17	https://portswigger.net	GET	/images/logo/customer-choice.jpg		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
18	https://portswigger.net	GET	/images/company-logos/nasa.svg		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
19	https://portswigger.net	GET	/images/company-logos/barclays.svg		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
20	https://portswigger.net	GET	/images/company-logos/fedex.svg		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
21	https://portswigger.net	GET	/images/company-logos/axa.svg		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
22	https://portswigger.net	GET	/images/burp-suite.jpg		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
23	https://portswigger.net	GET	/images/burp-suite-small.svg		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
24	https://portswigger.net	GET	/images/web-security-academy.jpg		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
25	https://portswigger.net	GET	/images/portswigger-research.jpg		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
26	https://portswigger.net	GET	/images/research-small.svg		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
27	https://portswigger.net	GET	/images/academy-small.svg		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
28	https://portswigger.net	GET	/images/validate-your-certification.svg		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
29	https://portswigger.net	GET	/images/certification-resources-black.jpg		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
30	https://portswigger.net	GET	/content/images/logo/portswigger-white...		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
31	https://portswigger.net	GET	/images/community-stok.jpg		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
32	https://portswigger.net	GET	/content/images/logo/portswigger-resear...		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
33	https://portswigger.net	GET	/images/community-aleksandr.jpg		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
34	https://portswigger.net	GET	/images/community-katie.jpg		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
35	https://portswigger.net	GET	/images/daf-s.jpg		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
36	https://portswigger.net	GET	/images/community-wel.jpg		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
37	https://portswigger.net	GET	/images/banner-globe.jpg		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
38	https://portswigger.net	GET	/images/portswigger-blog.jpg		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
39	https://portswigger.net	GET	/images/full-width-staff-banner.jpg		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
40	https://portswigger.net	GET	/images/top-10-techniques-2023-banner-small...		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
41	https://portswigger.net	GET	/images/enterprise-vs-professional.jpg		200	1174					✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
42	https://portswigger.net	GET	/287552c2-4917-42e0-8862-ba994a2a73d7.js		200	23644	script	js			✓	20.79.102.66		12:03:16 10 Apr 2...	8080
43	https://ps.containers.piwik.pro	GET	/images/certification-product.png		200	23941	PNG	png			✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
44	https://portswigger.net	GET	/images/getting-started-with-the-web-security...		200	18294	JPEG	jpg			✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
45	https://portswigger.net	GET	/images/burp-suite-roadmap-update-july-2023...		200	64195	png	png			✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
46	https://portswigger.net	GET	/images/enterprise-edition-video-tutorials.jpg		200	16436	JPEG	jpg			✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
47	https://portswigger.net	GET	/content/images/logo/portswigger-logo.svg		200	4757	XML	svg			✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
48	https://portswigger.net	GET	/content/images/icon/icon-dot-scan.svg		200	9553	XML	svg			✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080
49	https://portswigger.net	GET	/content/images/icon/icon-dot-scan.svg		200	9553	XML	svg			✓	34.240.117.4	AWVSALBAPP-0=remov...	12:03:16 10 Apr 2...	8080

8th April 2024

Step 3: In the proxy history, notice that your image was fetched using a GET request to /files/avatars/<YOUR-IMAGE>. Send this request to Burp Repeater.

The screenshot displays the Burp Suite interface with the HTTP history tab active. The list of requests is filtered to show only those with a status code of 200. The selected request is #294, which is a GET request to /files/avatars/PurpleThing.jpg. The response is visible in the bottom pane, showing a 200 status code and the image content.

#	Host	Method	URL	Params	Status code	Length	MIME type	Extension	Title	Notes	TLS	IP	Cookies	Time	Listener port
236	https://googleads.g.doubleclick.net	GET	/pagead/ids/rd=1		200	836	JSON				✓	142.250.206.162		12:03:41 10 Apr 2...	8080
237	https://www.youtube.com	GET	/generate_2047-CO63Q		204	182					✓	142.250.182.174		12:03:41 10 Apr 2...	8080
240	https://portswigger.net	GET	/academy/labs/launch/535975a13e89837c0707c...		302	1479					✓	34.240.117.4		12:03:42 10 Apr 2...	8080
242	https://0ac9001a04ad72282141a6400...	GET	/		200	8641	HTML		Remote code execution via web ...		✓	79.125.84.16	AWSALBAPP-Grp_remo... session=0q5QCofmt...	12:03:56 10 Apr 2...	8080
244	https://0ac9001a04ad72282141a6400...	GET	/resources/labheader/js/labHeader.js		200	1673	script	js			✓	79.125.84.16		12:03:57 10 Apr 2...	8080
246	https://0ac9001a04ad72282141a6400...	GET	/resources/labheader/js/submitSolution.js		200	1333	script	js			✓	79.125.84.16		12:03:57 10 Apr 2...	8080
247	https://0ac9001a04ad72282141a6400...	GET	/resources/images/blog.svg		200	7499	image	svg			✓	79.125.84.16		12:03:57 10 Apr 2...	8080
248	https://0ac9001a04ad72282141a6400...	GET	/image/blog/posts/43.jpg		200	261948	JPEG	jpg			✓	79.125.84.16		12:03:57 10 Apr 2...	8080
249	https://0ac9001a04ad72282141a6400...	GET	/academyLabHeader		101	147					✓	79.125.84.16		12:03:57 10 Apr 2...	8080
250	https://0ac9001a04ad72282141a6400...	GET	/image/blog/posts/63.jpg		200	162457	JPEG	jpg			✓	79.125.84.16		12:03:57 10 Apr 2...	8080
251	https://0ac9001a04ad72282141a6400...	GET	/image/blog/posts/12.jpg		200	28387	JPEG	jpg			✓	79.125.84.16		12:03:57 10 Apr 2...	8080
252	https://0ac9001a04ad72282141a6400...	GET	/image/blog/posts/8.jpg		200	32889	JPEG	jpg			✓	79.125.84.16		12:03:58 10 Apr 2...	8080
253	https://0ac9001a04ad72282141a6400...	GET	/image/blog/posts/27.jpg		200	32603	JPEG	jpg			✓	79.125.84.16		12:03:58 10 Apr 2...	8080
254	https://0ac9001a04ad72282141a6400...	GET	/image/blog/posts/53.jpg		200	203033	JPEG	jpg			✓	79.125.84.16		12:03:58 10 Apr 2...	8080
255	https://0ac9001a04ad72282141a6400...	GET	/image/blog/posts/22.jpg		200	47389	JPEG	jpg			✓	79.125.84.16		12:03:58 10 Apr 2...	8080
256	https://0ac9001a04ad72282141a6400...	GET	/image/blog/posts/13.jpg		200	41997	JPEG	jpg			✓	79.125.84.16		12:03:58 10 Apr 2...	8080
257	https://0ac9001a04ad72282141a6400...	GET	/image/blog/posts/29.jpg		200	113819	JPEG	jpg			✓	79.125.84.16		12:03:58 10 Apr 2...	8080
258	https://0ac9001a04ad72282141a6400...	GET	/image/blog/posts/66.jpg		200	268200	JPEG	jpg			✓	79.125.84.16		12:03:58 10 Apr 2...	8080
259	https://0ac9001a04ad72282141a6400...	GET	/resources/labheader/images/logoAcademy.svg		200	8852	image	svg			✓	79.125.84.16		12:03:58 10 Apr 2...	8080
260	https://0ac9001a04ad72282141a6400...	GET	/resources/labheader/images/ps-lab-notsolved.svg		200	942	image	svg			✓	79.125.84.16		12:03:58 10 Apr 2...	8080
262	https://0ac9001a04ad72282141a6400...	GET	/favicon.ico		200	15540	image	ico			✓	79.125.84.16		12:03:58 10 Apr 2...	8080
267	https://googleads.g.doubleclick.net	GET	/pagead/ids/rd=1		302	745	HTML				✓	142.250.206.162		12:05:41 10 Apr 2...	8080
268	https://googleads.g.doubleclick.net	GET	/pagead/ids/rd=1		200	836	JSON				✓	142.250.206.162		12:05:41 10 Apr 2...	8080
269	https://0ac9001a04ad72282141a6400...	GET	/my-account		302	86					✓	79.125.84.16		12:05:55 10 Apr 2...	8080
270	https://0ac9001a04ad72282141a6400...	GET	/login		200	3530	HTML		Remote code execution via web ...		✓	79.125.84.16		12:06:02 10 Apr 2...	8080
273	https://0ac9001a04ad72282141a6400...	GET	/resources/labheader/js/labHeader.js		200	1673	script	js			✓	79.125.84.16		12:06:13 10 Apr 2...	8080
274	https://0ac9001a04ad72282141a6400...	GET	/resources/labheader/js/submitSolution.js		200	1333	script	js			✓	79.125.84.16		12:06:13 10 Apr 2...	8080
275	https://0ac9001a04ad72282141a6400...	GET	/resources/labheader/images/ps-lab-notsolved.svg		200	942	image	svg			✓	79.125.84.16		12:06:13 10 Apr 2...	8080
276	https://0ac9001a04ad72282141a6400...	GET	/resources/labheader/images/logoAcademy.svg		200	8852	image	svg			✓	79.125.84.16		12:06:13 10 Apr 2...	8080
277	https://0ac9001a04ad72282141a6400...	GET	/academyLabHeader		400	130	text				✓	79.125.84.16		12:06:13 10 Apr 2...	8080
279	https://0ac9001a04ad72282141a6400...	GET	/my-account/subscribe		200	4322	HTML		Remote code execution via web ...		✓	34.246.129.62		12:10:31 10 Apr 2...	8080
282	https://0ac9001a04ad72282141a6400...	GET	/resources/labheader/js/labHeader.js		200	1673	script	js			✓	34.246.129.62		12:11:31 10 Apr 2...	8080
283	https://0ac9001a04ad72282141a6400...	GET	/resources/labheader/js/submitSolution.js		200	1333	script	js			✓	34.246.129.62		12:11:31 10 Apr 2...	8080
284	https://0ac9001a04ad72282141a6400...	GET	/resources/images/avatars/default.svg		200	10015	image	svg			✓	34.246.129.62		12:11:31 10 Apr 2...	8080
285	https://0ac9001a04ad72282141a6400...	GET	/resources/labheader/images/ps-lab-notsolved.svg		200	942	image	svg			✓	34.246.129.62		12:11:31 10 Apr 2...	8080
286	https://0ac9001a04ad72282141a6400...	GET	/resources/labheader/images/logoAcademy.svg		200	8852	image	svg			✓	34.246.129.62		12:11:31 10 Apr 2...	8080
287	https://0ac9001a04ad72282141a6400...	GET	/academyLabHeader		400	130	text				✓	34.246.129.62		12:11:31 10 Apr 2...	8080
290	https://0ac9001a04ad72282141a6400...	GET	/favicon.ico		200	15540	image	ico			✓	34.246.129.62		12:12:25 10 Apr 2...	8080
291	https://0ac9001a04ad72282141a6400...	GET	/my-account		200	4318	HTML		Remote code execution via web ...		✓	34.246.129.62		12:12:29 10 Apr 2...	8080
293	https://0ac9001a04ad72282141a6400...	GET	/resources/labheader/js/labHeader.js		200	1673	script	js			✓	34.246.129.62		12:12:30 10 Apr 2...	8080
294	https://0ac9001a04ad72282141a6400...	GET	/files/avatars/PurpleThing.jpg		200	165064	image	jpg			✓	34.246.129.62		12:12:30 10 Apr 2...	8080
296	https://0ac9001a04ad72282141a6400...	GET	/resources/labheader/js/submitSolution.js		200	1333	script	js			✓	34.246.129.62		12:12:30 10 Apr 2...	8080
297	https://0ac9001a04ad72282141a6400...	GET	/resources/labheader/images/logoAcademy.svg		200	8852	image	svg			✓	34.246.129.62		12:12:30 10 Apr 2...	8080
298	https://0ac9001a04ad72282141a6400...	GET	/resources/labheader/images/ps-lab-notsolved.svg		200	942	image	svg			✓	34.246.129.62		12:12:30 10 Apr 2...	8080
299	https://0ac9001a04ad72282141a6400...	GET	/academyLabHeader		400	130	text				✓	34.246.129.62		12:12:30 10 Apr 2...	8080
300	https://portswigger.net	GET	/academy/labs/launch/535975a13e89837c0707c...		302	1479					✓	34.240.117.4	AWSALBAPP-Grp_remo...	12:12:58 10 Apr 2...	8080

Request

Event log (2) All issues

Response

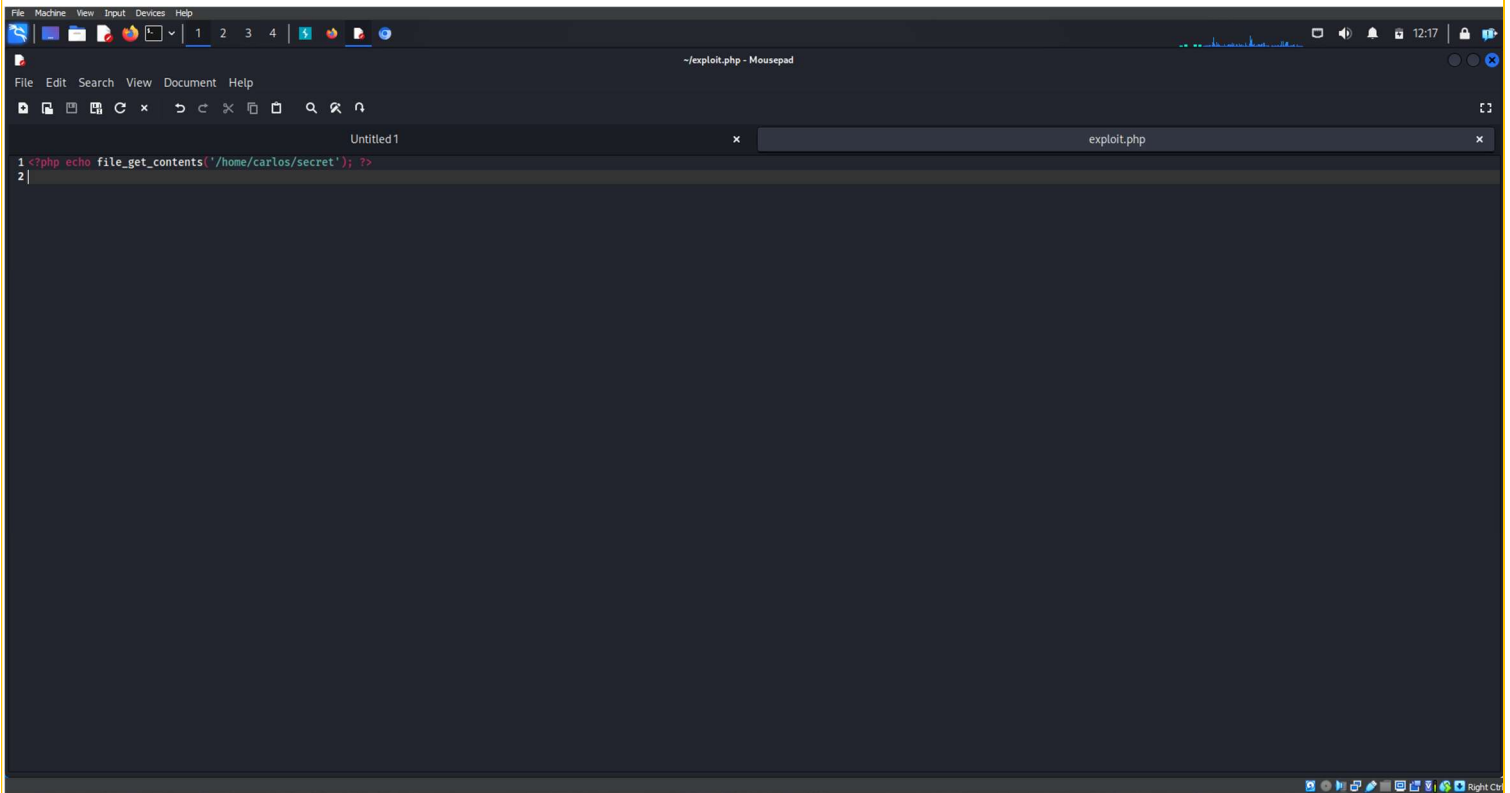
Inspector

Request attributes

Memory: 150.7MB

8th April 2024

Step 4: On your system, create a file called exploit.php, containing a script for fetching the contents of Carlos's secret file.

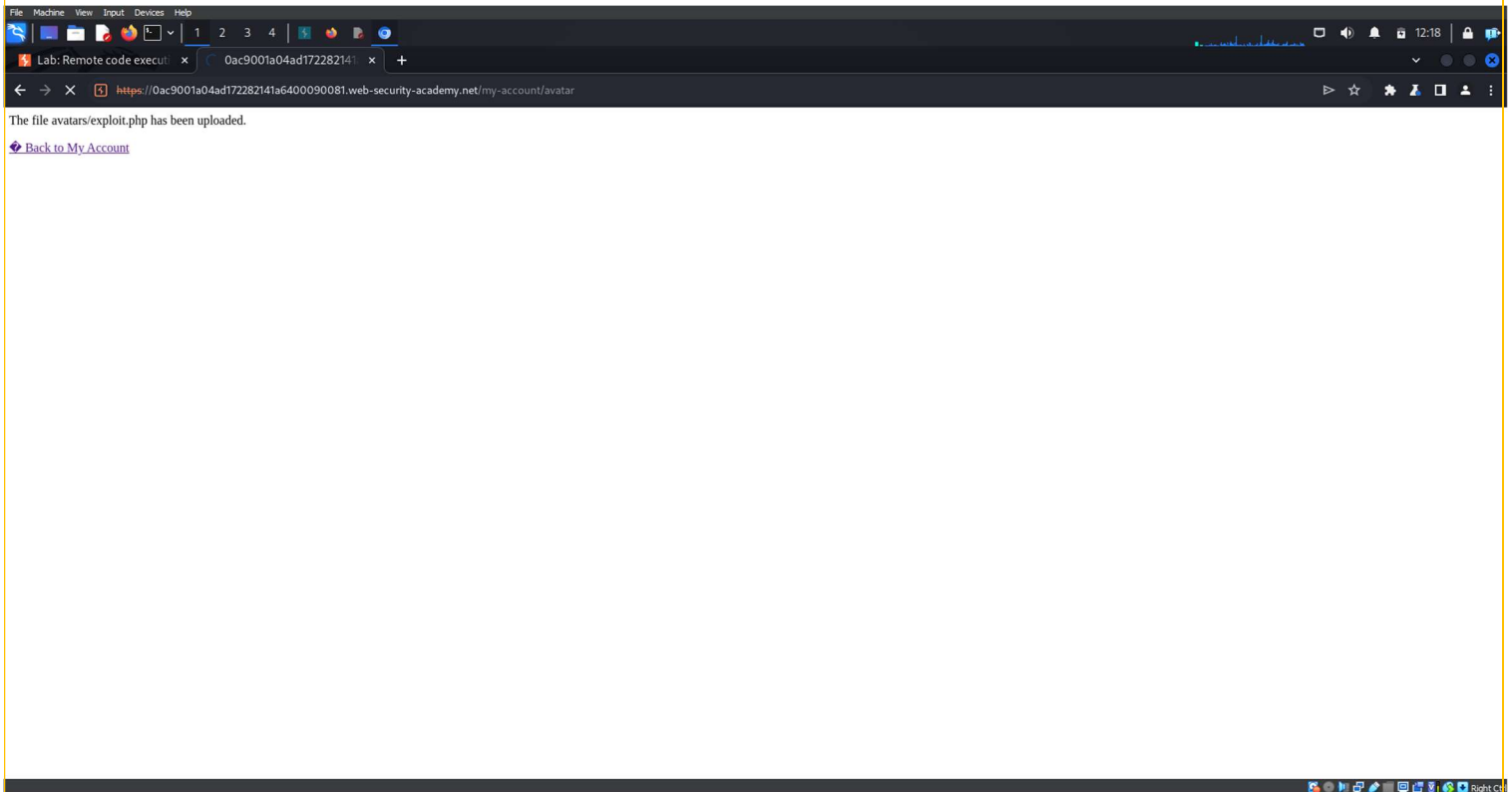
A screenshot of a Linux terminal window. The window has a title bar that reads "-/exploit.php - Mousepad". Below the title bar is a menu bar with "File", "Edit", "Search", "View", "Document", and "Help". The main area of the window is a dark-themed text editor. It shows two tabs: "Untitled1" and "exploit.php". The "exploit.php" tab is active, and it contains the following PHP code:

```
1 <?php echo file_get_contents('/home/carlos/secret'); ?>
2 |
```

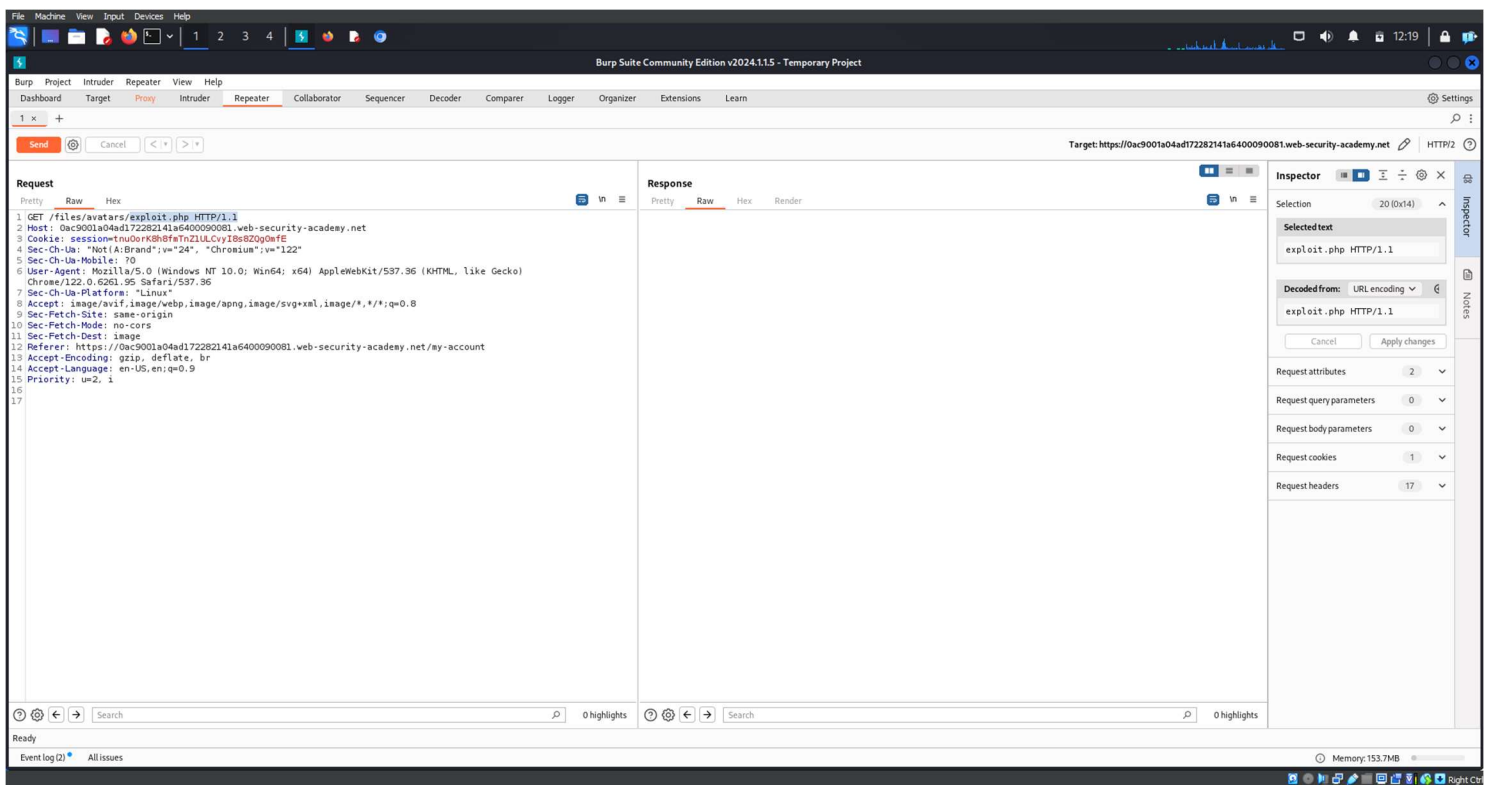
The cursor is at the end of the second line. The terminal window is open on a desktop environment, with various icons visible in the top and bottom panels.

8th April 2024

Step 5: Use the avatar upload function to upload your malicious PHP file. The message in the response confirms that this was uploaded successfully.

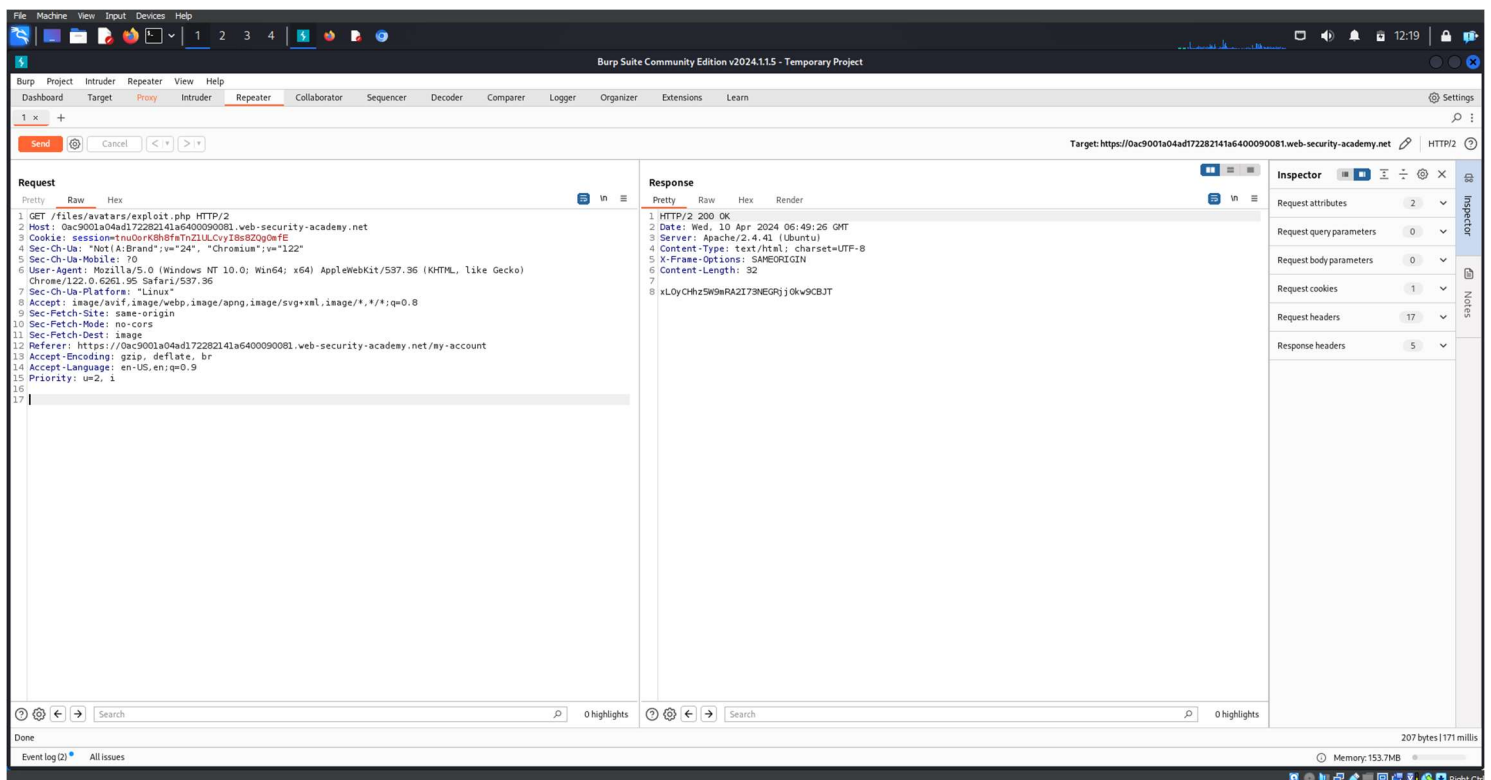


Step 6: In Burp Repeater, change the path of the request to point to your PHP file:

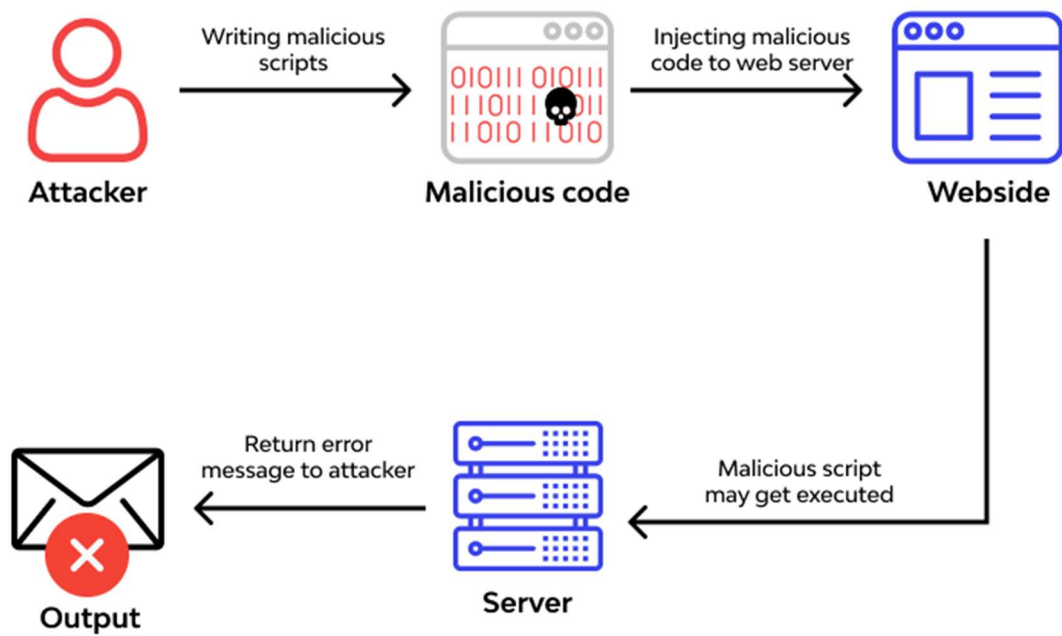


8th April 2024

Step 7: Send the request. Notice that the server has executed your script and returned its output (Carlos's secret) in the response. We successfully got the secret in response hence task completed successfully



7- Flow Diagram:



8- Recommendation: -

- 1- **Implement Strict File Type Validation**: Ensure that the image upload functionality strictly validates file types to only allow legitimate image files. Use server-side validation mechanisms and file type detection libraries to verify uploaded files.
- 2- **Secure File Permissions**: Set appropriate file permissions on the server to restrict access to sensitive directories and files. Limit the permissions of uploaded files to prevent execution of malicious scripts.
- 3- **Content Disposition Headers**: Set Content-Disposition headers to prevent browsers from executing uploaded files. This can help mitigate risks associated with file execution vulnerabilities.
- 4- **Use Content Security Policy (CSP)**: Implement CSP headers to restrict the sources from which content, such as scripts and stylesheets, can be loaded. This can help prevent cross-site scripting (XSS) attacks.
- 5- **Regular Security Audits**: Conduct regular security audits and penetration testing of the web application to identify and remediate vulnerabilities proactively.
- 6- **Secure Coding Practices**: Train developers on secure coding practices to avoid common security pitfalls, such as inadequate input validation and improper file handling.

9- Conclusion: -

In conclusion, the flow diagram highlights the potential risks associated with vulnerabilities in the image upload functionality of web applications. Exploiting these vulnerabilities can lead to unauthorized access to servers, execution of arbitrary commands, and exfiltration of sensitive data. To mitigate these risks, it is essential for organizations to implement robust security measures, including strict file type validation, secure file permissions, regular security audits, and adherence to secure coding practices. By prioritizing security measures and staying vigilant against emerging threats, organizations can bolster the resilience of their web applications and safeguard against potential exploitation.

References:-

- 1- Web security Academy
- 2- [Lab: Remote code execution via web shell upload | Web Security Academy \(portswigger.net\)](#)
- 3- [8.1 Lab: Remote code execution via web shell upload | 2023 | by Karthikeyan Nagaraj | Medium](#)