



Ciências  
ULisboa

# Sistemas de Segurança de Software

Mestrado em Engenharia Informática

## Segurança de Software

Mestrado em Informática

### Class Project: Experiments with WebGoat (class 2)

Gustavo Henriques Nº 64361

Leonardo Monteiro Nº 58250

Maria Figueirinhas Nº 46494

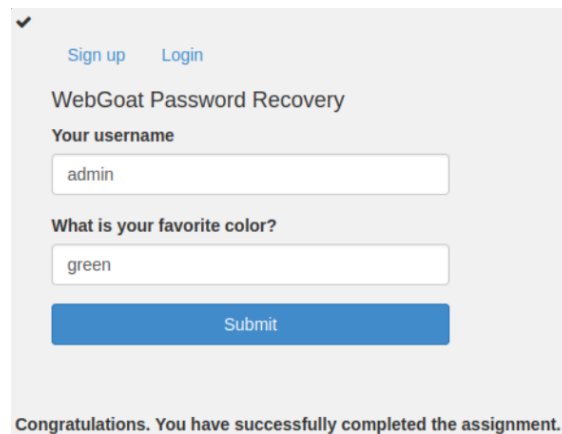
## 2. Broken Authentication

Solving Identity & Auth Failures >> Password Reset >> Steps 4.

Lesson 4: Retrieving the password.

- 1) Entre os utilizadores "tom," "admin" e "larry," a conta "admin" é a mais relevante para comprometer, pois geralmente possui os maiores privilégios e controlo sobre o sistema, tornando-a a mais suscetível e valiosa num ataque.

- 2) Para este exercício fizemos brute force para a cor do admin e descobrimos que era a verde.



✓

[Sign up](#) [Login](#)

WebGoat Password Recovery

Your username

admin

What is your favorite color?

green

Submit

Congratulations. You have successfully completed the assignment.

- 3) As perguntas que estão no passo 5 são as seguintes:



What is your favorite animal?

In what year was your mother born?

What was the time you were born?

What is the name of the person you first kissed?

What was the house number and street name you lived in as a child?

In what town or city was your first full time job?

In what city were you born?

On which wrist do you wear your watch?

What was the last name of your favorite teacher in grade three?

What is the name of a college/job you applied to but didn't attend?

What are the last 5 digits of your drivers license?

What was your childhood nickname?

Who was your childhood hero?

What is your favorite color?

What is your favorite animal? check

Muitas dessas perguntas de segurança apresentam vulnerabilidades porque as respostas podem ser descobertas através de informações públicas ou deduzidas por pessoas próximas. Dados como o nome do teu primeiro emprego, o apelido de infância ou a cidade onde nasceste são, muitas vezes, partilhados em redes sociais ou acessíveis por conhecidos. Este tipo de perguntas, ao depender de respostas que podem ser obtidas ou adivinhadas, reduz a eficácia das perguntas de segurança, tornando-as um alvo fácil para ataques de engenharia social.

Solving Identity & Auth Failures >> Secure Passwords >> Step 4.

**R:** Primeiro começamos por testar umas das passwords sugeridas e observamos que eram muito fracas:

password ☐ Show password

**Submit**

You have failed! Try to enter a secure password.

Your Password: \*\*\*\*\*

Length: 8

Estimated guesses needed to crack your password: 3

Score: 0/4

Estimated cracking time: 0 years 0 days 0 hours 0 minutes 0 seconds

Warning: This is a top-10 common password.

Suggestions:

- Add another word or two. Uncommon words are better.

Score: 0/4

Ao metermos a pass 'Oteugato\_123' conseguimos chegar a uma password que é 4/4 segura.

✓ Enter a secure password... ☐ Show password

**Submit**

You have succeeded! The password is secure enough.

Your Password: \*\*\*\*\*

Length: 12

Estimated guesses needed to crack your password: 100000010000

Score: 4/4

Estimated cracking time: 317 years 35 days 18 hours 3 minutes 20 seconds

Score: 4/4

Solving Sensitive Data Exposure >> Insecure Login >> step 2

**R:** Para esta lição o que fizemos foi pôr um break entre a aplicação e o servidor e, ao carregar no botão de login foi feita, desta forma, uma request para o servidor. Depois disto fomos avançando passo a passo até aparecer os dados do user no proxy, como é possível ver abaixo:

Content-Length: 50  
Origin: https://127.0.0.1:8080  
DNT: 1  
Connection: keep-alive  
Referer: https://127.0.0.1:8080/WebGoat/start.mvc?username=guestss

{ "username": "CaptainJack", "password": "BlackPearl" }

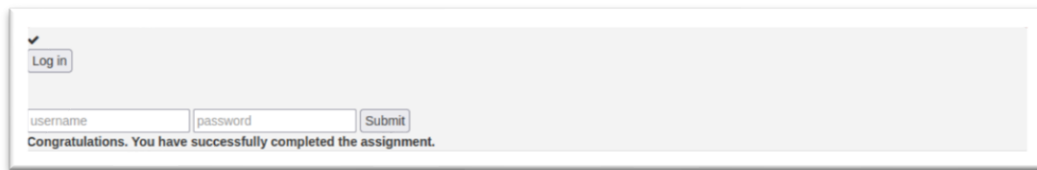
← Response

Header: Text Body: Text

HTTP/1.1 405 Method Not Allowed

⚠ Content Modified

Com isto foi só pôr estes dados nos campos de input que apareciam na lição e completamos assim a mesma:



✓  
Log in

username password Submit

Congratulations. You have successfully completed the assignment.

Solving Identity & Auth Failure >> Authentication Bypasses >> step 2

1) Começamos por responder às perguntas da seguinte maneira:



Verify Your Account by answering the questions below:

What is the name of your favorite teacher?

test

What is the name of the street you grew up on?

test

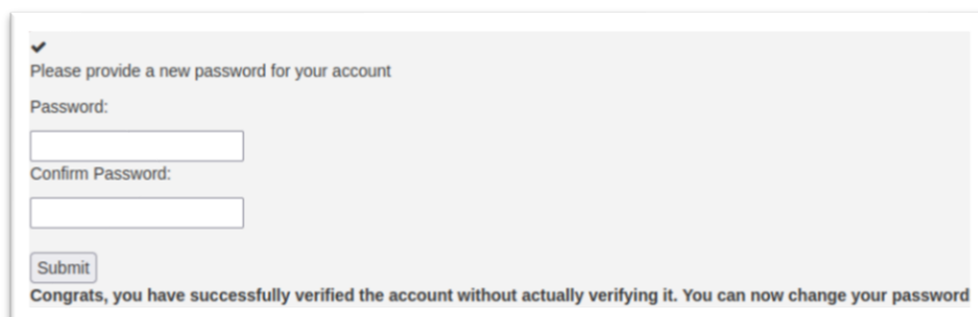
Submit

Not quite, please try again.

2-3) Para este exercício o que fizemos foi apanhar a request ao carregar no botão Submit no proxy. De segui alteramos o nome dos campos, desta maneira é gerado um erro do servidor que nos deixa fazer o login à mesma, porque os campos não foram apagados, mas sim foram alterados os seus nomes:

```
secQuestion2=test&secQuestion3=test&jsEnabled=1&verifyMethod=SEC_QUESTIONS&userId=12309746
```

Com isto conseguimos completar esta passo:



✓  
Please provide a new password for your account

Password:

Confirm Password:

Submit

Congrats, you have successfully verified the account without actually verifying it. You can now change your password

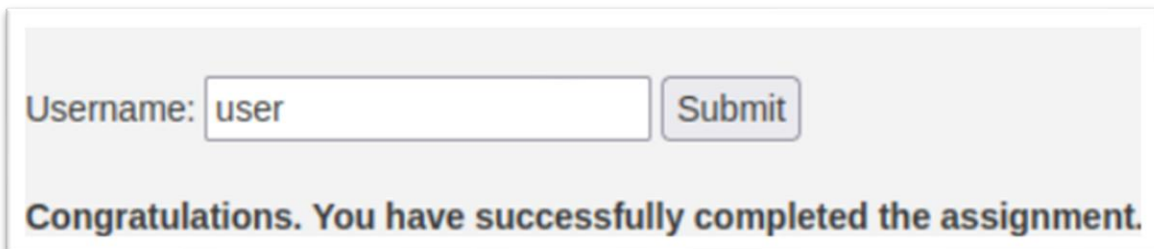
Solving Identity & Auth Failure >> JWT tokens >> step 4

Lesson 4: Finding out the username.

Para isto abrimos o WebWolf fizemos login com username guestss e pass guestss. De seguida abrimos o JWT e metemos para lá o token e conseguimos assim obter o username:



Depois disto metemos o username no input pedido e completamos assim mais uma lição:

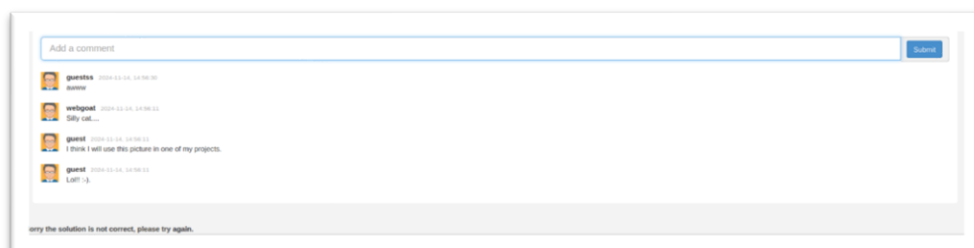


### 3. XML External Entities (XXE)

Solving Security Misconfigurations >> XXE >> step 4.

Lesson 4: Listing all the files in the root directory.

Nesta lição começamos por enviar um simples comentário e obtivemos a seguinte conversa:



Depois disto intercetamos a request no proxy e mudamos para o seguinte:

```
<?xml version="1.0"?>
<!DOCTYPE foo [
    <!ENTITY xxe SYSTEM "file:///">
]>
<comment>
    <text>&xxe;</text>
</comment>
```

Com isto conseguimos obter a lista dos ficheiros que estão na root, como mostra a imagem abaixo:



## 4. Server-Side Request Forgery (SSRF)

Solving Security Misconfigurations >> XXE >> step 4.

Lesson 2: Retrieving the jerry picture.

Nesta lição intercetamos mais uma vez o request no proxy e alteramos o nome de tom para jerry, como mostra a imagem:



Depois disto voltamos ao WebGoat e conseguimos ver a imagem do jerry:



Lesson 3: Getting information from <http://ifconfig.pro>.

Para esta lição, mais uma vez interceptamos a request no proxy e alteramos o url para o que mostra a imagem:



Depois disto voltamos ao WebGoat e vimos que conseguimos obter informação do <http://ifconfig.pro>, como mostra a imagem:

