

Desenvolvimento Web Seguro

Semana 02 - Atividade Prática

Prof. Jefferson O. Andrade

2024-05-15

1 Parte 1 - Resolução de Problemas

1.1 Instruções

Seu grupo deve analisar a aplicação de exemplo, identificar possíveis vulnerabilidades e discutir possíveis soluções.

1.1.1 Identificação de Vulnerabilidades:

- Analisar onde e como as senhas são armazenadas no banco de dados.
- Verificar as rotas e funções que controlam o acesso às funcionalidades da aplicação.

1.1.2 Correção das Vulnerabilidades:

- Implementar hashing seguro de senhas usando `werkzeug.security` (ou outra biblioteca de hashing).
- Melhorar o controle de acesso, garantindo que apenas usuários autorizados possam acessar determinadas funcionalidades.

1.2 Configuração do Ambiente

1.2.1 Instalar Python e pip

- Certifique-se de que você tenha o Python e o pip instalados no seu sistema. Você pode baixar o Python [aqui](#).

1.2.2 Clone o repositório do Codeberg

1. Acesse o repositório da disciplina no `Codeberg.org`.
2. Clone o repositório em algum local adequado no seu computador local:

```
git clone https://codeberg.org/profjeffandrade/secure-web-dev.git
```

3. Vamos chamar o diretório onde o você clonou o repositório de `$WD`.

1.2.3 Criar e Ativar um Ambiente Virtual

1. Navegue até o diretório `$WD/tasks`.
2. Execute os comandos abaixo:

```
python -m venv venv  
source venv/bin/activate # No Windows, use `venv\Scripts\activate`
```

1.2.4 Instalar Flask e Werkzeug

- Execute o comando abaixo:

```
pip install Flask Werkzeug
```

1.2.5 Inicializar o banco de dados.

- O banco de dados de exemplo (example.db) é iniciado com o usuário admin (senha admin123). Se precisar apagar o banco de dados e reiniciá-lo, execute o comando abaixo uma vez:

```
python app.py
```

1.2.6 Rodar a Aplicação:

- Após inicializar o banco de dados, você pode rodar a aplicação com:

```
flask run
```

1.2.7 Acessar a Aplicação

- Abra um navegador e vá para <http://127.0.0.1:5000/login> para acessar a aplicação.

2 Parte 2 - Implementação

Para a parte 2, seu grupo deverá desenvolver uma aplicação web simples seguindo as indicações abaixo:

1. Configuração do Ambiente:

- Configure um ambiente de desenvolvimento usando uma linguagem de programação e framework de sua escolha (por exemplo, Python com Flask, ou JavaScript com Node.js).

2. Autenticação:

- Implemente um sistema de login básico utilizando nome de usuário e senha.
- Adicione a funcionalidade de autenticação multifator (MFA) para maior segurança.

3. Autorização:

- Crie diferentes níveis de permissão (por exemplo, usuário comum, administrador).
- Implemente controles de acesso para restringir ações baseadas nas permissões do usuário.