Centro Universitário Álvares Penteado - FECAP

Criptografia | Cybersegurança e Defesa Cibernética

- Emily Mickeli Depine da Silva - 23025480
- Gustavo Henrique Santos Araújo - 23025397
- Fernando José dos Santos - 23025299
- Renan Teixeira Pinheiro - 23025274

Introdução

Com o avanço das ameaças digitais, proteger dados sensíveis se tornou essencial em qualquer aplicação web. No projeto **Comparadrive,** nosso site de comparação de preços entre apps de mobilidade urbana, optamos por aplicar criptografia das informações de endereço de partida e chegada inseridas pelos usuários, garantindo mais privacidade e segurança.

Criptografia Utilizada: Fernet

- Para proteger os dados, utilizamos o algoritmo **Fernet**, da biblioteca cryptography. Essa tecnologia aplica **criptografia simétrica**, ou seja, usa a **mesma chave** para cifrar e decifrar os dados.
- Fernet é baseado no padrão **AES (Advanced Encryption Standard)** no modo **CBC**, combinado com **HMAC-SHA256**, que garante a integridade dos dados. Essa combinação protege contra alterações não autorizadas e garante confidencialidade.
- A criptografia é aplicada assim que o usuário insere os endereços, mantendo essas informações protegidas durante todo o uso do sistema.

Função no Projeto - Comparadrive

No contexto do site Comparadrive:

- Protege os endereços inseridos pelos usuários.
- Garante que dados sensíveis não fiquem visíveis em plaintext.
- Simula acesso restrito com senha, como uma camada de autenticação para administradores

Justificativa técnica:

- Segurança embutida (criptografia + autenticação).
- Evita erros manuais como IV fixo ou ausência de verificação.
- Formato seguro e padronizado com timestamp.

```
from cryptography.fernet import Fernet
from IPython.display import clear_output

def limpar_console():
    clear_output(wait=True)

chave = Fernet.generate_key()
fernet = Fernet(chave)

endPartida = input("Digite o endereço de partida: ")
endChegada = input("Digite o endereço de chegada: ")

endPartida_cripto = fernet.encrypt(endPartida.encode())
endChegada_cripto = fernet.encrypt(endChegada.encode())

limpar_console()

print("\nEndereço Partida (criptografado):", endPartida_cripto.decode())
print("Endereço Chegada (criptografado):", endChegada_cripto.decode())

print("\nVocê é o administrador?")
print("1 - Sim\n2 - Não")
resposta = input("Escolha uma opção: ")
```

```python
if resposta == "1":
    while True:
        senha = input("Digite a senha do administrador: ")
        if senha == "1234":
            endPartida_original = fernet.decrypt(endPartida_cripto).decode()
            endChegada_original = fernet.decrypt(endChegada_cripto).decode()
            print("\nEndereço de Partida (original):", endPartida_original)
            print("Endereço de Chegada (original):", endChegada_original)
            break
        else:
            print("Senha incorreta. Tente novamente.")
else:
    print("Programa finalizado.")
```

```
Endereço Partida (criptografado): gAAAAABoJ9HL0X16uOautiHLG_wZ7mS0tMbMTxTrhfp6256kW9E0ybSnC4IOQRHVgmdJ9yLnrVI0z7WZDUzOEWpQZCkOSdDPnA==
Endereço Chegada (criptografado): gAAAAABoJ9HL7r_NMOnw-A3_EZhGzkKl2WZ-3Es0Ff6y_hgbP7141T6_ozoY2guc67UHoOti_aZOWUbCZDte2oNkYMALCOW1XA==

Você é o administrador?
1 - Sim
2 - Não
Escolha uma opção: 5
Programa finalizado.
```

Armazenamento Seguro da Chave de Criptografia

Embora a chave Fernet seja gerada automaticamente no nosso ambiente de testes, em um sistema real ela deve ser **armazenada com segurança**, como em variáveis de ambiente ou serviços de gerenciamento de segredos (ex: AWS Secrets Manager).

Isso garante que os dados criptografados possam ser acessados no futuro e que a chave não fique exposta no código, alinhando o sistema a boas práticas de segurança e à **LGPD (Lei Geral de Proteção de Dados)**.

Controle de Acesso e Autenticação

O sistema conta com uma **simulação de autenticação**, onde apenas o "administrador" que souber a senha pode descriptografar os endereços. Apesar de simples, esse recurso demonstra a importância do **controle de acesso** aos dados sensíveis.

Em produção, essa autenticação deve ser reforçada com **hash seguro de senhas** (como bcrypt) e, se possível, **autenticação multifator (MFA)**. Isso evita acessos indevidos e reforça a proteção dos dados dos usuários.

Considerações Finais

A implementação da criptografia no Comparadrive é um passo importante para tornar o sistema mais seguro. O uso do Fernet garantiu confidencialidade e integridade dos dados de forma prática e eficaz.

Apesar das simplificações feitas para fins didáticos (como geração da chave em tempo real e senha fixa), o projeto mostra a importância de **proteger informações sensíveis** desde o início. Para um ambiente real, recomendamos melhorar o armazenamento de chaves, usar autenticação robusta e aplicar práticas modernas de segurança digital.