UNIVERSIDADE FEDERAL DO PIAUÍ
CAMPUS MINISTRO PETRÔNIO PORTELLA
CURSO DE CIÊNCIA DA COMPUTAÇÃO

CRIPTOGRAFIA:

SISTEMA DE GERENCIAMENTO DE
CHAVES PÚBLICAS E CRIPTOGRAFIA

BIANCA BASTOS DE SOUZA
JOÃO PAULO ROCHA DE OLIVEIRA
SAMUEL DA SILVA CRUZ

TERESINA – PI
2024

RESUMO

Este projeto tem como objetivo desenvolver um sistema abrangente para o gerenciamento de pares de chaves criptográficas, incluindo tanto chaves públicas quanto privadas. O sistema será capaz de realizar diversas operações essenciais para a segurança da informação, tais como a geração de novos pares de chaves, armazenamento seguro das chaves, bem como a importação e exportação das mesmas. Além disso, o projeto inclui a implementação de funcionalidades para criptografia e descriptografia de arquivos, utilizando os pares de chaves gerados.

Dessa forma, o sistema não apenas facilita a criação e o gerenciamento de chaves criptográficas, mas também garante a segurança dos dados através de operações criptográficas robustas.

SUMÁRIO

1 INTRODUÇÃO

O trabalho visa criar um sistema de gerenciamento de chaves criptográficas que permite gerar, carregar, salvar, listar, buscar, apagar chaves e criptografar/descriptografar mensagens utilizando essas chaves.

2 BIBLIOTECAS UTILIZADAS

As bibliotecas utilizadas são componentes do pacote “cryptography” em Python e servem para diferentes propósitos relacionados à criptografia de dados. Essas bibliotecas em conjunto fornecem um conjunto poderoso de ferramentas para a implementação de um sistema de criptografia robusto.

2.1 “CRYPTOGRAPHY.HAZMAT.PRIMITIVES.SERIALIZATIONS”

Essa sub-biblioteca é usada para a serialização e desserialização de chaves criptográficas. A serialização é o processo de converter uma chave em um formato que pode ser facilmente armazenado ou transmitido (como PEM ou DER). Desserialização é o processo inverso, ou seja, converter o formato serializado de volta para um objeto de chave utilizável.

Exemplo de uso:

• Serializar uma chave privada para armazená-la em um arquivo.

• Carregar uma chave pública de um arquivo para realizar operações criptográficas.

2.2 "CRYPTOGRAPHY.HAZMAT.BACKENDS.DEFAULT_BACKEND”

Essa biblioteca fornece o backend padrão para operações criptográficas. Backends são implementações específicas dos algoritmos criptográficos fornecidos pelo pacote cryptography. O backend padrão geralmente utiliza bibliotecas nativas otimizadas para a plataforma em que o código está sendo executado, garantindo assim segurança e desempenho adequados.

Exemplo de uso:

• Especificar o backend ao gerar ou carregar chaves criptográficas.

2.3 "CRYPTOGRAPHY.HAZMAT.PRIMITIVES.ASYMMETRIC.RSA”

Essa sub-biblioteca fornece funcionalidades para criar e manipular chaves RSA (Rivest-Shamir-Adleman), que é um dos algoritmos de criptografia assimétrica mais populares. A criptografia assimétrica usa um par de chaves (pública e privada) para criptografar e descriptografar dados.

Exemplo de uso:

• Gerar um novo par de chaves RSA.

• Criptografar dados com a chave pública.

• Descriptografar dados com a chave privada.

2.4 "CRYPTOGRAPHY.HAZMAT.PRIMITIVES.ASYMMETRIC.PADDING”

Essa biblioteca define esquemas de padding (preenchimento) para algoritmos de criptografia assimétrica. Padding é necessário para garantir que os dados criptografados tenham um formato seguro e adequado para operações criptográficas.

Exemplo de uso:

• Aplicar um esquema de padding ao criptografar ou descriptografar dados com RSA (como OAEP ou PSS).

2.5 "CRYPTOGRAPHY.HAZMAT.PRIMITIVES.HASHES"

Essa sub-biblioteca oferece uma coleção de algoritmos de hash, que são usados para garantir a integridade dos dados. Funções de hash mapeiam dados de tamanho arbitrário para um valor de tamanho fixo (o hash), que deve ser único para diferentes dados de entrada.

Exemplo de uso:

• Criar um hash de uma mensagem ou arquivo para verificar sua integridade.

• Usar algoritmos de hash em esquemas de padding (por exemplo, OAEP com SHA-256).

2.6 "OS”

Essa é uma biblioteca padrão do Python que fornece uma maneira de interagir com o sistema operacional. Ela é frequentemente usada em operações de criptografia para gerenciar arquivos de chaves, diretórios, e outros recursos do sistema.

Exemplo de uso:

• Ler e escrever arquivos que contêm chaves criptográficas.

• Gerenciar permissões de arquivos para garantir que as chaves sejam armazenadas com segurança.

3 INSTRUÇÕES PARA UTILIZAÇÃO DO PROGRAMA

Para utilizar o programa o usuário deve executar o arquivo “main.py” através de uma IDE que lhe permita executar códigos escrito na linguagem Python. Caso utilize o Visual Studio Code (VSCode) a opção deverá aparecer no canto superior direito como ilustrado na Imagem 1.

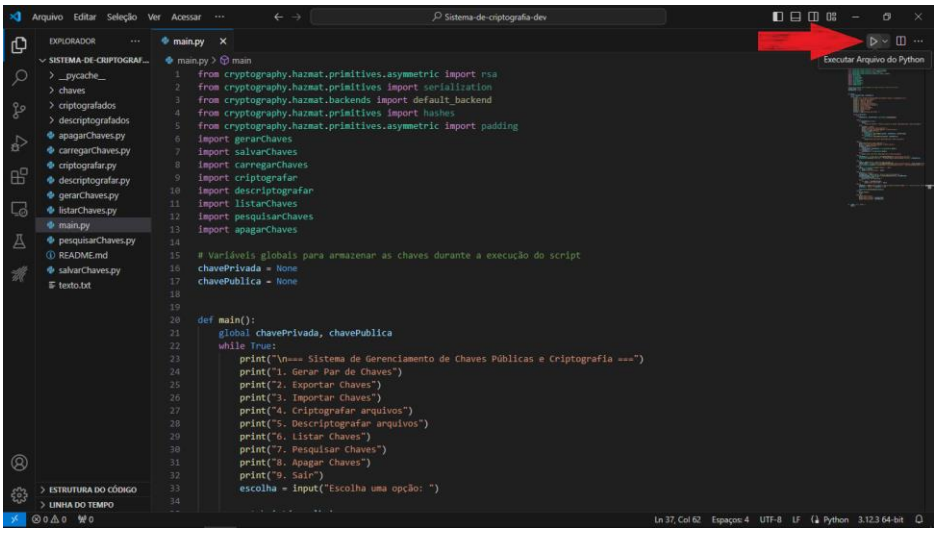

Imagem 1 – Executar Arquivo do Python

Após executar o código um terminal irá se abrir na parte inferior da tela do VSCode com as opções oferecidas pelo programa como mostra a Imagem 2.

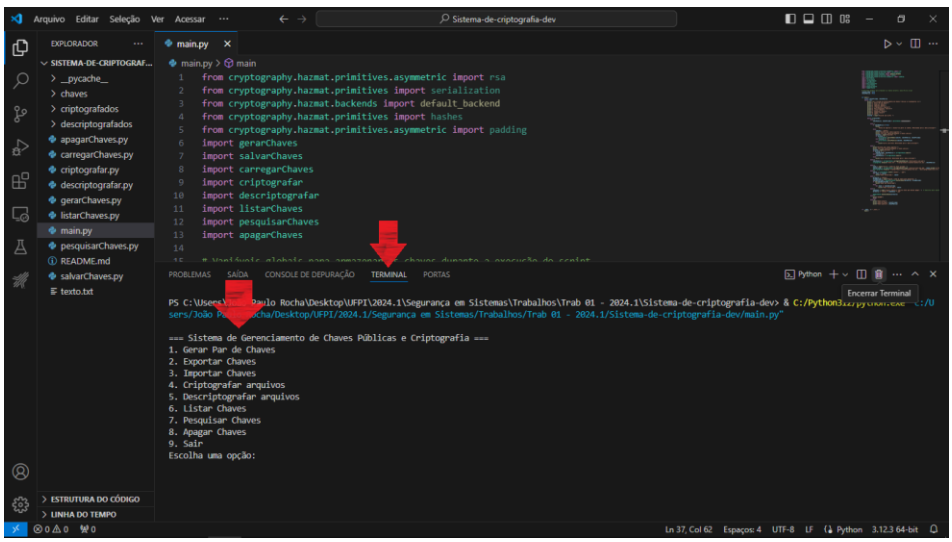

Imagem 2 – Terminal VSCode

Feito isso, basta navegar pelas opções desejadas para interagir com o programa.

3.1 GERAR PAR DE CHAVES

Esta opção irá gerar um par de chaves RSA (pública e privada). Primeiramente é solicitado que o usuário digite um nome para identificar a chave, caso já exista uma chave com o nome, o programa informa ao usuário, solicita que que ele digite um outro nome e em seguida é solicitada uma senha para proteger a chave privada. Note que a senha não aparece no terminal, porém, o programa consegue identificar o que o usuário digita mesmo assim. Esta foi uma abordagem tomada visando a privacidade do usuário. É feita então uma serialização das chaves em formato .PEM.

```
=== Sistema de Gerenciamento de Chaves Públicas e Criptografia ===
1. Gerar Par de Chaves
2. Exportar Chaves
3. Importar Chaves
4. Criptografar arquivos
5. Descriptografar arquivos
6. Listar Chaves
7. Pesquisar Chaves
8. Apagar Chaves
9. Sair
Escolha uma opção: 1
Digite o nome para as chaves: joao
Chave já existe
Digite o nome para as chaves: samuel
Defina a senha para a chave privada:
Chaves geradas e salvas com sucesso!
```

Imagem 3 – Gerar Par de Chaves

3.2 EXPORTAR CHAVES

Ao selecionar essa opção o programa irá informar ao usuário que é necessário gerar um par de chaves antes caso ele ainda não o tenha feito, caso contrário o programa irá perguntar se o usuário deseja exportar as duas chaves ou apenas a chave pública. Se o usuário escolher exportar apenas a chave pública, o programa irá

salvá-la no diretório “chaves_externas”. Por outro lado, se o usuário optar por exportar ambas as chaves, elas serão salvas no diretório “chaves”.

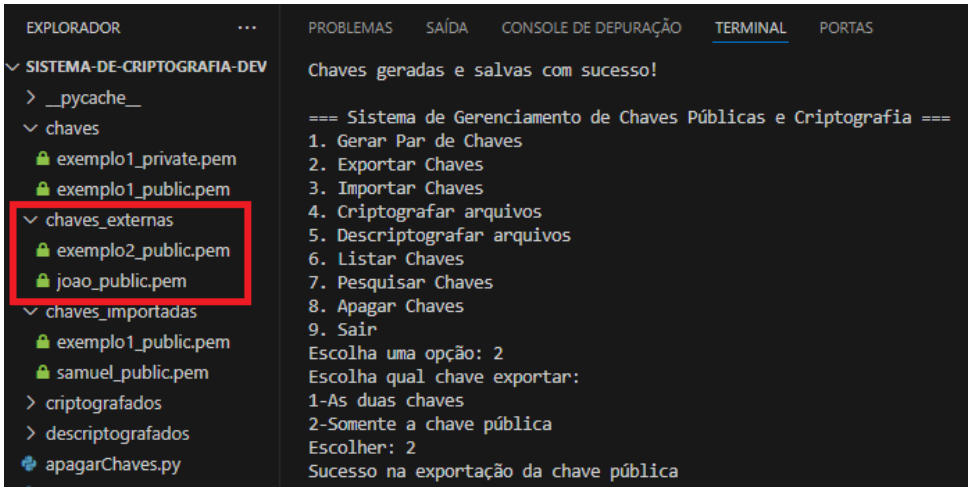

Imagem 4 – Exportar Chave pública

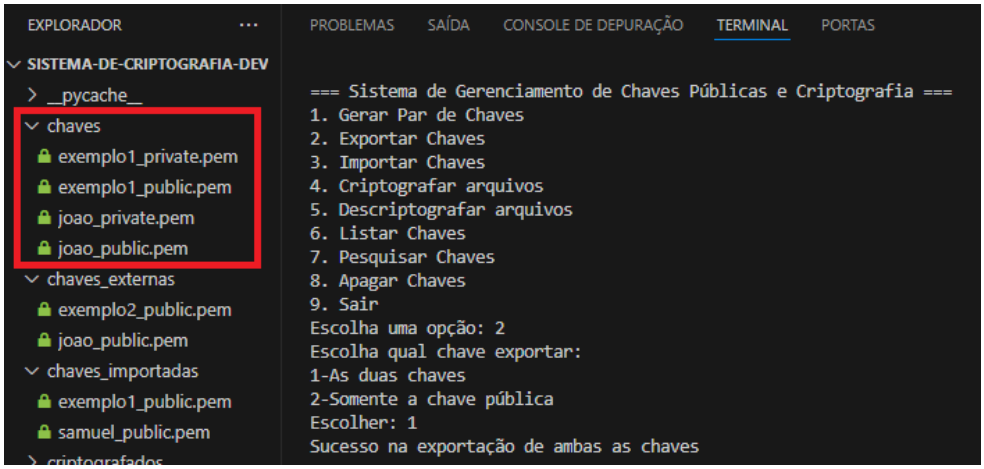

Imagem 5 – Exportar ambas as Chaves

3.3 IMPORTAR CHAVES

Nessa opção, o programa exibe uma lista de chaves públicas encontradas no diretório “chaves_externas” e solicita que o usuário digite o nome da chave que ele deseja importar. Após o usuário informar a chave desejada, o programa move o arquivo para o diretório “chaves_importadas”.

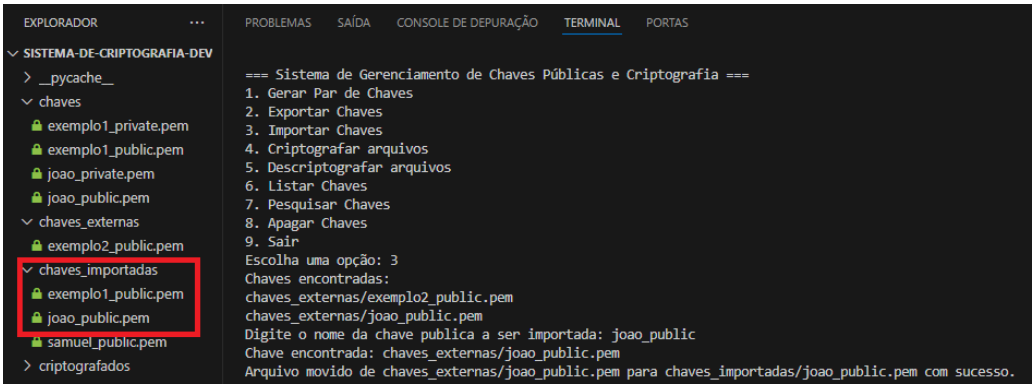

Imagem 6 – Importar Chaves

3.4 CRIPTOGRAFAR ARQUIVOS

Ao selecionar a opção de Criptografar arquivos, o programa pergunta se o usuário deseja fazer a criptografia utilizando uma chave pessoal ou uma chave importada. Após o usuário escolher uma das opções o programa lista todas as chaves disponíveis para a opção selecionada e pede pra que o usuário escolha uma delas. Depois que o usuário informa a chave, o programa executa a criptografia.

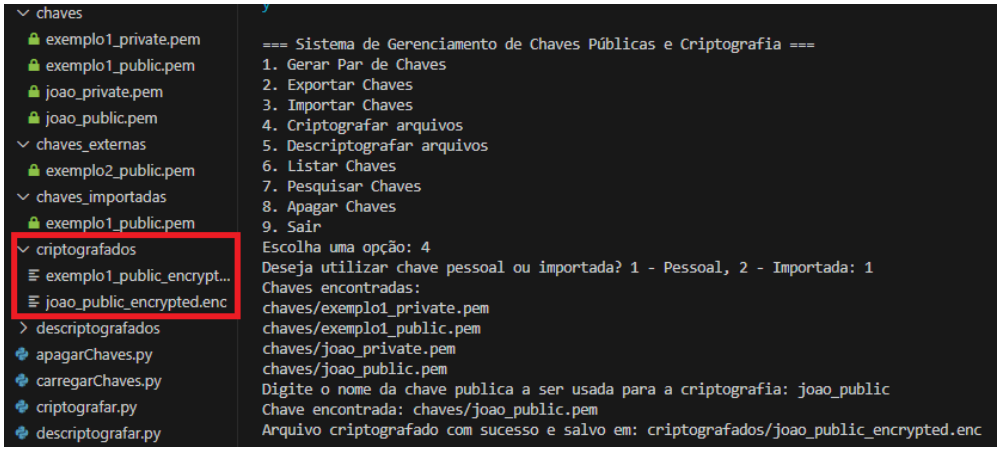

Imagem 7 – Criptografar Arquivos

3.5 DESCRIPTOGRAFAR ARQUIVOS

A opção de Descriptografar arquivos lista para o usuário todos os arquivos criptografados encontrados e pede para que ele selecione um dos arquivos, e em seguida pede para que ele escolha a chave privada para descriptografar o arquivo. Note que a chave deve ser o par da chave pública usada para a criptografia. Será solicitada a senha do usuário para confirmar a criptografia. Feito isso, o arquivo será descriptografado e salvo no repositório “descriptografados”.

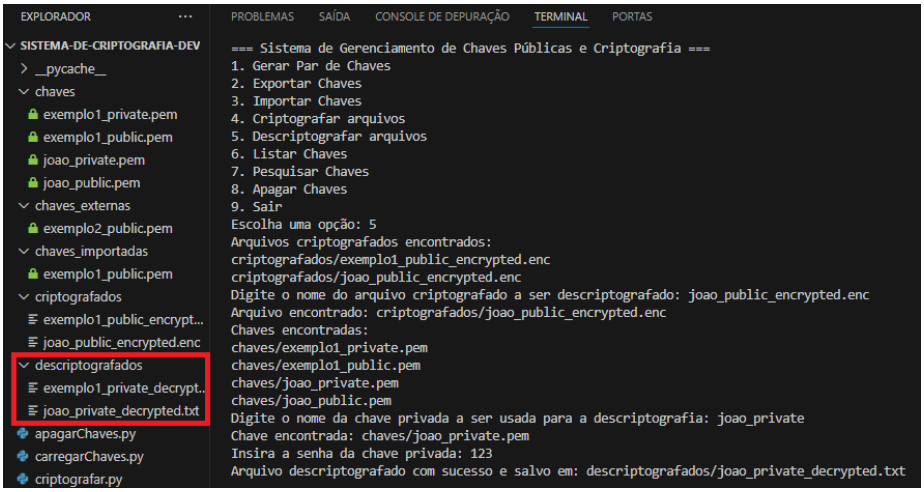

Imagem 8 – Descriptografar Arquivos

3.6 LISTAR CHAVES

O programa mostra uma lista com todas as chaves cadastradas.

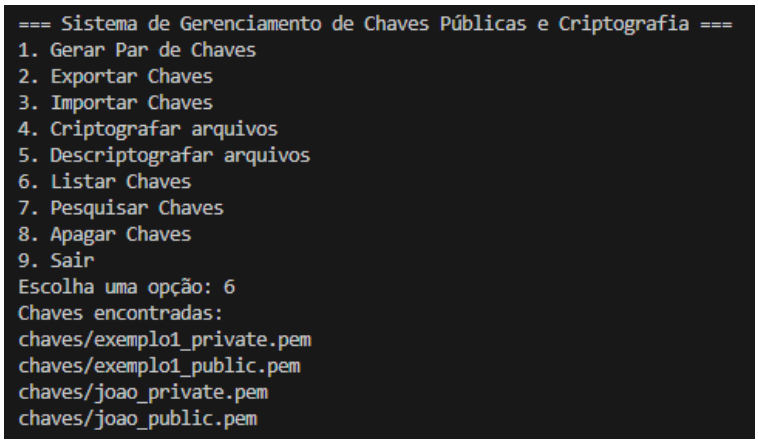

Imagem 9 – Listar Chaves

3.7 PESQUISAR CHAVES

Ao escolher a opção “pesquisar chaves” o programa irá solicitar ao usuário o nome de uma chave cadastrada (o nome informado deve conter a extensão .pem). Caso a chave não exista é exibida a mensagem “Chave não encontrada”.

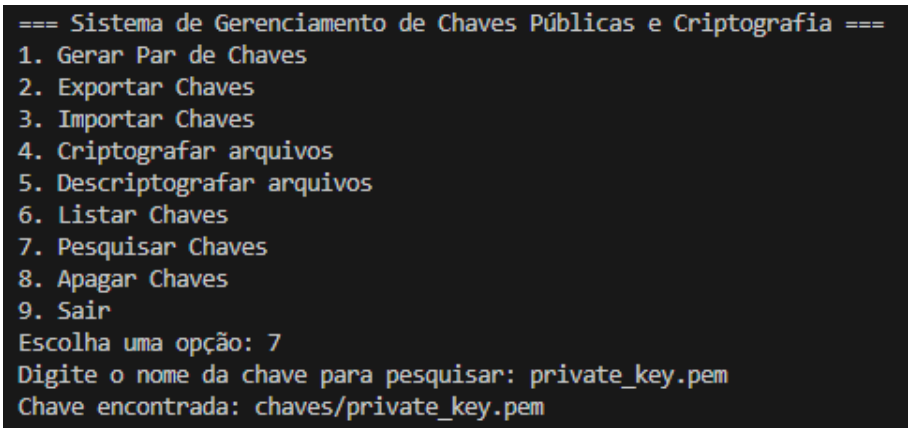

Imagem 10 – Pesquisar Chaves

3.8 APAGAR CHAVES

Ao selecionar a opção, o usuário deverá informar o nome da chave que deseja apagar, caso a chave não exista, o programa irá informar que o arquivo da chave não foi encontrado e irá retornar ao menu principal. Caso contrário, a chave será apagada do diretório para o qual havia sido exportada anteriormente.

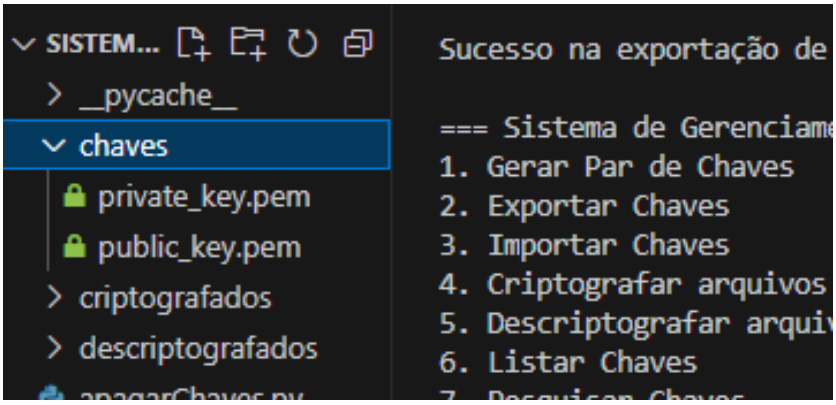

Imagem 11 – Apagar Chaves

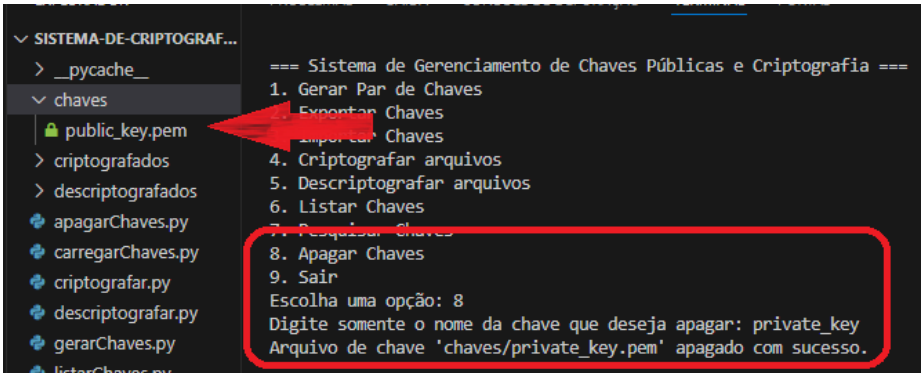

Imagem 12 – Apagar Chaves

3.9 SAIR

O programa é encerrado.