ATIVIDADE FORMATIVA: SEMANA 6

Criptografia simétrica e assimétrica

A atividade formativa proposta para esta semana visa trabalhar com os principais algoritmos de criptografia simétrica e assimétrica. O propósito dessa aula é utilizar de maneira prática os algoritmos de criptografia, conhecendo os parâmetros, vantagens e limitações de cada um dos algoritmos.

Cenário:

Como a segurança dos sistemas computacionais no banco de Tóquio foi comprometida é necessário a substituir todas as chaves públicas e privadas do sistema, vamos adicionar algoritmos de criptografia mais robustos. Para isso seu desafio é desenvolver um programa que utilize a criptografia de chave pública e simétrica.

Especificação da Atividade:

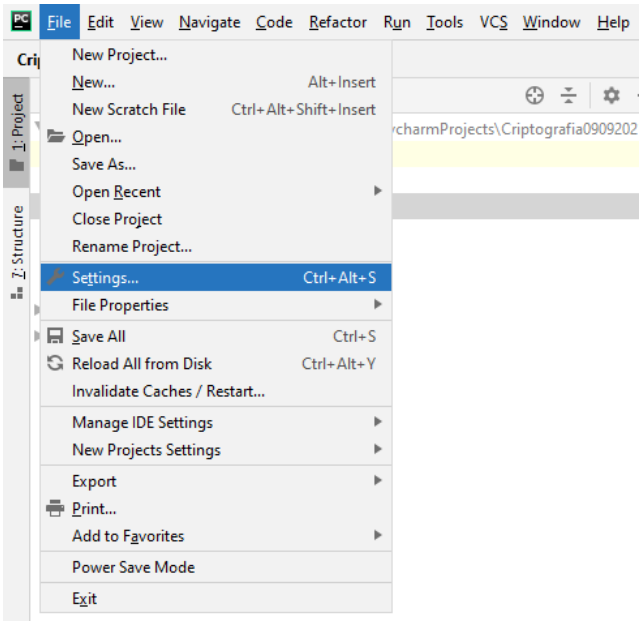
Nesta atividade você deverá cifrar uma mensagem utilizando os algoritmos de criptografia RSA e AES. O algoritmo RSA deve ser usado para chave pública e privada considerando diferentes tamanhos (1024, 2048, 4096 e 8192 bits) para o algoritmo AES utilizar a chave de 16 bytes. Por meio deste experimento você conseguirá notar que o desempenho dos algoritmos de criptografia com chave pública e privada está diretamente relacionado ao tamanho da chave.

Dicas:

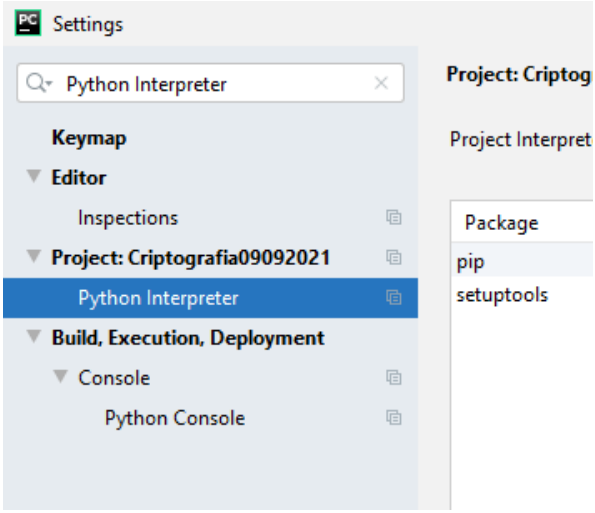
Na realização dos experimentos, para te auxiliar, está sendo fornecido a implementação dos algoritmos de criptografia RSA e AES, implementados na linguagem de programação Python. Para executar os algoritmos é necessário instalar a biblioteca “pycryptodome”. Se você deseja realizar a atividade no GoogleColab, deverá executar o comando seguir:

```
!pip install pycryptodome
```

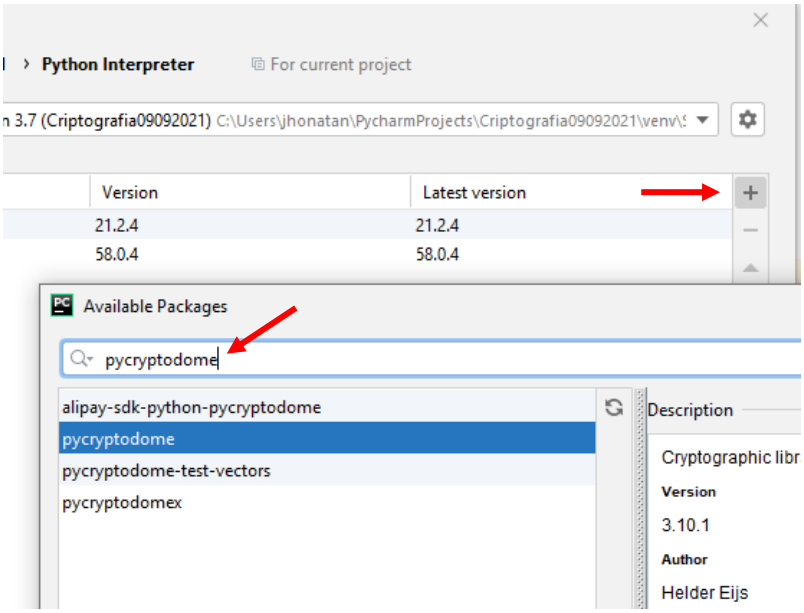
A atividade também poderá ser realizada na IDE do Pycharm. Para tal, crie um novo projeto no ambiente, acesse no menu principal “File”, clique na opção “Settings”.



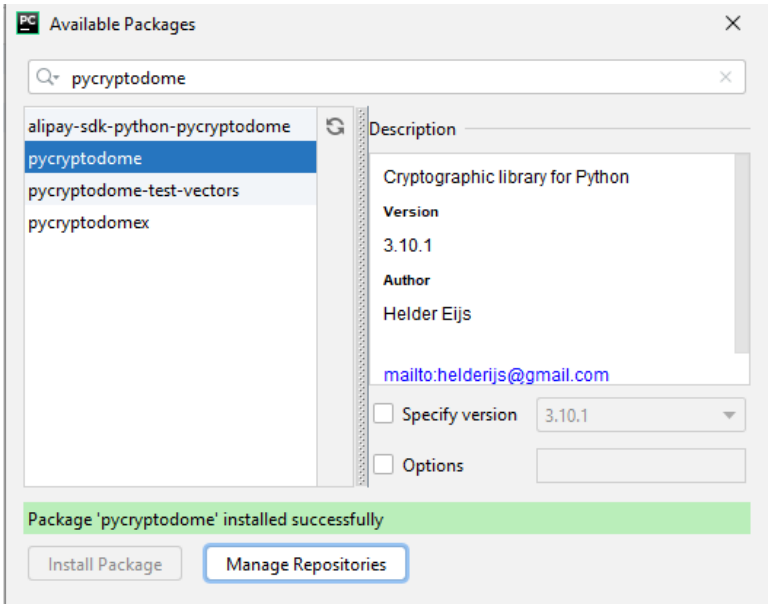
Na janela “Settings” na opção de buscar digite: “Python Interpreter”.



Clique na opção incluir (+), será aberto a janela referente aos pacotes disponíveis, então digite “pycryptodome”.



Efetue a instalação do pacote. Agora você poderá executar os algoritmos criptografia RSA e AES, correspondente aos arquivos “CriptRSA.py” e “CriptAES.py”



Para obter o desempenho dos algoritmos de criptografia RSA e AES, está sendo fornecido como exemplo o código implmenetado no arquivo “Benchmark.py”, disposto na figura a seguir.

Benchmark.py

```python
import time

inicio = time.time()
print("Vamos ajudar o banco de Tóquio!!")
fim = time.time()
benchmark = fim - inicio

print("Benchmark: ",benchmark)

```

Este código deverá ser integrado dentro dos arquivos “CryptRSA.py” e “CryptAES.py”. Modifique os arquivos para obter o desempenho de cada algoritmo. Alterar o tamanho da chave conforme o item de especificação, anote o tempo decorrido de cada execução e compare o desempenho dos algoritmos.

No código do algoritmo RSA para alterar o tamanho das chaves modifique a linha a seguir, o tamanho de chave diferente em cada execução (respectivamente 1024, 2048, 4096 e 8192 bits):

```
keyPair = RSA.generate(1024)
```

Acesse o código fonte (base) que está em anexo.