Universidade de Brasília – UnB

Departamento de CIC

Segurança Computacional

Alunos: Alexandre Victor Curcino Vasconcelos Cruvinel e Lucas Gonçalves Morais.

Matrículas: 19/0023546 e 19/0033240.

Relatório do Trabalho 2

1. Introdução

A princípio o RSA é um método de criptografia assimétrica mais usado no mundo, por meio da internet. É assimétrico porque usa chave pública e privada, isso permite uma segurança maior na troca de dados e integridade de informações. Além disso, o RSA, para obter uma segurança maior pode trocar mensagens em que são marcadas com uma assinatura, permitindo assim que os originadores criem mensagens inteligíveis apenas para os destinatários pretendidos.

2. Arquitetura

O projeto foi feito na linguagem Python, a pasta Gerador-Verificador-de-Assinaturas-RSA-Arquivos contém dois arquivos, o primeiro tem o mesmo nome da pasta que é onde tem todas as funções segundo as especificações os quais foram solicitadas, já o segundo arquivo.py define as interfaces para criar um arquivo de texto usado nos contextos de criptografia e tem funções para entradas e saídas.

Além disso, tivemos que fazer uma serialização desses dados, porque sob outra perspectiva não iríamos conseguir cifrar ou decifrar arquivos, mas também, precisamos do tamanho da chave o qual foi gerada junto com os dados sobre a paginação (padding do arquivo) que foi cifrado. Já o arquivo.py permite colocar os metadados em arquivos.txt gerados gravando-os em disco.

3. Limitações

Se for colocado texto com acentuação para ser cifrado vão ser gerados caracteres estranhos no momento da decifração, esse problema

somente vai ocorrer onde estiver esse acento sendo todo o restante do texto decifrado de maneira correta, caso o texto não possua acentuação vai ser decifrado sem nenhum problema. Outra limitação é que não conseguimos implementar o AES em modo CTR. Ademais, algumas vezes pode demorar um pouco para gerar chaves entre outras funções feitas no código, devido ao algoritmo ser um pouco lento.

4. Comandos para Utilizar as Aplicações Solicitadas no Trabalho

A princípio esse arquivo que a gente está passando abaixo é um que criamos e deixamos dentro da pasta do trabalho e ele se chama “arquivo”, caso o nome fosse outro deveria utilizar o correspondente. Lembrando que esses comandos funcionam somente se estiver a pasta do trabalho aberta no terminal do Windows/Linux ou no VScode, por fim, a versão do python tem que está na **“3.10.2”**.

1- Para gerar o par de primos basta inserir o seguinte comando no terminal com a pasta do arquivo do trabalho aberta:

```
python3 .\Gerador-Verificador-de-Assinaturas-RSA-Arquivos.py
-geracaoparprimo
```

2- Para encriptar basta inserir o seguinte comando:

```
python3 .\Gerador-Verificador-de-Assinaturas-RSA-Arquivos.py -encript
-arquivo arquivo.txt -descricaochave chave_publica.public
```

3- Para decriptar o arquivo com a mensagem:

```
python3 .\Gerador-Verificador-de-Assinaturas-RSA-Arquivos.py -decript
-arquivo arquivo.txt.encript -descricaochave chave_privada.priv
```

4- Para assinar o arquivo:

```
python3 .\Gerador-Verificador-de-Assinaturas-RSA-Arquivos.py
-assinaturadoarquivo  chave_publica.public chave_privada.priv
arquivo.txt.encript
```

5- Para validar o arquivo:

```
python3 .\Gerador-Verificador-de-Assinaturas-RSA-Arquivos.py -valida
chave_publica.public chave_privada.priv arquivo.txt.encript
```