Algoritmo RSA

O algoritmo RSA envolve a criação de dois conjuntos de chaves: um público e outro privado. A ideia é usar o par de chaves pública para cifrar uma mensagem, mas somente a chave privada correspondente pode decifrá-la. É crucial destacar que a chave privada não pode ser deduzida a partir da chave pública. Dessa forma, isso garante que somente alguém autorizado a enviar uma mensagem cifrada possua o primeiro conjunto de chaves, e somente o destinatário com o segundo conjunto de chaves possa descriptografá-la.

1. Geração das Chaves

Primeiramente são gerados dois primos aleatórios de 1024 bits. A chave é gerada utilizando uma biblioteca que cria um número grande até que um número primo seja identificado, o qual então é designado como o valor final. Para determinar se o número inicial é primo, é utilizado o método de Miller-Rabin. Se o valor não passar nesse teste, é considerado definitivamente composto; caso contrário, é considerado primo.

O próximo passo é gerar o resultado da multiplicação dos dois primos encontrados: n = pq. Em seguida gerar a função totiente de Euler: φ: φ(n) = (p − 1)(q − 1). Em que o φ(n) é usado para gerar o ” e”. A geração da chave pública é feita pelo “n” e pelo “e”. No intervalo [3, φ(n)) seja escolhido o primo “e” que o mdc = 1 com φ(n). A geração da chave privada é feita pelo “n” e pelo “d”. Em que “d” é a multiplicação inversa da chave pública: e . d = 1.mod(φ(n)).

2. Cifração/Decifração

Considerando a mensagem “m” a cifra é feita a partir de F(m, e) = m^e. mod(n) = c e a decifração “c” é gerada a partir de F(c, d) = c^d. mod(n) = m.

3. OAEP (Optimal Asymmetric Encryption padding)

A segurança do algoritmo RSA-OAEP está vinculada à robustez dos elementos fundamentais usados na codificação e decodificação do RSA, assim como à segurança do procedimento criptográfico chamado de OAEP (Optimal Asymmetric

Encryption Padding). Por meio desse método, um adversário não consegue gerar uma mensagem cifrada válida sem ter acesso à chave privada correspondente. Além disso, a segurança do RSA-OAEP não pode ser demonstrada utilizando o modelo convencional de problemas do RSA. O OAEP cumpre dois objetivos primordiais: introduz um fator aleatório, transformando o processo de cifragem em um esquema probabilístico, e previne a decodificação parcial das mensagens cifradas.

4. Execução do Programa

O documento teste.txt contém a mensagem a ser codificada e pode ser modificado para realizar testes. Por padrão, a mensagem "Hello World" foi incluída. Depois de realizar as alterações desejadas no arquivo de teste, execute o seguinte comando no terminal Python: **python main.py**.