XXXI Congresso de Iniciação Científica Unicamp 2023

Estudo sobre a implementação híbrida de algoritmos criptográficos pré e pós-quânticos em dispositivos restritos

Palavras-Chave: criptografia pos-quântica; segurança; CRYSTALS-Kyber

Autores:
Prof. Dr. Marco Aurélio Amaral Henriques - Unicamp
João Paulo Pereira de V. Mendes – Unicamp

1. Introdução

A segurança antes atribuída aos criptossistemas convencionais está em sério risco devido aos avanços da computação quântica. Com o desenvolvimento do algoritmo de Shor [Shor 1997], os sistemas criptográficos amplamente utilizados, como RSA e ECDSA, enfrentam a possibilidade real de serem quebrados em tempo polinomial por computadores quânticos. Contudo, pesquisadores e especialistas têm tomado contramedidas para enfrentar os desafios colocados pela quebra potencial dos criptossistemas clássicos. Surgiu, então, a área da criptografia pós-quântica, dedicada ao desenvolvimento de esquemas criptográficos resistentes a ataques quânticos.

No ano de 2022, o National Institute of Standards and Technology (NIST) adicionou oficialmente alguns algoritmos ao seu padrão criptográfico pós-quântico, adotando CRYSTALS-Kyber como algoritmo de encriptação assimétrica e os algoritmos FALCON, CRYSTALS-Dilithium e SPHINCS+ para fins de assinatura digital e com a crescente urgência e rápido desenvolvimento tecnológico da computação quântica, é provável que veremos a implementação desses algoritmos PQC em muitos dispositivos em breve [NIST 2022].

Entretanto, é importante destacar que ser novo não é sinônimo de ser imediatamente seguro. Na criptografia, a confiabilidade de algo antigo e exaustivamente testado é muito superior a algo novo sem falhas aparentes. Um exemplo disso é a quebra de dois candidatos que chegaram à Rodada 3 da competição do NIST, Rainbow e SIKE [1]. Porém, estas ocorrências não significam que a criptografia pós-quântica esteja fadada ao fracasso ou que não seja confiável, mas sim que devemos exercer cautela e não abandonar prontamente nossos métodos tradicionais. Portanto, a recomendação atual no cenário, como a sugerida pela Agência Nacional de Segurança Cibernética da França (ANSSI), é a adoção de um modelo híbrido que combina criptografia pós-quântica com técnicas clássicas. Essa abordagem proporciona uma camada adicional de segurança e permite uma transição mais segura para um cenário pós-quântico.

A abordagem proposta também abrange o preparo contra a variante de ataques conhecida como "armazenar agora, descriptografar depois", que consiste em armazenar os dados e mensagens trocados atualmente para descriptografá-los assim que os computadores quânticos estiverem disponíveis. A existência deste tipo de ataque implica que a segurança da maior parte da infraestrutura digital baseada em criptografia de chave pública (PKC) pode já estar vulnerável a um eventual ataque quântico. O que traz a tona a urgência da adoção do modelo híbrido.

Deste modo, o projeto tem compromisso a investigação da implementação híbrida de algoritmos pré e pós-quânticos, uma técnica promissora destinada a garantir a segurança digital durante a transição da computação clássica para a quântica, focando em dispositivos com recursos restritos de memória e velocidade de processamento. A estratégia busca unir a segurança dos algoritmos criptográficos pré e pós-

quânticos, com a premissa de preservar a integridade do algoritmo final desde que pelo menos um dos esquemas permaneça seguro.

2. Criptografia baseada em reticulados

2.1. Mecanismo de Encapsulamento de Chave (KEMs)

Os algoritmos pós-quânticos podem ser divididos em duas classes principais: KEM (Key Encapsulation Mechanism) e algoritmos de assinatura. A decisão de priorizar os algoritmos que executam a troca de chaves (KEMs) neste trabalho se deve à urgência e à importância da confidencialidade das informações especialmente para preservar o sigilo de informações confidenciais, resguardando dados sensíveis em ambientes web e outras aplicações críticas.

O propósito de um KEM é estabelecer de forma segura uma chave secreta compartilhada entre duas partes. Ele é constituido por um conjunto de algoritmos (**Gen**, **Encap**, **Decap**) de forma que:

1. O algoritmo de geração de chaves, **Gen**, é uma função de geração de par de chaves que retorna um par de chaves contendo uma chave pública *pk* e uma chave *sk*.

2. Uma função de encapsulamento de chave, **Encap**, chamada pelo remetente, que recebe a chave pública do destinatário e uma opção de criptografia; ela retorna uma chave secreta *K* e uma mensagem de encapsulamento de chave de texto cifrado c. O remetente envia a mensagem de encapsulamento de chave para o destinatário.

3. Uma função de desencapsulamento de chave, **Decap**, chamada pelo destinatário, que recebe a chave privada do destinatário e a mensagem de encapsulamento de chave recebida; ela retorna a chave secreta *K*.

2.2. CRYSTALS-Kyber

Entre os algoritmos KEM, o NIST aprovou exclusivamente o CRYSTALS-Kyber, tornando-o opção única de escolha para o nosso trabalho.

O Kyber adquire sua segurança através da abordagem do problema matemático chamado Module Learning With Errors (MLWE), o qual é definido em um reticulado - uma estrutura algébrica abstrata. Um reticulado representa um espaço vetorial discreto que contém um conjunto de pontos distribuídos em um espaço n-dimensional. Cada ponto do conjunto S pode ser expresso em termos de uma base composta por n vetores linearmente independentes em S. É importante destacar que existem várias bases possíveis para um reticulado, e algumas delas podem ser mais eficazes do que outras em termos de expressar os pontos em S através de combinações lineares dos vetores da base.

2.2.1. MODULE LEARNING WITH ERRORS (MLWE)

O problema LWE (Learning With Errors) é composto pela solução de sistemas matriciais que utilizam a combinação linear de vetores pertencentes a um reticulado S específico. No entanto, para garantir a segurança dos dados, esse processo é dificultado pela introdução de um ruído aleatório nas equações, tornando a resolução do sistema não trivial.

Dessa forma, considerando $\mathbb{Z}_q$ como um anel de inteiros módulo q e $\mathbb{Z}_q^n$ como um conjunto de n-vetores sobre $\mathbb{Z}_q$, o problema consiste em determinar a dificuldade de distinguir um vetor qualquer $\boldsymbol{t} \in \mathbb{Z}_q^n$, tal que $\boldsymbol{t} = \mathbf{A} \cdot \boldsymbol{s_1}$ e $\boldsymbol{t'} = \mathbf{A} \cdot \boldsymbol{s_1} + \boldsymbol{s_2}$, onde $\boldsymbol{s_1}$ e $\boldsymbol{s_2}$ pertencem a $\mathbb{Z}_q^n$ e A é uma matriz cujas entradas

também estão definidas em $\mathbb{Z}_q$. Neste caso, $\boldsymbol{s_2}$ se comporta como gerador de ruido, $\boldsymbol{s_1}$ a como a chave e $\boldsymbol{t}$ como chave pública.

O problema MLWE (Module Learning With Errors) é uma extensão do LWE (Learning With Errors), no qual é usado um polinomial $\mathbb{Z}_q[X]/(X^n + 1)$ ao invés de inteiros módulo *q*, como no LWE. Os valores para a construção deste anel polinomial são estão fixados para o Kyber em *q* = 3329 e *n* = 256. O que nos proporciona vetores cujos elementos são polinômios indo do grau 0 até o grau 255 e custosas operações de multiplicação polinomial. Sua matriz **A** também é fonte de um alto custo, principalmente de memória, uma vez que é constituida por polinômios.

2.2.2. Number Theoretic Transform (NTT) e outros métodos de multiplicação polimonial

A transformada numérica teórica (NTT - Number Theoretic Transform) é uma técnica fundamental amplamente utilizada em criptografia e processamento de sinais, especialmente em sistemas baseados em reticulados. A NTT é altamente eficiente para realizar operações matemáticas em anéis numéricos específicos, simplificando as operações entre vetores de polinômios. No entanto, sua aplicação requer um custo alto de uso de memória se não tomados os devidos cuidados.

Dentro do contexto da biblioteca *pqm4*, que tem como objetivo implementar várias inovações criptográficas para a família de microcontroladores ARM Cortex-M4, os proponentes sugeriram duas opções de otimização para reduzir o consumo de memória para o esquema de criptografia kyber [Kannwischer 2019]

A primeira opção é utilizar a NTT inteiramente in-place, ou seja, realizar a transformação diretamente no mesmo espaço de memória dos dados de entrada, sem a necessidade de alocar memória adicional para o resultado. Essa abordagem pode economizar espaço de memória, embora exija cuidados adicionais para garantir que os dados sejam manipulados corretamente.

A segunda opção é empregar técnicas alternativas de multiplicação, como o método de Karatsuba e o método de Toom-Cook que são usados no esquema do algoritmo criptográfico pós-quântico SABER. Essas técnicas permitem realizar a multiplicação de polinômios de maneira mais eficiente, o que pode reduzir a necessidade de transformações NTT e NTT inversas, também contribuindo para economia de memória, mesmo que a NTT seja mais rápida.

Após análise, a equipe de desenvolvimento da biblioteca *pqm4* optou por fazer otimizações na NTT. A escolha do uso do método Toom-Cook foi adotada neste trabalho a fim de eliminar e alcançar uma melhor eficiência de memória.

Karkamar et al no artigo *Time-memory trade-off in Toom-Cook multiplication: an application to module-latticebased cryptography* [Karmakar 2020], sugeriu uma nova aplicação do Toom-Cook, tornando-o in-place.

3. Otimizações

Dadas estas considerações sobre os algoritmos pós-quânticos, pode-se aferir que eles acabam sendo o problema quando se trata de portar um algoritmo híbrido. Uma vez que os esquemas de criptografia assimétrica clássicos não lidam com problemas matemáticos complexos.

As otimizações foram feitas em cima do código original do Kyber, disponibilizado no final da terceira rodada do NIST.

3.1. Geração de Matriz A in-place

Output: Secret key $sk \in \mathcal{B}^{12 \cdot k \cdot n/8}$
Output: Public key $pk \in \mathcal{B}^{12 \cdot k \cdot n/8+32}$

1: $d \leftarrow \mathcal{B}^{32}$
2: $(\rho, \sigma) := \mathrm{G}(d)$
3: $N := 0$
4: **for** i from 0 to $k-1$ **do** ▷ Generate matrix $\hat{\mathbf{A}} \in R_q^{k \times k}$ in NTT domain
5: **for** j from 0 to $k-1$ **do**
6: $\hat{\mathbf{A}}[i][j] := \mathsf{Parse}(\mathsf{XOF}(\rho, j, i))$
7: **end for**
8: **end for**
9: **for** i from 0 to $k-1$ **do** ▷ Sample $\mathbf{s} \in R_q^k$ from B_{η_1}
10: $\mathbf{s}[i] := \mathsf{CBD}_{\eta_1}(\mathsf{PRF}(\sigma, N))$
11: $N := N + 1$
12: **end for**
13: **for** i from 0 to $k-1$ **do** ▷ Sample $\mathbf{e} \in R_q^k$ from B_{η_1}
14: $\mathbf{e}[i] := \mathsf{CBD}_{\eta_1}(\mathsf{PRF}(\sigma, N))$
15: $N := N + 1$
16: **end for**
17: $\hat{\mathbf{s}} := \mathsf{NTT}(\mathbf{s})$
18: $\hat{\mathbf{e}} := \mathsf{NTT}(\mathbf{e})$
19: $\hat{\mathbf{t}} := \hat{\mathbf{A}} \circ \hat{\mathbf{s}} + \hat{\mathbf{e}}$
20: $pk := (\mathsf{Encode}_{12}(\hat{\mathbf{t}} \bmod^+ q) \| \rho)$ ▷ $pk := \mathbf{As} + \mathbf{e}$
21: $sk := \mathsf{Encode}_{12}(\hat{\mathbf{s}} \bmod^+ q)$ ▷ $sk := \mathbf{s}$
22: **return** (pk, sk)

Figura 1. Pseudocodigo para função de geração de chaves do Kyber.

A geração da matriz, processo descrito da linha 4 até a linha 6 foi modificado, tal qual a geração de ***s*** e ***e***. Agora, sem a NTT, assumiremos que $\hat{\mathbf{A}}$ passa a ser apenas **A** e sendo **k** a ordem da matriz quadrada **A** , que pode valer 2, 3 ou 4 dependendo do nível de segurança trabalhado.

Depois da geração de ***s*** e ***e***, pulamos as linhas **17** e **18** afinal não estamos fazendo o uso da NTT e aplicamos o método Toom-Cook in-place (pol_mul) e depois uma adição polinomial, tal que:

```
1.    for i from 0 to k-1 do
2.        for i from 0 to k-1 do
3.'           A[i][j] := Parse(XOF(σ,j,i)
4.            t[i] = pol_mul(A[i][j], s[j])
5.            t[i] = pol_add(t[i])
6.        end for
7.    end for
```

Figura 2. Pseudocodigo para função de geração de chaves do Kyber modificado.

3.2. Aplicação do Toom-Cook in-place

Toom-Cook é um algoritmo de multiplicação que se baseia na decomposição dos números a serem multiplicados em várias partes menores e, em seguida, utiliza essas partes para calcular o produto final. Esse algoritmo é uma alternativa mais rápida para a multiplicação tradicional, especialmente quando se lida com números grandes. O Toom-Cook é baseado no princípio da divisão e conquista e sua versão in-place opera em uma estrutura de dados existente sem requerer espaço adicional para armazenar os resultados intermediários. Isso significa que os cálculos são feitos diretamente nos dados de entrada.

4. Metodologia

Realizamos a medição de ciclos do uso de RAM para a função Keygen, que trabalha com a geração da matriz **A** e uso da função **NTT**. Comparamos a versão *ref* do Kyber disponibilizada pela equipe pq-crystals. Foi utilizado o software Massif, disponível no programa de avaliação Valgrind 3.15.0, como

ferramenta para medição da memória RAM e o nível de segurança do kyber escolhido foi o nível 1 (kyber-512), onde k = 2.

5. Resultados e Discussões

Função KeyGen	Pico de RAM
Kyber512 *ref*	**13.3548 B**
Kyber512 mod	**9.113 B (-32%)**

Figura 3. Pseudocódigo para função de geração de chaves do Kyber modificado.

O resultado apresentado demonstra uma diminuição considerável de pico de RAM dada as modificações realizadas na função de KeyGen, porém o uso do Toom-Cook in-place é uma alternativa mais lenta que a aplicação da NTT. Porém a implementação em C puro usando o Toom-Cook in-place é uma alternativa promissora quando não se tem features para aceleração de Hardware disponíveis.

A otimização do uso de memória do Kyber torna possível sua união com algorítimos clássicos, uma vez que o principal gargalo para sua aplicação é justamente o consumo de memória do algorítimo pós-quântico.

6. Próximos passos

Os próximos passos englobam a tarefa de analisar alternativas para hashes criptográficos mais eficientes, explorar a velocidade de utilização do Toom-Cook em conjunto com o Kyber, e meticulosamente avaliar os ganhos de eficiência no uso da memória para as demais versões do Kyber.

7. Próximos passos

NIST (2022). Nist announces first four quantum-resistant cryptographic algorithms. https: //www.nist.gov/news-events/news. Acessado em 26/06/2022.

Kyber team. (2021). Kyber: A CRYPTOGRAPHIC LIBRARY FOR THE POST-QUANTUM WORLD (Round 3 Specification - January 31, 2021). Obtido de https://pq-crystals.org/kyber/data/kyber-specification-round3-20210131.pdf. Acessado em 26/06/2022.

Bermudo Mera, J. M. ., Karmakar, A., & Verbauwhede, I. (2020). Time-memory trade-off in Toom-Cook multiplication: an application to module-lattice based cryptography. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, *2020*(2), 222–244. https://doi.org/10.13154/tches.v2020.i2.222-244

Botros, L., Kannwischer, M. J., Schwabe, P. (2019). Memory-Efficient High-Speed Implementation of Kyber on Cortex-M4. Apresentado na Africacrypt 2019, Rabat, Marrocos, em 10 de julho de 2019. Disponível em: https://pdfs.semanticscholar.org/4161/587d381d0dbec7a897d01abff09f8c6f3b1a.pdf