Explorando Criptografia de Reticulados

Guilherme Cappelli

Grupo de Resposta a Incidentes de Segurança - UFRJ

GRIS Week
22/08/2024

Introdução

- NIST Post-Quantum Cryptography Standardization
- Algoritmo de Shor
- Trabalho Pioneiro de Ajtai
- Resistência à Ataques na Computação Quântica

Reticulados

- **Definição:**
  - Um reticulado $\mathcal{L}$ gerado por uma base $B \in \mathbb{Z}^{m \times n}$ é o conjunto de pontos num espaço vetorial que podem ser alcançados por combinações lineares inteiras dos vetores que compõe a base B, i.e.,

$$\mathcal{L}(\mathbf{B}) = \{\mathbf{Bx} : \mathbf{x} \in \mathbb{Z}^n\}$$

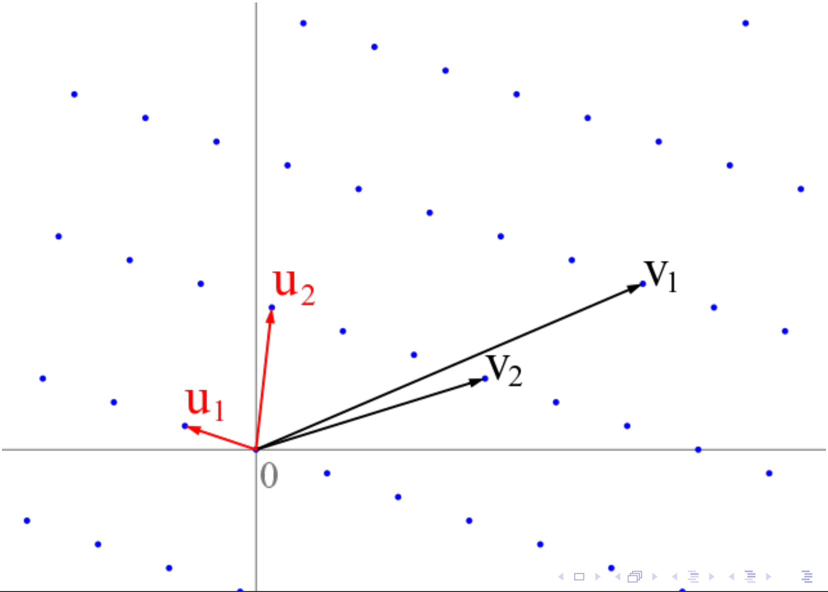
- **Observação:**
  - Note que, o espaço vetorial do reticulado $\mathcal{L}$ não é o mesmo que o **span** da base B, porque nesta nova estrutura estamos limitados ao conjunto dos inteiros $\mathbb{Z}$, enquanto o espaço vetorial gerado pela base B aceita qualquer número real x como coeficiente da combinação linear.

$$\textbf{span}(\mathbf{B}) = \{\mathbf{Bx} : \mathbf{x} \in \mathbb{R}^n\}$$

Ex: $\begin{bmatrix} 1 & 2 \\ 3 & 5 \end{bmatrix} \cdot \begin{bmatrix} 0.2 \\ 3.7 \end{bmatrix} \notin \mathcal{L}$

Ex2: $\begin{bmatrix} 1 & 2 \\ 3 & 5 \end{bmatrix} \cdot \begin{bmatrix} 5 \\ 9 \end{bmatrix} \in \mathcal{L}$

Reticulados



Problemas Difíceis em Reticulados

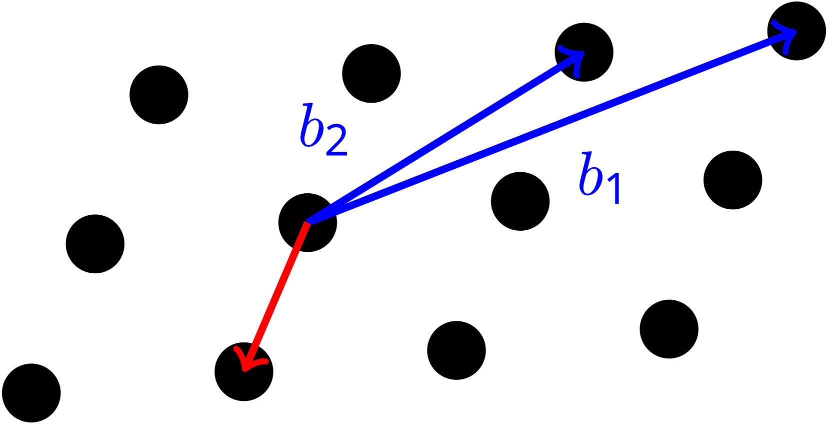
- **Shortest Vector Problem (SVP)**
  - O problema do vetor mais curto é resolvido ao encontrar um vetor não nulo $w \in \mathcal{L}$ que minimize o tamanho $\|w\|$.
  - Esse problema tem uma versão aproximada chamada de apprSVP, onde esse tamanho é limitado por um fator de $\gamma \cdot \lambda(\mathcal{L})$, onde $\gamma \geq 1$ e

$$\lambda(\mathcal{L}) = \mathbf{inf}\{\|w\| : w \in \mathcal{L}\backslash\{0\}\}$$

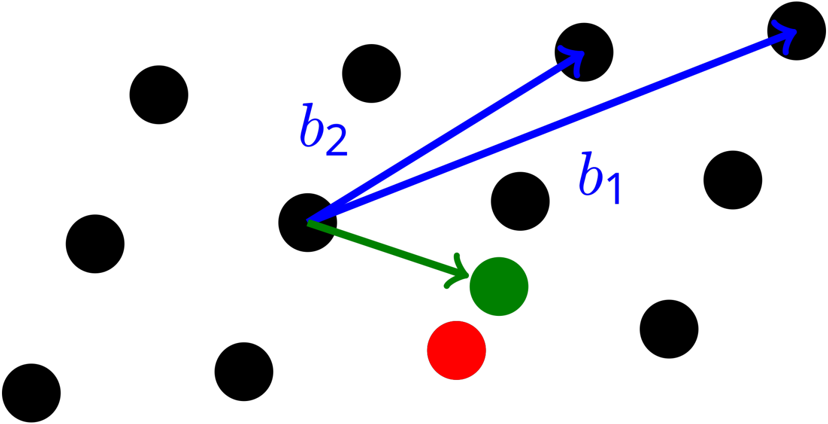
- **Closest Vector Problem (CVP)**
  - Dado um reticulado $\mathcal{L}$ gerado pela base $\mathbf{B} \in \mathbb{Z}^{m \times m}$ e um vetor $w \in \mathbb{R}^m$, geralmente $w \notin \mathcal{L}$, encontre o vetor $v \in \mathcal{L}$ mais próximo de w que minimize o tamanho de $\|v - w\|$.
  - Como no SVP, temos também a versão aproximada do problema, chamada de apprCVP, onde a distância entre esses vetores é limitada pelo fator $\gamma \cdot \text{dist}(w, \mathcal{L})$, onde $\gamma \geq 1$ e

$$\text{dist}(w, \mathcal{L}) = \mathbf{inf}\{\|v - w\| : v \in \mathcal{L}\}$$

Problemas Difíceis em Reticulados (SVP)



Problemas Difíceis em Reticulados (CVP)



Razão de Hadamard

- **Definição:**
  - A razão de Hadamard avalia a ortogonalidade de uma base $\mathbf{B}$, tal que, seu valor numérico cresce conforme os vetores são mais ortogonais entre si.

$$\mathcal{H}(\mathbf{B}) = \left(\frac{\det(\mathcal{L})}{\|b_1\| \cdot \|b_2\| \dots \|b_n\|} \right)^{1/n}, \mathcal{H}(\mathbf{B}) \in (0, 1]$$

- **Aplicações:**
  - No Closest Vector Problem visto no frame anterior, é interessante possuir uma base quase ortogonal.
  - Segue que, uma boa base possui uma razão de Hadamard próximo de 1, enquanto uma base ruim tem uma razão baixíssimo.

Matriz Unimodular

- **Definição:**
  - Uma matriz $U \in \mathbb{Z}^{n \times n}$ não singular é unimodular se:

$$1. \quad \det(U) = \pm 1$$

$$2. \quad U^{-1} \text{ também é unimodular.}$$

- **Teorema:**
  - Sejam duas bases distintas B e C. Os reticulados $\mathcal{L}(B)$ e $\mathcal{L}(C)$ são iguais sse existe U unimodular tal que $C = BU$.
  - Em outras palavras, eu posso encontrar uma nova base para um mesmo reticulado $\mathcal{L}$ a partir de uma matriz unimodular.

Goldreich-Goldwasser-Halevi (GGH)

- **Chave Privada**
  - O CVP é resolvido sem grandes dificuldades quando temos boas bases, isto é, bases com razão de Hadamard próximas de 1. Então, é justo que esta vantagem fique com os portadores da chave privada.
  - Para obter uma matriz com essa qualidade, primeiro geramos a matriz $R \in \mathbb{Z}^{n \times n}$ uniformemente distribuída de parâmetro l, geralmente 4, onde n é o tamanho da mensagem.
  - Esta é então somada a matriz $k \cdot \mathbf{I}$, tal que $k = \lfloor \sqrt{n} \cdot l \rceil$ e $\mathbf{I}$ é a matriz identidade de dimensão n. Enquanto a razão de Hadamard não for suficientemente próximo de 1, repita essas etapas.

$$\mathsf{V} = R + k \cdot \mathbf{I}, \text{ onde } R = \{-l, \ldots, +l\}^{n \times n}$$

Goldreich-Goldwasser-Halevi (GGH)

- **Chave Pública**
  - Já a chave pública deve ser de 'má' qualidade para que qualquer um que queira solucionar o CVP tenha muita dificuldade. Contudo, a chave pública **W** precisa gerar o mesmo reticulado $\mathcal{L}$ que a base **V** para conseguirmos decriptar a mensagem.
  - Uma forma de garantir isso é aplicar o Teorema 1, que diz que dois reticulados $\mathcal{L}(\mathbf{B})$ e $\mathcal{L}(\mathbf{C})$ são iguais se e somente se $\exists U$ unimodular tal que $C = BU$.
  - Então, precisamos gerar uma matriz unimodular para transformar a chave privada na pública.

$$\mathbf{W} = \mathbf{UV}$$

Goldreich-Goldwasser-Halevi (GGH)

- **Vetor de Erros**
  - Para desviar o 'ciphertext' do reticulado (e dificultar a vida de quem queira quebrar o GGH), é gerado um vetor efêmero de erros público $r \in \{\pm\sigma\}^n$, onde $\sigma \in \mathbb{N}$ é o parâmetro threshold.
  - Este valor é calculado através da norma l_1 das linhas de V^{-1}. Denote $p \in \mathbb{R}^+$ como a maior norma dessas linhas, então, enquanto $\sigma < \frac{1}{2p}$, não há como ocorrer erros na decriptação da mensagem.
  - Fica na responsabilidade de quem encripta a mensagem gerar esse vetor aleatório com threshold σ. Esta etapa é feita escolhendo as entradas do vetor $r \in \mathbb{Z}^n$ entre $+\sigma$ e $-\sigma$ com probabilidade $1/2$.

Ex: $r = \begin{bmatrix}-1, 1, 1, -1, -1, 1\end{bmatrix}$, onde $\sigma = 1$

Goldreich-Goldwasser-Halevi (GGH)

- **Encriptar**
  - O processo de encriptar uma mensagem m exige que esta seja codificada primeiro em um vetor.
  - É importante que os elementos do vetor mensagem estejam no conjunto $\{-n, -(n-1), \ldots, -1, 0, 1, \ldots, n-1, n\}$, onde n é o tamanho do vetor.

$$c = m \cdot W + r$$

  - Percebe-se então, que a dimensão das chaves pública e privada dependem exclusivamente do tamanho da mensagem.

Goldreich-Goldwasser-Halevi (GGH)

- **Decriptar**
  - O processo de decriptar um 'ciphertext' é feito usando o algoritmo de Babai 'NearestPlane' com a chave privada V.
  - Assim, a mensagem original é obtida computando

$$m = \lfloor c \cdot V^{-1} \rceil \cdot V \cdot W^{-1}$$

  - Porque $W = UV$ e $c = mW + r$

$$m = \lfloor (mW + r) \cdot V^{-1} \rceil \cdot V \cdot (UV)^{-1}$$

$$m = \lfloor mUV \cdot V^{-1} + r \cdot V^{-1} \rceil \cdot U^{-1}$$

$$m = \lfloor mU + r \cdot V^{-1} \rceil \cdot U^{-1}$$

$$m = mU \cdot U^{-1}$$

Goldreich-Goldwasser-Halevi (GGH)

- Com esse novo vetor, basta decodificá-lo da mesma forma em que a mensagem foi transformada em um vetor.
- Observe, que se a matriz unimodular U usada para gerar a chave pública for armazenada, podemos simplificar este cálculo

$$m = \lfloor c \cdot V^{-1} \rceil \cdot V \cdot V^{-1} U^{-1}$$

- Então,

$$m = \lfloor c \cdot V^{-1} \rceil \cdot U^{-1}$$

Goldreich-Goldwasser-Halevi (GGH)

- **Exemplo Prático**
  - Sejam V a chave privada e W a chave pública do sistema de criptografia GGH, onde $\mathcal{H}(V) = 0.97$ e $\mathcal{H}(W) = 0.008$.

$$\mathbf{V} = \begin{bmatrix} 11 & -4 & 1 \\ 3 & 7 & -4 \\ -3 & 4 & 8 \end{bmatrix} \qquad \mathbf{W} = \begin{bmatrix} 878 & -1004 & -667 \\ 926 & 1579 & -1301 \\ 207 & -404 & -43 \end{bmatrix}$$

  - Neste caso, o threshold público σ limita os elementos do vetor efêmero r em 2.
  - Para encriptar a mensagem $m =$ "ABC"

$$c = \left[1, 2, 3\right] \cdot \begin{bmatrix} 878 & -1004 & -667 \\ 926 & 1579 & -1301 \\ 207 & -404 & -43 \end{bmatrix} + \left[2, -2, 2\right]$$

$$c = \left[-1101, 940, 1058\right]$$

Goldreich-Goldwasser-Halevi (GGH)

- **Exemplo Prático**
  - Para decriptar o 'ciphertext' c é computado

$$m = \left\lfloor c \cdot \begin{bmatrix} \frac{8}{97} & \frac{4}{97} & \frac{-4}{291} \\ \frac{-4}{291} & \frac{91}{873} & \frac{47}{873} \\ \frac{11}{291} & \frac{-32}{873} & \frac{89}{873} \end{bmatrix} \right\rceil \cdot \begin{bmatrix} 3464 & 2560 & 1397 \\ 1239 & 909 & 496 \\ 1391 & 1028 & 561 \end{bmatrix}$$

$$m = [1, 2, 3]$$

  - Convertendo cada elemento de volta para a tabela combinada, obtemos a mensagem original.

$$m = \text{"ABC"}$$

Ataques ao GGH

- Computando a Chave Privada
- Resolvendo o CVP
- Ataque de Nguyen

Computando a Chave Privada

- Este é um ataque bem simples que se baseia no uso do algoritmo Lenstra-Lenstra-Lovász (LLL pros íntimos).
- Imagine que você tem uma caixa mágica que performa o algoritmo LLL. Você coloca uma matriz $B \in \mathbb{Z}^{n\times n}$ e depois de 10 segundos você tira uma matriz $A \in \mathbb{Z}^{n\times n}$ diferente, mas que gera o mesmo reticulado.
- Essa nova matriz A possui vetores mais ortogonais entre si, i.e., possui razão de Hadamard próxima de 1.

Computando a Chave Privada

- Para chaves privadas e públicas de dimensão < 400, aplicar o algoritmo LLL na chave pública se torna muito perigoso, porque LLL(W) retorna a matriz V.
- Assim, podemos calcular $U = W \cdot V^{-1}$ e em seguida obter m.

$$1. \quad V = \mathsf{LLL}(W)$$

$$2. \quad U = W \cdot V^{-1}$$

$$3. \quad m = \lfloor c \cdot V^{-1} \rceil \cdot U^{-1}$$

Resolvendo o CVP

- Neste segundo ataque usamos a técnica de incorporação, onde anexamos o vetor c encriptado na base do reticulado, gerando $\mathcal{L}'$.

$$W' = \begin{bmatrix} | & | & & | & | \\ w_1 & w_2 & \dots & w_n & c \\ | & | & & | & | \\ 0 & 0 & & 0 & 1 \end{bmatrix}^T$$

- Se rodarmos LLL nessa nova matriz, a primeira linha vai ser o vetor de erros $e = (c - v, 1) \in \mathbb{Z}^{n+1}$.
- Portanto, o ciphertext é decriptado tirando a inversa da chave pública W.

$$m = (c - e) \cdot W^{-1}$$

Ataque de Nguyen

- Suponha que a mensagem $m \in \mathbb{Z}^n$ foi encriptada pelo GGH com chave pública $W \in \mathbb{Z}^{n \times n}$ e o vetor de erro $r \in \{-\sigma, +\sigma\}$ em $c = mW + r$.
- A ideia principal desse ataque é reduzir c em um módulo muito bem escolhido para eliminar r da equação.
- OBS: Para o ataque funcionar, a base W deve possuir inversa no módulo 2σ

Ataque de Nguyen

- Dado que o threshold σ é uma informação pública, a estratégia é gerar um vetor $s = \{+\sigma\}^n$ e somá-lo aos erros r, porque $r + s = \{0, 2\sigma\}^n$.

$$c + s = mW + r + s$$

- E reduzirmos ambos os lados módulo 2σ, a soma $r + s$ será $\{0\}^n \in \mathbb{Z}_{2\sigma}^n$ e portanto, r não faz mais parte da equação.

$$c + s \equiv mW \mod 2\sigma$$

- Segue que,

$$m \equiv (c + s) \cdot W^{-1} \mod 2\sigma$$

Ataque de Nguyen

- Perceba que este vetor m não é a mensagem original, mas informações parciais dela.
- Esse vetor é chamado de $m_{2\sigma}$, que satisfaz $m \equiv m_{2\sigma} \mod 2\sigma$, e por definição, existe $m' \in \mathbb{Z}^n$ tal que,

$$m - m_{2\sigma} = 2\sigma m'$$

- Subtraímos o vetor $m_{2\sigma}W$ da equação original para extrair a informação que ainda não possuímos de m.

$$c - m_{2\sigma}W = (m - m_{2\sigma})W + r$$

- Assim,

$$\frac{c - m_{2\sigma}W}{2\sigma} = m'W + \varepsilon$$

Ataque de Nguyen

- Sejam o reticulado $\mathcal{L}$ gerado pela base $B \in \mathbb{Z}^{n \times n}$, c o vetor dado no problema de CVP e $v \in \mathcal{L}$ o vetor que minimiza a distância. Essa técnica incorpora o vetor c na base do reticulado gerando $\mathcal{L}'$.

$$B' = \begin{bmatrix} | & | & | & & | \\ c & b_1 & b_2 & \dots & b_n \\ | & | & | & & | \\ 1 & 0 & 0 & & 0 \end{bmatrix}$$

- Então, o algoritmo LLL de redução de base vai tentar resolver o SVP de B' e seu output vai ser $(c - v, 1) \in \mathbb{Z}^{n+1}$ que tem de ser curto e pertencente a $\mathcal{L}$.
- Somente assim, essa nova instância de CVP é solucionada pelo SVP do reticulado $\mathcal{L}'$.

Agradecimentos

Obrigado!