Cryptanalysis of LWE-based Cryptography

Students:

- Nguyễn Huy San - 23521335
- Bùi Hữu Tùng - 23521735

Class: NT219.P21.ANTN - Cryptography

Lecturer: Nguyễn Ngọc Tự

Overview

The Lattice and Learning With Errors (LWE) problem is a fundamental mathematical problem that underpins many modern cryptographic constructions, particularly in the field of Post-Quantum Cryptography (PQC). Its security relies on the hardness of certain lattice problems, making it a strong candidate for building cryptographic systems resilient to attacks from quantum computers. LWE-based cryptography offers versatile primitives, including encryption, key exchange, and digital signatures, and is a cornerstone of ongoing standardization efforts by organizations like NIST.

Mathematical Background

Lattices

A **lattice** $\mathcal{L}$ in $\mathbb{R}^n$ is a discrete additive subgroup of $\mathbb{R}^n$, which can be generated as all integer linear combinations of a set of linearly independent basis vectors $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_k \in \mathbb{R}^n$:

$$\mathcal{L} = \mathcal{L}(\mathbf{B}) = \left\{ \sum_{i=1}^{k} z_i \mathbf{b}_i \ : \ z_i \in \mathbb{Z} \right\}$$

where $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_k)$ is called a **basis** of the lattice, and $k \leq n$ is the **rank** of the lattice.

The Learning With Errors (LWE) Problem

The LWE problem, introduced by Oded Regev in 2005, is a generalization of the Learning Parity with Noise (LPN) problem. It involves recovering a secret vector s from a set of noisy linear equations.

Definition and Parameters ($n, q, \mathcal{X}$)

An LWE instance is defined by three main parameters:

- n: The dimension of the secret vector s (a positive integer).
- q: The modulus, typically a prime integer or a power of 2. All operations are performed modulo q.
- $\mathcal{X}$: The error distribution, usually a discrete Gaussian distribution or a uniform distribution over a small range, from which the noise e is sampled.

Given a secret vector $s \in \mathbb{Z}_q^n$, an LWE sample is a pair (a, b) where a is a randomly chosen vector from $\mathbb{Z}_q^n$ and b is computed as $b = \langle a, s \rangle + e \pmod{q}$, where e is a small error sampled from $\mathcal{X}$. The LWE problem are:

- Search-LWE: To recover s given $\mathbf{(a, c)} \in \mathbb{Z}_\mathbf{q}^\mathbf{n} \times \mathbb{Z}_\mathbf{q}$ sample according to $L_{s,\mathcal{X}}$
- Decision-LWE: To distinguish whether a pair $\mathbf{(a, c)} \in \mathbb{Z}_\mathbf{q}^\mathbf{n} \times \mathbb{Z}_\mathbf{q}$ are sampled according to $L_{s,\mathcal{X}}$ or the uniform distribution on $L_{s,\mathcal{X}}$

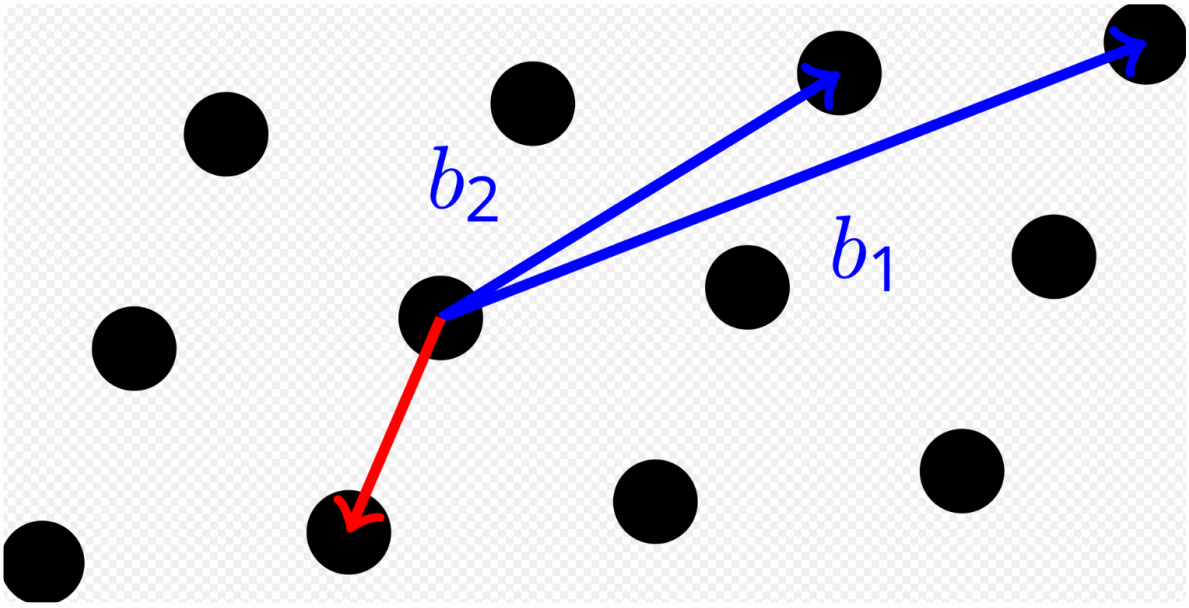
Secret Key and Error Distribution

The secret key s is typically chosen uniformly at random from $\mathbb{Z}_q^n$. The error e is crucial for the security of LWE, as it introduces noise that prevents direct linear algebra attacks. The choice of error distribution $\mathcal{X}$ (e.g., its standard deviation) significantly impacts the hardness of the problem.

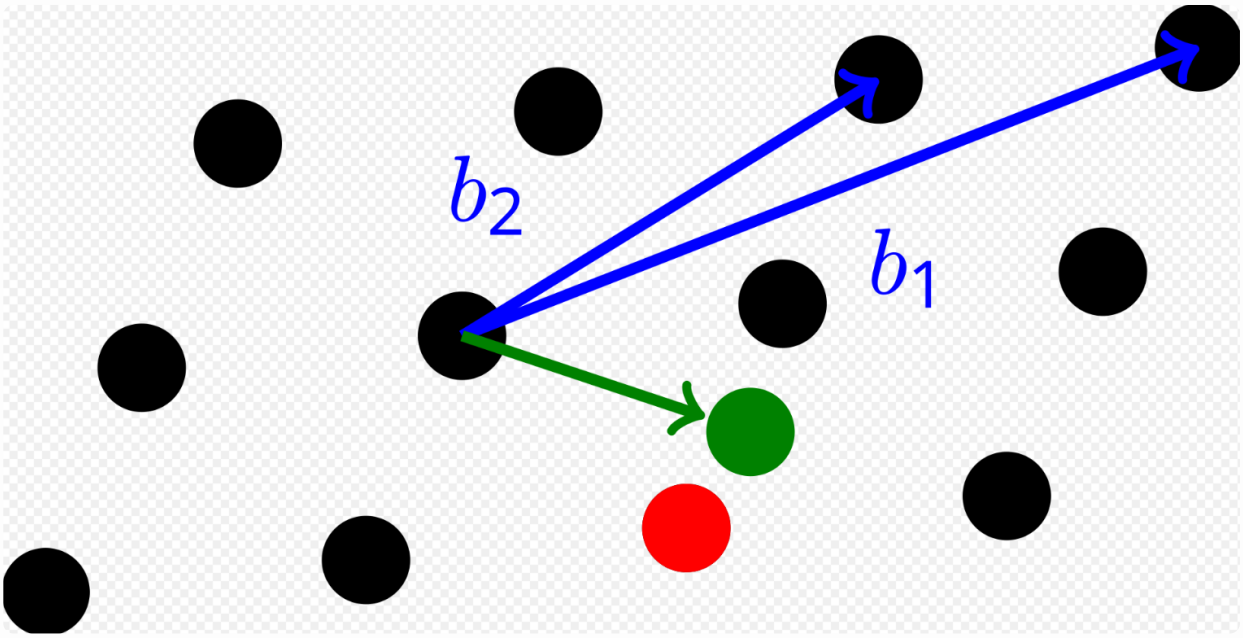
Hardness Assumptions

The security of LWE-based cryptography relies on the computational hardness of certain problems on lattices.

- **Shortest Vector Problem (SVP):** Given a basis for a lattice, SVP asks to find the shortest non-zero vector in the lattice.



- **Closest Vector Problem (CVP):** Given a basis for a lattice and a target vector, CVP asks to find the lattice vector closest to the target vector.



- **Shortest Independent Vectors Problem (SIVP):** Given a lattice L of dimension n, asks to find n linearly independent vectors that are as short as possible.

LWE in the context of cryptography

LWE has become a versatile basis for cryptographic constructions due to its hardness and connections to worst-case lattice problems.

LWE-based Encryption Schemes

LWE is a fundamental building block for public-key encryption schemes [7]. A simple scheme involves generating a public key from a matrix A and a vector $b = As + e$, where s is the secret key and e is the error. Encrypting a message involves adding it to a noisy LWE sample. Examples include Regev's original scheme and those based on Module-LWE like CRYSTALS-Kyber.

LWE-based Key Exchange

LWE is used in Key Encapsulation Mechanisms (KEMs) for key exchange. Schemes like FrodoKEM and CRYSTALS-Kyber (which is built upon Module-LWE) are prominent examples that have been part of NIST's post-quantum standardization process. The security of these protocols is proven based on the hardness of solving the LWE problem.

LWE-based Digital Signatures

LWE and its variants, such as Ring-LWE (RLWE), are also used to construct digital signature schemes. Dilithium, for instance, is a signature scheme based on Module-LWE [5].

LWE Attack and Threat Analysis

Table of LWE Cryptography Vulnerabilities

Name of Attack	Weakness Exploited	Exploit Mechanism	Countermeasure
Arora-Ge Attack [6, 9]	Specific algebraic structure, small dimension n.	Transforms LWE into multivariate polynomial equations, solved via linearization or Gröbner bases.	Use sufficiently large n and carefully chosen q.
BKW (Blum-Kalai-Wasserman) Algorithm [9, 10]	Combinatorial nature, effective with large noise and many samples.	Reduces LWE to a simpler problem by combining samples to eliminate secret components.	Limit number of available samples; careful parameter selection.
Lattice Reduction Attacks (LLL, BKZ) [3]	LWE can be formulated as finding a short vector in a lattice	Uses LLL or BKZ to find a short basis, from which the secret key or error can be extracted.	Choose LWE parameters $(n, q, \mathcal{X})$ to make SVP intractable; increase n and use appropriate modulus q.
Primal Attack [4]	Directly targets the secret key by solving a CVP/SVP instance.	Embeds LWE into a lattice problem (unique-SVP) and uses lattice reduction (e.g., BKZ) to find the shortest vector.	Robust parameter selection $(n, q, \mathcal{X})$ to ensure hardness of underlying lattice problems.

Detailed Attack Algorithms

Arora-Ge Attack

- **Attack Mechanism:** This attack transforms LWE samples $(a, b = a^T s + e)$ into polynomial equations

$$f_{\mathbf{a},b}(\mathbf{s}) = \prod_{x \in S}(b - \langle \mathbf{a}, \mathbf{s*} \rangle - x) \mod q$$

Where a and b are known and s is treated as the unknow variable. If (a, b) is a LWE sample, then $f_{\mathbf{a},b}(\mathbf{s}) = 0 \mod q$, else it isn't. Solving the system of polynomial equations

$$\{f_{\mathbf{a}_i,b_i}(\mathbf{s}) = 0 \mod q\}_{i=1}^{m}$$

of degree $|S|$ will give us the secret [13].

These systems can then be solved using algebraic techniques such as linearization or Gröbner bases. The original Arora-Ge algorithm has a complexity of $2^{O(n^2)}$ operations. As soon as $m = O(n^2)$, we can solve LWE with binary error in polynomial time [6].

- **Weakness Exploited:** The Arora–Ge attack exploits the algebraic structure of the LWE problem by transforming LWE samples into polynomial equations under the assumption that the error terms are small. Its effectiveness depends primarily on the dimension n, since the number of variables and the complexity of solving the resulting system grow rapidly with n.

- **Countermeasures:** To mitigate this attack, it is crucial to use a sufficiently large n and carefully chosen modulus q.

BKW (Blum-Kalai-Wasserman) Algorithm

- **Attack Mechanism:** Originally developed by Blum, Kalai, and Wasserman for the Learning Parity with Noise (LPN) problem, the BKW algorithm was later extended to LWE. The core idea is to combine multiple LWE samples to eliminate components of the secret, eventually reducing the problem to a simpler form that can be solved. Techniques like Lazy Modulus Switching, Coded BKW, and the Fast Walsh-Hadamard Transform are used to optimize its performance [8].
- **Weakness Exploited:** The BKW algorithm is a combinatorial attack that can be effective against LWE, especially for parameter choices with large noise. It assumes access to an unbounded number of LWE samples.
- **Countermeasures:** The primary defense against BKW-style attacks is to limit the number of available LWE samples. Additionally, careful parameter selection is necessary to ensure that the computational effort required by BKW remains infeasible.

Lattice Reduction Attacks (LLL, BKZ)

- **Attack Mechanism:** Algorithms like LLL (Lenstra-Lenstra-Lovász) and BKZ (Block Korkine-Zolotarev) are used to find a "good" (short) basis for a given lattice. LLL is a foundational algorithm, and BKZ is a more advanced variant that combines LLL reduction with enumeration, pruning, and block reduction steps to find shorter vectors.

These algorithms iteratively improve the lattice basis, and the primal attack often employs them to find the secret key.

- **Weakness Exploited:** LWE can be reformulated as finding a short vector in a high-dimensional lattice. The effectiveness of these attacks depends on the quality of the lattice basis.

- **Countermeasures:** To prevent successful lattice reduction attacks, LWE parameters ($n, q, \mathcal{X}$) must be chosen such that the Shortest Vector Problem (SVP) in the corresponding lattice is computationally intractable [3]. This typically involves increasing the dimension n and carefully selecting the modulus q. Experiments in [3] indicate that an LWE instance can be solved by LLL even for large key size n if the modulus q is chosen too large. A similar behavior was observed for the BKZ algorithm.

Primal Attack

- **Attack Mechanism:** This attack transforms the search-LWE instance into a unique-SVP problem using an embedding technique. It then employs lattice reduction algorithms, such as BKZ, to find the shortest vector in this lattice, which corresponds to the secret key or error, depend on the specific embedding type [4]. There are three type of embedding:

  - The Kannan's embedding [2] reduces the BDD problem to SVP. The corresponding embedding lattice is

$$\mathcal{L}_K = \left\{ \mathbf{y} \in \mathbb{Z}^{m+1} : \mathbf{y} = \mathbf{A*x} \mod q, \forall \mathbf{x} \in \mathbb{Z}^{n+1}, \mathbf{A*} = \begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{0} & \mu \end{bmatrix} \in \mathbb{Z}^{(m+1)\times(n+1)} \right\}$$

    and in practice it is preferable to use $\mu = 1$. $\mathbf{v} = (\mathbf{e} \mid \mathbf{1})$ is a short vector in the lattice.
  - The dual embedding proposed by Bai and Galbraith [1] constructs a lattice related to both secret s and error e. The corresponding embedding lattice is:

$$\mathcal{L}_D = \{\mathbf{x} \in \mathbb{Z}^{m+n+1} : (\mathbf{I}_m | \mathbf{B} | -\mathbf{b})\mathbf{x} = \mathbf{0} \mod q\},$$

    which has a basis

$$\mathbf{B} = \begin{bmatrix} q\mathbf{I}_m & -\mathbf{A} & \mathbf{b} \\ \mathbf{0} & \mathbf{I}_n & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & 1 \end{bmatrix}$$

The vector $\mathbf{v} = (\mathbf{e} \mid \mathbf{s} \mid \mathbf{1})$ is a short vector in the lattice.

- The Bai-Galbraith embedding improves dual embedding for such LWE instance that secret and error are chosen from different distributions, its core idea is to balance the size of the error and the secret. Specially, the short vector in lattice $\mathcal{L}_D$ can be re-balance as $\mathbf{v} = (\mathbf{e} \mid \omega\mathbf{s} \mid \omega)$ with scaling factor $\omega = \frac{\sigma_e}{\sigma_s}$ and the new embedding lattice is

$$\mathcal{L}_\omega = \{\mathbf{x} \in \mathbb{Z}^{m+n+1} : (\mathbf{I}_m | \frac{1}{\omega}\mathbf{B} | - \frac{1}{\omega}\mathbf{b})\mathbf{x} = \mathbf{0} \mod q\}$$

- **Weakness Exploited:** The primal attack directly targets the secret key by formulating the LWE problem as a Closest Vector Problem (CVP) or Shortest Vector Problem (SVP) instance in a specially constructed lattice.

- **Demonstration**:

```
fuyosuru@HuuTung:~/hoctap/MMH/doan$ sage primal_attack.py
Primal Attack on LWE (n=100, q=65537, m=200)
LWE instance generated with small secret and bounded noise.
Recovering error vector via lattice construction and LLL...
Recovering secret from corrected samples...
Recovered secret s': [65535, 65535, 65532, 65535, 2, 65532, 65534, 3, 4, 65534, 65532, 65534, 3, 65535, 1, 65535, 5, 65535, 2, 65535, 4, 65535,
65536, 1, 2, 65532, 65536, 0, 3, 4, 3, 3, 4, 65535, 3, 0, 4, 65534, 65535, 65533, 5, 65532, 0, 3, 5, 65534, 65533, 65535, 2, 1, 65534, 2, 1, 655
33, 4, 65532, 4, 3, 4, 5, 65535, 65536, 3, 1, 65534, 2, 3, 65534, 0, 65532, 65533, 65532, 5, 0, 5, 0, 2, 1, 0, 2, 65535, 4, 0, 65532, 4, 2, 6553
3, 65534, 65536, 65532, 0, 5, 65535, 1, 2, 65533, 65536, 65535, 65534, 1]
Original secret s : [-2, -2, -5, -2, 2, -5, -3, 3, 4, -3, -5, -3, 3, -2, 1, -2, 5, -2, 2, -2, 4, -2, -1, 1, 2, -5, -1, 0, 3, 4, 3, 3, 4, -2, 3,
0, 4, -3, -2, -4, 5, -5, 0, 3, 5, -3, -4, -2, 2, 1, -3, 2, 1, -4, 4, -5, 4, 3, 4, 5, -2, -1, 3, 1, -3, 2, 3, -3, 0, -5, -4, -5, 5, 0, 5, 0, 2, 1
, 0, 2, -2, 4, 0, -5, 4, 2, -4, -3, -1, -5, 0, 5, -2, 1, 2, -4, -1, -2, -3, 1]
Success: YES
Time taken: 677.39 seconds
```

- **Countermeasures:** The primary defense against primal attacks is to select LWE parameters $(n, q, \mathcal{X})$ that ensure the underlying lattice problems (SVP/CVP) are computationally hard. This includes choosing a sufficiently large dimension n and an appropriate error distribution $\mathcal{X}$.

Summary of broken parameter settings

Algorithm	(Some) Broken Parameter setting
Arora-Ge	$m = \Omega(n^B)$ samples + time where $\lvert\text{Supp}(\mathcal{X})\rvert \leq B < q$ [13]
Blum-Kalai-Wasserman	$m > q^{n/\log(q/n)}$ [13]
Lattice Reduction	$m = \text{poly}(n, \log q)$ and $q/B = \Omega(2^n)$ and $\text{poly}(n, \log q)$ time [13]
Primal Attack	$\sigma\sqrt{b} \leq \sigma_0^{2b-d}.\text{Vol}(B)^{\frac{1}{d}}$ [14]

Implementation

All scripts can be found here

Previous work

Bit complexity estimation for various attack strategies on Kyber512. Complexity of lattice-based attacks are computed via the lattice estimator. [15]

Method	BKW	Primal-uSVP	BDD	Dual	Dual Hybrid
Samples	2^{167}	512	512	512	512
Complexity (bits)	179	144	140	150	140

Bit complexity estimation for various attack strategies on CRYSTALS-Dilithium (n = 1024, k = 4, q = 8380417, σ = 3.16). [12]

Method	primal-uSVP	Dual
Samples	1024	1024
Complexity (bits)	125	125

Proposed Solution

Secure Parameter Selection

- The most critical aspect of securing LWE-based cryptography is the careful selection of parameters $(n, q, \mathcal{X})$. Adhering to standardized parameter sets, such as those recommended by the NIST Post-Quantum Cryptography standardization process, is crucial. These parameters are chosen based on extensive cryptanalysis to provide a desired security level.
- **Importance of n, q, and Error Distribution $\mathcal{X}$:** The dimension n, modulus q, and error distribution $\mathcal{X}$ directly influence the hardness of the underlying lattice problems. Increasing n and choosing q appropriately (not too large or too small) makes lattice reduction attacks more difficult [3]. The error distribution $\mathcal{X}$ must be carefully chosen to ensure sufficient noise to hide the secret, while still allowing for correct decryption.

Robust Implementation Practices

- **Secure Random Number Generation:** The generation of random numbers, especially for the secret s, must use a cryptographically secure pseudo-random number generator (CSPRNG) to ensure unpredictability and prevent statistical biases that could be exploited by attacks.

Deployment

Recommended LWE Schemes and Parameters for Specific Applications

For practical deployment, it is recommended to use LWE-based schemes that have undergone examinations and analysis and are part of standardization efforts. For instance, some selected schemes for the purpose of key exchange, based on LWE problem:

- ML-KEM is a NIST-selected KEM for general encryption, built on Module-LWE based on CRYSTALS-Kyber [11].
- FrodoKEM is another strong candidate but failed in round 3 because of performance reasons. FrodoKEM is built on top of FrodoPKE, which is a public key encryption (PKE) algorithm, can be used for encrypting fixed length messages, offering IND-CPA security.

References

[1]. Albrecht, M.R. On Dual Lattice Attacks Against Small-Secret LWE and Parameter Choices in HElib and SEAL. In Advances inCryptology—EUROCRYPT 2017; Springer International Publishing: Berlin/Heidelberg, Germany, 2017; pp. 103–129

[2]. Kannan R. Minkowski's convex body theorem and integer programming. Math Oper Res, 1987, 12: 415–440

[3]. Köppl, T.; Zander, R.; Henkel, L.; Tcholtchev, N. A parameter study for LLL and BKZ with application to shortest vector problems. arXiv, 2025

[4]. Zhang, X.; Zheng, Z.; Wang, X. A detailed analysis of primal attack and its variants. Tsinghua University Institutional Knowledge Repository, n.d.

[5]. Richter, M.; Seidensticker, J.; Bertram, M. A (somewhat) gentle introduction to lattice-based post-quantum cryptography. 2023, June 14.

[6]. Arora, S.; Ge, R. New Algorithms for Learning in Presence of Errors. In Aceto, L.; Henzinger, M.; Sgall, J. (Eds.), Proceedings of the 38th International Colloquium on Automata, Languages and Programming (ICALP); Lecture Notes in Computer Science, Vol. 6755; Springer Verlag: Berlin/Heidelberg, Germany, 2011; pp. 403–415.

[7]. DI Management. A simple lattice-based encryption scheme, 2024.

[8]. MDPI. Improvements on Making BKW Practical for Solving LWE. (n.d.).

[9]. https://65610.csail.mit.edu/2024/lec/l20-lweattack.pdf

[10]. Albrecht, M.R.; Fitzpatrick, R.; Scott, M. On the Complexity of the BKW Algorithm on LWE. Cryptology ePrint Archive, 2012.

[11]. https://csrc.nist.gov/projects/post-quantum-cryptography/selected-algorithms

[12]. https://estimate-all-the-lwe-ntru-schemes.github.io/docs/

[13]. https://people.csail.mit.edu/vinodv/CS294/lecture2.pdf

[14]. https://www.maths.ox.ac.uk/system/files/attachments/lattice-reduction-and-attacks.pdf

[15]. https://github.com/malb/lattice-estimator