2-2022

Provably Weak Instances of PLWE Revisited, Again

Katherine Mendel
College of Saint Benedict/Saint John's University, kmendel001@csbsju.edu

Provably Weak Instances of PLWE Revisited, Again

Katherine Mendel
College of Saint Benedict and Saint John's University

February 2021

Approved by:

Sunil Chetty

Thesis Advisor
Professor of Mathematics

Imad Rahal

Faculty Reader
Chair, Professor of Computer Science

Kristen Nairn

Faculty Reader
Professor of Mathematics

Noreen Herzfeld

Faculty Reader
Professor of Computer Science, Theology

Channa Kumarage

Faculty Reader
Professor of Computer Science

Anne Sinko

Chair, Professor of Mathematics

Abstract

Learning with Errors has emerged as a promising possibility for post-quantum cryptography. Variants known as RLWE and PLWE have been shown to be more efficient, but the increased structure can leave them vulnerable to attacks for certain instantiations. This work aims to identify specific cases where proposed cryptographic schemes based on PLWE work particularly poorly under a specific attack.

Contents

1 Introduction

1.1 A Brief History of Cryptography

Many scholars believe that cryptology, the science of secret writing, has been around almost as long as written communication itself. The first evidence of encoding messages dates back to as early as 1900 BCE from ancient Egyptian, Greek and Babylonian civilizations. During this period, secret messages were based on linguistic techniques like manual substitution and transposition. These ciphers were mainly based on private key schemes, meaning they could only be decoded by someone who had a specific secret piece of information, the same secret information used to encode the message. Secret information could be in the form of a code word or set of substitutions.

A substitution cipher is based on taking certain letters or phrases and swapping them for another. The two parties that are communicating agree upon the private key, which is the set of letter to letter substitutions for a substitution cipher.

Example 1.1.1. *To encode the phrase 'a brief history of cryptography', first remove all spaces, then substituting each letter according to Figure 1, the corresponding encoded message would be 'zyirvusrhglibluxibkgltizksb'.*

a b c d e f g h i j k l m n o p q r s t u v w x y z
↓
z y x w v u t s r q p o n m l k j i h g f e d c b a

Figure 1: Example private key for a substitution cipher

However, as time went on, more sophisticated cryptanalysis, the study of techniques to break secret writing systems, spurred the need for more complex cryptographic techniques. Cryptography is the term used for describing the study of techniques used to create systems for secret writing. Frequency analysis was a mathematical technique used to break many subsitution ciphers. The letters of a given alphabet appear with a certain frequency in each language. By analysing and comparing the frequency of letters in a cipher, along with guessing and slight adjustments, simple subsitution ciphers can be broken. Another weakness of private key schemes is that private keys and code books were hard to keep private, and cryptanalysts were able to break many of the ciphers without even needing to know the secret keys. As a result, cryptography has become increasingly automated and mathematical. [1]

One notable development in the field of cryptology occurred in 1976, when Whitfield Diffie and Martin Hellman published a work which proposed the idea of public key cryptography. Public key cryptography calls for an algorithm that can encode messages using a public key which can then only be decoded with a very specific secret key. While the public key is made available by an individual to allow others to encode messages to send to them, the complimentary secret key is known by the individual decoding the message alone. The work Diffie

and Hellman published laid the groundwork for exchanging keys over an insecure channel of communication and verifying the identity of parties involved in communication, concepts which secure much of the digital world today. While Diffie and Hellman came up with the concept of public key cryptography, they provided no algorithms that fulfilled the requirements. About a year later, MIT professors Ron Rivest, Adi Shamir and Len Adleman introduced the RSA algorithm [2], whose security is based on the computational difficulty of finding the prime factors of a large composite number. Clifford Cocks discovered an equivalent algorithm in 1950 during his time working at the United Kingdom Government Communications Headquarters, but his work was classified [3]. RSA has been the primary algorithm for encryption online since the beginning of the 1980s. While the recommended key size has been increasing because of the ingenuity of cryptanalysts and hackers as well as increased computing power, it is still generally considered secure.

Algorithm 1.1 RSA Encryption Key Generation

1: Let p and q be two randomly generated large primes
2: **procedure** KEY GENERATION
3: Let $n \leftarrow p \cdot q$
4: $\phi \leftarrow (p-1)(n-1)$
5: Choose an integer e such that $1 < e < \phi$ and $gcd(e, \phi) = 1$
6: Compute secret exponent d such that $1 < d < \phi$ and $ed \equiv 1 \pmod{\phi}$
7: **end procedure**
8: **public key** (n, e)
9: **private key** (n, d)

Then the plain-text, M, is represented as number between 0 and $n-1$ based on a conversion method agreed upon between the communicating parties. Then M is encrypted as the cipher text C by computing $C = M^e \pmod{n}$. Decryption is accomplished by computing $M = C^d \pmod{n}$.

Example 1.1.2. *To illustrate the RSA cryptography scheme, consider this simple example of the RSA algorithm:*

1. *Let* $p = 3$ *and* $q = 11$ *be two "randomly generated" "large" primes*
2. *Let* $n = pq = 3 \cdot 11 = 33$ *and* $\phi = (p-1)(n-1) = 2 \cdot 10 = 20$
3. *Let* $e = 3$ *and verify that* $1 < e < \phi$ *and* $gcd(e, \phi) = 1$
4. *Let* $d = 7$ *and verify that* $1 < d < \phi$ *and* $ed \equiv 1 \ (mod\ \phi)$. *It is true that* $21 \ (mod\ 20) \equiv 1$.

Thus we obtain a public key $(n, e) = (33, 3)$ *and private key* $(n, d) = (33, 7)$.

Consider a scenario where Alice would like to send the message '5' to Bob. First she takes his public key and encrypts the message using the formula discussed earlier.

$$C = M^e \ (mod\ n) = 5^3 \ (mod\ 33) = 26$$

Bob receives the cipher text '26' and decodes it using his private key.

$$M = C^d \ (mod\ n) = 26^7 \ (mod\ 33) = 5.$$

To use RSA in practice, n, known as the modulus, must be a large number to be secure. The larger the modulus, the higher the security. The size of a given modulus is measured by the number of bits needed to store the number on a classical computer. Currently, modulus sizes of 2048 or 4096 bits are recommended. [4]

Another example of public key cryptography is Elliptic Curve Cryptography [5] [6] which is considered to be faster than RSA because it achieves greater security with smaller key sizes. Generally, the larger the key size for a certain algorithm, the longer it will take to encrypt and decrypt messages. Specific mathematical properties of elliptic curves make them a good choice to generate shared keys for encryption that are secure and relatively short.

Quantum computers with the ability to break many encryption techniques used today are predicted to become reality in the near future. Shor's quantum algorithm in particular threatens to make efficient prime factorization of a large composite number, whose intractability is essential to the security of RSA, an achievable task. Thus, research into encryption techniques that could withstand the computational power of quantum computers, an area known as post-quantum cryptography, has been pursued with urgency. [7]

1.2 Quantum Computing

The vast majority of computers today use binary to store information with each bit being either one or zero. Quantum computing, on the other hand, leverages concepts from quantum theory as the basis for storing information in quantum bits, or qubits. Qubits can be either one, zero or both simultaneously, a state which is referred to as a superposition and will be described in further detail later.

The basis of quantum computation is quantum probability. Quantum probability can be seen as a modification of classical probability theory but instead of strictly positive probabilities, it uses complex numbers called probability amplitudes. The squares of the absolute value of these probability amplitudes represent probabilities. The following two rules apply when combining probability amplitudes:

1. When an event can happen in a sequence of independent steps, multiply the amplitudes of each step.

2. When an event can happen in several alternative ways, add the amplitudes of each way.

Though these rules appear simple, they have huge implications for computing.

Consider an experiment known as the double slit experiment as described by [8]. In this experiment, particles are emitted from source S and pass through a barrier with two slits to reach a detector D. Let the particle have a probability amplitude z_1 for passing through the top slit and z_2 for the bottom. Thus, the probability p_1 for taking the path with the upper slit is $|z_1|^2$ whereas the probability of taking the path with the lower slit $p_2 = |z_2|^2$. Based on classical probability theory, the particle should reach D with probability $p_1 + p_2 = |z_1|^2 + |z_2|^2$ since these are mutually exclusive events.

However, according to the quantum rules outline earlier, the probability p of the particle reaching D must be calculated by adding the amplitudes and then squaring the absolute value of that value.

$$p = |z|^2 = |z_1 + z_2|^2 = |z_1|^2 + |z_2|^2 + z_1^{\cdot} z_2 + z_1 + z_2^{\cdot} \tag{1}$$

It follows that

$$p = p_1 + p_2 + 2\sqrt{p_1 \cdot p_2} \cdot cos(\varphi_1 - \varphi_2) \tag{2}$$

Without the term $2\sqrt{p_1 \cdot p_2} \cdot cos(\varphi_1 - \varphi_2)$, known as the interference term, the probability calculation would follow the classical theory of probability. Thus, depending upon what the value of the relative phase $(\varphi_1 - \varphi_2)$ is, the interference term can either enhance or suppress the overall probability p.

The relative phase is more important than the actual quantity of φ_1 and φ_2 themselves. The behavior of the particle is influenced only by the difference of the two phases. However since each value pertains to a separate path, then the particle must have experienced both paths. If the particle only traveled along one path, then it would not make sense for the interference term to contain values unique to each individual path. So the particle did not strictly travel either through the upper or the lower slit. Instead, it travelled through both. Though this may seem impossible, actual results from running the double slit experiment support this theory.

Quantum computers rely on the same principles. They follow all computational paths at the same time, producing answers that depend on all these alternative calculations. This also explains how qubits may be in either of the base states 0 or 1, or in an intermediate state known as a super position. These superpositions are typically described using $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ which denotes the qubit with amplitude α in the basis state 0 and β in the basis state 1.

1.3 Shor's Quantum Algorithm

One particularly interesting application for quantum computation is Shor's algorithm, which is able to efficiently find prime factorizations, which, as mentioned earlier, undermines the security of RSA. Recall that on a classical computer, given a, n and e, with $0 < a < n$ and $1 < e$ calculating $a \cdot e(mod(n))$ is easy, whereas calculating $a^e \cdot (mod(n))$ is hard for large n. However, Shor's algorithm makes calculating $a^e(mod(n))$ computationally feasible.

The algorithm is named after Peter Shor, a mathematician who came up with the idea while working at Bell Laboratories. His work was inspired by Jim Simon's algorithm that showed exponential speedup with quantum computers when applied to the problem of finding the period on the vertices of a high dimensional cube. Shor realized that finding periodicity would help solve the discrete log problem and, shortly after, the factoring problem. The algorithm is composed of two parts, the first transforms the factoring problem into the problem of finding the period of a function. The second leverages the Quantum Discrete Fourier (QDF) transform to find the desired period with high probability. Quantum computation allows the QDF to test all values of the periodic function simultaneously which accounts for the incredible speed at which the prime factorizations can be found. [9] Estimates suggest that a quantum computer with twenty million qubits could break the standard 2048-bit RSA encryption in eight hours. For comparison, the highest RSA number factored on a classical computer was RSA-768, which has 768 bits, and took two years to compute. [10]

Luckily, there are many bottlenecks that quantum computing must overcome before Shor's algorithm can truly undermine the security of RSA. To date, the largest quantum computer is IBM's Eagle processor [11] which has 127 qubits. Additionally, the largest number that Shor's Algorithm has factored successfully is 21 [12]. While quantum computers pose no immediate threat, it is still important to develop post-quantum cryptographic techniques for when they become more advanced in the future.

1.4 The LWE Problem

This thesis will focus on the theoretical and computational aspects of the Learning with Errors (LWE) problem introduced by [13]. The LWE problem is believed to be quantum resistant, and has many cyptographic applications including identity based encryption and fully homomorphic encryption. Fully homomorophic encryption is a large motivation behind the development of LWE schemes because it allows computations to be performed on encrypted data without ever decrypting it, which can be useful for processing sensitive data in the cloud, among other scenarios.

There are two versions of the LWE problem, search and decision. This thesis will focus on the search problem, in which the goal is to find an efficient algorithm to solve a noisy system of linear congruences.

Consider $n \geq 0$, prime q and some error distribution, typically a Gaussian distribution, χ on $(\mathbb{Z}/q\mathbb{Z})$. Let A be a matrix with an arbitrary number of rows where each row, $a_i \in (\mathbb{Z}/q\mathbb{Z})$, is taken from a uniformly random distribution. Then selecting some error, e_i, for each row from χ, construct a vector b, where each entry of $b_i = \langle a_i, s_i \rangle + e_i$ for each integer i. Given n number of sample pairs in the form (a_i, b_i), this can be rewritten as a system of linear congruences modulo q like the following

$$\begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix} \cdot \begin{pmatrix} s_1 \\ s_2 \\ \vdots \\ s_n \end{pmatrix} \approx \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

If there is no error, then solving for the secret s could be accomplished very efficiently using linear algebraic techniques, like Gauss-Jordan elimination. Introducing perturbations into the system, however, makes it unlikely the secret will be recoverable by Gauss-Jordan elimination due to error amplification.

Example 1.4.1. *Consider the following system of equations with all operations performed modulo 1373:*

$$\begin{gathered} 757 \cdot x + 1028 \cdot z + 411 \cdot y + 181 = 0 \\ 479 \cdot x + 543 \cdot z + 432 \cdot y + 85 = 0 \\ 237 \cdot x + 84 \cdot z + 1301 \cdot y + 1235 = 0 \\ 235 \cdot x + 223 \cdot z + 1274 \cdot y + 312 = 0 \\ 866 \cdot x + 962 \cdot z + 657 \cdot y + 441 = 0 \end{gathered}$$

This system was constructed so that $x = 103, z = 1072, y = 693$. The system can be represented by the following matrix:

$$\begin{pmatrix} 757 & 1028 & 411 & 0 & 1192 \\ 479 & 543 & 432 & 0 & 1288 \\ 237 & 84 & 1301 & 0 & 138 \\ 235 & 223 & 1274 & 0 & 1061 \\ 866 & 962 & 657 & 0 & 932 \end{pmatrix}$$

Using Gauss-Jordan elimination to obtain an equivalent representation of the system in row-echelon form results in the correct solution for all indeterminates.

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 103 \\ 0 & 1 & 0 & 0 & 1072 \\ 0 & 0 & 1 & 0 & 693 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Example 1.4.2. *The following system of equations, is the same as Example 3, but with small errors injected into three of the rows.*

$$\begin{pmatrix} 757 & 1028 & 411 & 0 & 1192 \\ 479 & 543 & 432 & 0 & 1288 \ \boldsymbol{+\ 1} \\ 237 & 84 & 1301 & 0 & 138 \ \boldsymbol{+\ 1} \\ 235 & 223 & 1274 & 0 & 1061 \\ 866 & 962 & 657 & 0 & 932 \ \boldsymbol{+\ 1} \end{pmatrix}$$

Performing Gauss-Jordan elimination on the new system shows that even introducing very small errors left essentially no information about the original solution.

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

1.5 Variants of Learning With Errors

The same concepts discussed in the definition of the LWE problem earlier can also be applied to different rings, a variant which is known as ring learning with errors [14]. This variant of LWE is interesting because it has been shown to have better efficiency. However, the extra structure may allow for special attacks that take advantage of the structure.

In order to understand RLWE, it is first important to understand rings, ideals and quotient rings [15].

Definition 1.5.1 (Ring)**.** *A set S under two binary operators + an * (typically referred to as addition and multiplication, respectively) is considered a* ***ring*** *if the following conditions are satisfied:*

1. *Additive associativity: For all $a, b, c \in S$, $(a + b) + c = a + (b + c)$*
2. *Additive commutativity: For all $a, b \in S$, $a + b = b + a$*
3. *Additive identity: There exists an element $0 \in S$ such that for all $a \in S$, $0 + a = a + 0 = a$*
4. *Additive inverse: For every $a \in S$ there exists $a^{-1} \in S$ such that $a + a^{-1} = a^{-1} + a = 0$*
5. *Left and right distributivity: For all $a, b, c \in S$, $a * (b + c) = (a * b) + (a * c)$ and $(b + c) * a = (b * a) + (c * a)$*
6. *Left and right distributivity: For all $a, b, c \in S$, $a * (b + c) = (a * b) + (a * c)$ and $(b + c) * a = (b * a) + (c * a)$*
7. *Multiplicative associativity*: For all $a, b, c \in S$, $(a * b) * c = a * (b * c)$ (a ring satisfying this property is sometimes explicitly termed an associative ring).*

* *There exist some non-associative rings, but multiplicative associativity is often included as a requirement in formal definitions nontheless.*

Definition 1.5.2 (Ideal)**.** *An ideal is a subset I of elements in a ring R that forms an additive group and has the property that, whenever $x \in R$ and $y \in I$, then $x \cdot y \in I$ and $y \cdot x \in I$.*

Definition 1.5.3 (Quotient Ring)**.** *If $\mathfrak{a}$ is an ideal of A, then the quotient ring $A/\mathfrak{a}$ is the set of equivalence classes under the relation $[x] - [y]$ iff $x - y$ in $\mathfrak{a}$. Additionally, the operations of $A/\mathfrak{a}$ come from A itself.*

Example 1.5.4. *Consider the quotient ring $\mathbb{Z}$ with the ideal multiples of 4 ($4\mathbb{Z}$). The quotient ring is $\mathbb{Z}_4 = \mathbb{Z}/4\mathbb{Z}$. The equivalence classes are as follows:*

$$[0] = 0, 4, 8, ..., 4x$$
$$[1] = 1, 5, 9, ..., 4x + 1$$
$$[2] = 2, 6, 10, ..., 4x + 2$$
$$[3] = 3, 7, 11, ..., 4x + 3$$

Example 1.5.5. *A slightly more complex quotient ring is $\mathbb{Z}[x]/(x^2 + 1)$. In this case, the quotient ring contains polynomials with integer coefficients. The equivalence classes are based on the remainder after Euclidean divisions by $x^2 + 1$.*

Theorem 1.5.6 (Maximal Ideals and Field)**.** *Given a field K and $p(x) \in K[x]$. Then the following conditions are equivalent:*

1. *$p(x)$ is irreducible over K.*
2. *$J = \langle p(x) \rangle$ is a maximal ideal in $K[x]$.*
3. *$K[x]/J$ is a field, where $J = \langle p(x) \rangle$.*

[16]
With these building blocks we can construct the following definitions:

Definition 1.5.7 (Search non-dual RLWE)**.** *The search RLWE problem is to find a random, but fixed, secret $s \in R_q$ where R_q is a ring, given many independent samples of the form $(a_i, b_i = s \cdot a_i + e_i \ (mod\ qR)) \in R_q \times R_q$, where each a_i is sampled from a uniform distribution on R_q and each e_i is drawn from the error distribution.*

A variant of RLWE known as polynomial learning with errors (PLWE) was introduced in [17], and can be defined as follows:

Definition 1.5.8 (Search PLWE)**.** *Let $f \in \mathbb{Z}[x]$ be a monic irreducible polynomial of degree n and let $q \geq 2$ be an integer modulus. Consider the quotient ring $P = \mathbb{Z}[x]/(f)$ and denote with P_q the residue ring P/qP. Given entries from a matrix a, a_i uniformly sampled from P_q and errors drawn from the error distribution on P_q. For a random but fixed choice of s uniformly sampled from P_q, the search PLWE problem is to recover s with high probability from arbitrarily many independent samples of the form (a_i, b_i) where $b_i = a_i \cdot s_i + e_i \ (mod\ qP)$.*

The error is typically sampled from a distribution centered around zero so that the errors are "small." Often this is done through using a discretized Gaussian distribution. In [18] they sample the error from the spherical Gaussian on $\mathbb{R}^n$ with parameter r, but interpreting it as a distribution on $P \otimes \mathbb{R}$. The standard basis of $\mathbb{R}^n$ is mapped to the basis $1, x, x^2, ..., x^{n-1}$ of P. For the purposes of this work, errors with either be sampled from the Gaussian distribution on P_q, or uniformly sampled from a set of polynomials with restricted coefficients and degree.

Example 1.5.9. *To illustrate a possible set up for PLWE, consider the scenario where* $f = x^6 + 18 \cdot x + 3$, $q = 19$ *such that* $P = \mathbb{Z}[x]/(x^6 + 18 \cdot x + 3)$. *Let* $s = 18 \cdot x + 18$. *Taking four uniformly random samples from* P_q, $a = (2x+18, 2 \cdot x+1, 7 \cdot x+14, x+14)$. *The corresponding errors were also randomly sampled to be* $e = (x + 1, x, 0, x + 1)$. *Thus,* $b = (17 \cdot x^2 + 2, 17 \cdot x^2 + 17 \cdot x + 18, 12 \cdot x^2 + 17 \cdot x + 5, 18 \cdot x^2 + 5 \cdot x + 6)$.

A key aspect of the attack leveraged in this research is that the chosen polynomial remains irreducible modulo q. The Frobenius Density Theorem (discussed in further detail as part of Theorem 4.0.1) describes the density of primes where a polynomial is irreducible. For this reason, for later results, it is necessary to have some knowledge of group theory. Readers who are already familiar with these concepts can skip to Section 1.6.

Definition 1.5.10 (Group)**.** *A group* $(G, *)$ *is a set* G *together with a binary operation * with the following properties.*

1. *The set* G *is closed under **.*
2. *The operation * is associative.*
3. *There is an identity element* $e \in G$ *such that for all* $g \in G$, $e*g = g*e = g$.
4. *There is a corresponding inverse for each* $g \in G$ *called* g' *where* $g' \in G$ *and* $g * g' = g' * g = e$.

This research is concerned with so called Galois groups which deal with field extensions and the intermediate fields. Informally, these Galois groups can be related to the factorization of a polynomial f.

Example 1.5.11. *Let* $f = x^2 + 1$. *In the reals,* $x^2 + 1$ *is irreducible. However, considering the extension including the complex number* i, *then* f *can be factored into linear factors, namely* $f = (x+i)(x-i)$. *Galois Theory deals with the roots and valid permutations of these roots (for the specific details on this process refer to [19]).*

Another group of interesting polynomials to consider are the cyclotomic polynomials, polynomials irreducible over the reals where the roots are the complex roots of unity. Consider the degree three cyclotomic polynomial $\Phi_3(x) = x^2 + x + 1$. This is irreducible in the reals. Consider the roots of unity denoted by ζ_k for the k^{th} root of unity. Then, $\Phi_3(x)$ factors into the terms as $(x - \zeta_3)(x - \zeta_3)^2$ when the reals are extended with ζ_2 and ζ_2. Similar to above, the Galois group of this splitting field can be viewed as the permutations of the roots of $\Phi_3(x)$.

Based on these properties of Galois groups, another interesting family of groups to consider are the symmetric groups. One intuitive way to understand this group is by viewing the symmetric group of size l, or S_l, as all possible permutations of l elements. In other words under the binary operation is composition of bijective functions $\gamma : \{1, 2, \ldots, l\} \to \{1, 2, \ldots, l\}$. Consider S_3, with elements A, B, and C. All possible permutations are as follows:

$$\begin{array}{ccc} A\ B\ C & A\ C\ B & B\ A\ C \\ B\ C\ A & C\ A\ B & C\ B\ A \end{array}$$

Definition 1.5.12 (Order of a group)**.** *The order of a group G refers to the number of elements it contains.*

For S_3 from above, the order is 6. In mathematical notation, this can be written as $\mid S_3 \mid = \#S_3 = 6$.

Definition 1.5.13 (Subgroup)**.** *A subset H of a group G is a subgroup if H itself is a group under the binary operation in G.*

The symmetric group S_2 is a subgroup of S_3 because it is a group itself and it is contained in S_3 (in the elements $A\ B\ C$ and $A\ C\ B$).

Groups also have the following properties:

Definition 1.5.14 (Left cosets of a subgroup)**.** *Given a group G and $a \in G$, the left coset of H containing a is $aH = \{ah : h \in H\}$.*

Definition 1.5.15 (Index of a group)**.** *The index of a subgroup $H \leq G$, denoted $(G : H)$ is the cardinality of the quotient group G/H, which is equal to the number of left cosets of H in G.*

Theorem 1.5.16 (Lagrange's Theorem)**.** *Suppose that G is a finite group. The order of any subgroup $H \leq G$ divides the order of G.*

Theorem 1.5.17 (Index of a subgroup)**.** *If G is a finite group and $H \leq G$, then $(G : H) = |G|/|H|$.*

Theorem 1.5.18 (Cauchy's Theorem)**.** *Let G be a finite group and p be a prime that divides the order of G, then G must contain an element of order p.*

1.6 LWE Based Cryptographic Scheme

To see how LWE works in a cryptographic sense, let Alice and Bob be two parties attempting to communicate with one another using an LWE based crytographic scheme [20]. First, Alice chooses the secret key s and a random m by n matrix A. Alice's public key is $(A^\intercal, b)$, where each entry of b is calculated as described earlier.

To illustrate, let $n = 1$, $q = 709$, $s = 145$ and $m = 10$.

$$\text{Let } A = \begin{pmatrix} 428 \\ 106 \\ 323 \\ 56 \\ 125 \\ 347 \\ 526 \\ 575 \\ 581 \\ 343 \end{pmatrix} \text{ and } e = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ -1 \\ -1 \\ -1 \\ -1 \\ -1 \\ 0 \end{pmatrix}, \text{ now calculate } b = \begin{pmatrix} 377 \\ 481 \\ 42 \\ 321 \\ 399 \\ 686 \\ 406 \\ 423 \\ 582 \\ 105 \end{pmatrix}.$$

If Bob wants to send an encoded bit to Alice, Bob first chooses an m-dimensional vector r, with entries of either 0 or 1.

$$\text{For the example, let } r = (0, 1, 1, 0, 1, 0, 0, 0, 0, 1).$$

Bob then multiplies r to the right of $A^\intercal$. This multiplication, $u = A^\intercal r$, results in vector, known as the preamble, that does not expose any secret information to anyone who intercepts it:

$$u = A^\intercal r = (0, 106, 323, 0, 125, 0, 0, 0, 0, 343)$$

Next, Bob computes $u = br$ and adds the product of his message bit μ, again either 0 or 1, and $\lfloor 2q \rfloor$, to compute the main cipher text, $u' = br + \mu\lfloor 2q \rfloor$ he will send to Alice. Based in the construction, this will either be br for $\mu = 0$, or something very far from br, for $\mu \neq 0$. Anyone who intercepts this information will not be able to tell which one it is.

$$\text{If } \mu = 1,\ u' = br = (354, 835, 396, 354, 753, 354, 354, 354, 354, 459).$$
$$\text{If } \mu = 0,\ u' = br + \lfloor 709/2 \rfloor = (0, 481, 42, 0, 399, 0, 0, 0, 0, 105).$$

In order to decrypt the message, Alice takes u' and subtracts out su where s is her secret key. The result of this subtraction will be approximately 0 when $\mu = 0$, or approximately $q/2$ wnen $\mu = 1$.

In the example from above, when

$$\mu = 0,\ u' - su = (0, 0, 1, 0, 708, 0, 0, 0, 0, 0).$$

Note that each entry $\approx 0 \pmod{709}$.

Alternatively, if

$$\mu = 1,\ u' = br + \mu\lfloor 709/2 \rfloor = (354, 354, 355, 354, 353, 354, 354, 354, 354, 354).$$

Note that each entry $\approx q/2 \pmod{709}$.

The example from this section outlines how to send a single bit encoded under an LWE encryption scheme. In practice, both the size of the message and the secret key would be substantially larger.

1.7 Hardness Guarantees

Research over the last decade has shown lattices to be a very promising mathematical foundation for cryptography. The appeal of lattice-based systems is that their security can be based on worst-case hardness assumptions, that they appear be quantum resistant, that they can be quite efficient, and that for certain advanced tasks, such as fully homomorphic encryption, no other cryptographic assumption is known to suffice. [21] shows that that the Learning with Errors (LWE) problem is classically at least as hard as standard worst-case lattice problems, even with polynomial modulus.

2 Background

2.1 Arora-Ge Algebraic Attack

As part of LWE, the errors are drawn from a finite set, a fact which makes it possible to attack LWE and recover the secret without knowing the secret key. Arora and Ge proposed an attack in [22] that leverages this property using algebraic methods to recover the secret.

For an intuitive example from [23] of the ideas behind this proposed attack, consider S as $\{0, 1\}$ and let $n = 1$. Take samples (a, b) where $b = a \cdot s + e$ and $e \in \{0, 1\}$. Then we know that

$$(b + a \cdot u) \cdot (b + a \cdot u - 1) \equiv 0 \pmod{q} \tag{3}$$

where u is the unknown. Since e is either 0 or 1, then either the first or second term must equal 0 when u equals $-s$. Through expanding the equation above, we obtain

$$b \cdot (b - 1) + (2b - 1) \cdot au + a^2u^2 \tag{4}$$

Then, linearize that equation by replacing u with u_1 and u^2 with u_2 where u_1 and u_2 are independent variables. After substitution, we obtain

$$b \cdot (b - 1) + (2b - 1) \cdot au_1 + a^2u_2 \tag{5}$$

Substituting $b = as + e$ to obtain

$$p'(a) = e(e - 1) + (2e - 1)(s + u1) \cdot a + (u_2 + 2su_1 + s2) \cdot a^2 \tag{6}$$

$$= (2 \cdot e - 1)(s + u_1) \cdot a + (u2 + 2su_1 + s2) \cdot a^2 \equiv 0 \pmod{q}$$

Now, we can think of this as a polynomial in a with coefficients chosen independent of a, for any fixed u_1 and u_2. There are no solutions with $u_1 \neq -s$. For a (u_1, u_2) where $u_1 \neq -s$. Then, $p'(a)$ is a non-zero polynomial in a which is 0 with probability at most $2/q$ over the choice of a. For the full proof, see [22].

2.2 Example of Arora-Ge

Consider an encryption scheme with most of same parameters as described above. Namely, with $n = 1$ but with the modification that $e \in \{-1, 0, 1\}$. Let the prime $q = 17$, and secret $s = 6$. Based on the recommended number of samples from [23], 14 integers were randomly selected. Let $a = (7, 10, 12, 9, 16, 12, 13, 5, 5, 10, 4, 0, 15, 16)$ and $e = (0, -1, -1, -1, 1, 0, 1, -1, 0, -1, -1, 0, 0, 1)$. The random values for a and e were obtained using SageMath. Thus, $b = (8, 8, 3, 2, 12, 4, 11, 12, 13, 8, 6, 0, 5, 12)$ where each $b[i]$ is calculated such that $b[i] = a[i] \cdot s + e[i \pmod{17}]$. For example, $b[0] = 7 \cdot 6 + 0 = 8$.

Following the Arora-Ge attack, take each $b[i] - a[i] \cdot u - e_i$ and for each possible error e_i (-1, 0 and 1), then multiply those together to obtain a polynomial of degree three or less. In the example we are looking at, for b[0], we obtain

$$(8 - 7 \cdot s) \cdot (8 - 7 \cdot s + 1) \cdot (8 - 7 \cdot s - 1)$$

Multiply those together to get $14 \cdot s^3 + 3 \cdot s^2 + 6 \cdot s + 11$. Repeat the process for each element of b, to obtain 14 polynomials represented below in array p where $p[i]$ represents the polynomial that corresponds with $b[i]$.

$$\begin{gathered}
p[0] = 14 \cdot u^3 + 3 \cdot u^2 + 6 \cdot u + 11, \\
p[1] = 3 \cdot u^3 + 3 \cdot u^2 + 11 \cdot u + 11, \\
p[2] = 6 \cdot u^3 + 4 \cdot u^2 + 11 \cdot u + 7, \\
p[3] = 2 \cdot u^3 + 10 \cdot u^2 + 3 \cdot u + 6, \\
p[4] = u^3 + 2 \cdot u^2 + 6 \cdot u + 16, \\
p[5] = 6 \cdot u^3 + 11 \cdot u^2 + 14 \cdot u + 9, \\
p[6] = 13 \cdot u^3 + u^2 + 3 \cdot u + 11, \\
p[7] = 11 \cdot u^3 + 16 \cdot u^2 + 4 \cdot u + 16, \\
p[8] = 11 \cdot u^3 + 6 \cdot u^2 + 3 \cdot u + 8, \\
p[9] = 3 \cdot u^3 + 3 \cdot u^2 + 11 \cdot u + 11, \\
p[10] = 4 \cdot u^3 + 16 \cdot u^2 + 14 \cdot u + 6, \\
p[11] = 8 \cdot u^3 + 9 \cdot u^2 + 12 \cdot u + 1, \\
p[12] = u^3 + 2 \cdot u^2 + 6 \cdot u + 16, \\
p[13] = 9 \cdot u^3 + 8 \cdot u^2 + 5 \cdot u + 16
\end{gathered}$$

After linearizing, the equations are as follows:

$$\begin{gathered}
p[0] = 14 \cdot u_3 + 3 \cdot u_2 + 6 \cdot u_1 + 11, \\
p[1] = 3 \cdot u_3 + 3 \cdot u_2 + 11 \cdot u_1 + 11, \\
p[2] = 6 \cdot u_3 + 4 \cdot u_2 + 11 \cdot u_1 + 7, \\
p[3] = 2 \cdot u_3 + 10 \cdot u_2 + 3 \cdot u_1 + 6, \\
p[4] = u_3 + 2 \cdot u_2 + 6 \cdot u_1 + 16, \\
p[5] = 6 \cdot u_3 + 11 \cdot u_2 + 14 \cdot u_1 + 9, \\
p[6] = 13 \cdot u_3 + u_2 + 3 \cdot u_1 + 11, \\
p[7] = 11 \cdot u_3 + 16 \cdot u_2 + 4 \cdot u_1 + 16, \\
p[8] = 11 \cdot u_3 + 6 \cdot u_2 + 3 \cdot u_1 + 8, \\
p[9] = 3 \cdot u_3 + 3 \cdot u_2 + 11 \cdot u_1 + 11, \\
p[10] = 4 \cdot u_3 + 16 \cdot u_2 + 14 \cdot u_1 + 6, \\
p[11] = 8 \cdot u_3 + 9 \cdot u_2 + 12 \cdot u_1 + 1, \\
p[12] = u_3 + 2 \cdot u_2 + 6 \cdot u_1 + 16, \\
p[13] = 9 \cdot u_3 + 8 \cdot u_2 + 5 \cdot u_1 + 16
\end{gathered}$$

The system of equations can be represented as a matrix with the coefficients of the terms as the entries in the matrix. In the matrix representation below, the constant term c of each $p[i]$ is replaced by 0. Then, a column of values $17 - c$ is appended.

$$\begin{pmatrix} 14 & 3 & 6 & 0 & 6 \\ 3 & 3 & 11 & 0 & 6 \\ 6 & 4 & 11 & 0 & 10 \\ 2 & 10 & 3 & 0 & 11 \\ 1 & 2 & 6 & 0 & 1 \\ 6 & 11 & 14 & 0 & 8 \\ 13 & 1 & 3 & 0 & 6 \\ 11 & 16 & 4 & 0 & 1 \\ 11 & 6 & 3 & 0 & 9 \\ 3 & 3 & 11 & 0 & 6 \\ 4 & 16 & 14 & 0 & 11 \\ 8 & 9 & 12 & 0 & 16 \\ 1 & 2 & 6 & 0 & 1 \\ 9 & 8 & 5 & 0 & 1 \end{pmatrix}$$

Then, simply perform Gauss-Jordan Elimination on the matrix, which yields

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 12 \\ 0 & 1 & 0 & 0 & 2 \\ 0 & 0 & 1 & 0 & 6 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

The secret will be stored in the row where the corresponding coefficient is nonzero. In this case, the matrix reveals that $u_1 = 6$, which is correct. It is also important to verify that the relationship between linearized variables is correct, not just the secret itself (meaning $u_2 = (u_1)^2$). The correct relationship between variables holds in this example.

The Arora-Ge attack is probabilistic depending on the number of sample pairs (a_i, b_i) obtained. The more samples, the higher the chance of recovering the correct secret. [23] recommends much greater than $\binom{n+|S|}{|S|}$ samples for a high probability of success where n is the length of the secret and $|S|$ is the degree in the domain where the system of equations, p, lives. If the number of samples is much greater than $\binom{n+|S|}{|S|}$, there are more equations than unknowns which is why there are so many zero rows in the matrix. It is also possible to show that when m is large enough, s is the unique solution.

Example 2.2.1. *To illustrate the probabilistic nature of the Arora-Ge attack, consider the case of PLWE where* $q = 61$, $f = x^6 + 57 * x + 3$, $n = 3$ *and errors randomly sampled from* $\mathbb{Z}/2\mathbb{Z}$ *with degree less than 2.*

Figure 2 shows the results running the Arora-Ge on a randomly generated secret 100 times on each number of samples from 1 to 10 and measuring the percentage of correctly guessed secrets. As the number of samples increases, so does the percentage of correctly guessed secrets.

Recall that a high probability of correct guesses can be expected when the number of samples is much greater than $\binom{2+2}{2} = 6$. *Indeed, at 9 samples and above, the attack worked with 100% success. In this case, at 6 samples to success rate was 90%, which may be considered high probability, but it is important to note that this instantiation of PLWE is a small case, and we could expect to see a more well defined difference in probability in PLWE with a larger modulus and secret size.*

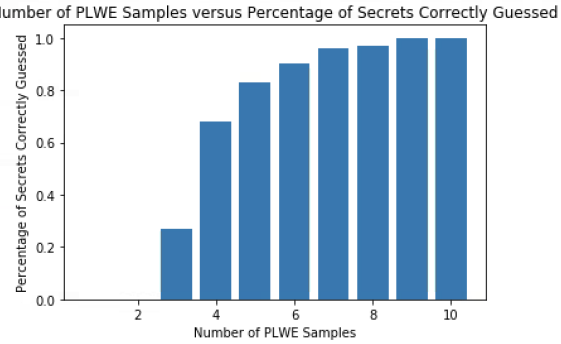

Figure 2: Percentage of Polynomial LWE Samples versus Percentage of Secrets Correctly Guessed

2.3 Gröbner Bases

In 1965, Bruno Buchberger introduced a concept called a Gröbner basis, and an algorithm, Buchberger's algorithm, to construct these bases, as part of his doctoral thesis [24]. They are named after his advisor, Wolfgang Gröbner. A Gröbner basis is a type of generating set for an ideal that has certain properties which simplify many difficult mathematical problems, like solving a system of polynomial equations.

To understand Gröbner bases, it is important to define ideals, generating sets, subtraction-polynomials (S-polynomials) and monomial orderings.

Definition 2.3.1 (Generating set)**.** *A generating set of an ideal* I *is a subset* S *such that every element of* I *can be expressed as a finite combination of elements from* S.

Definition 2.3.2 (Variety)**.** *A variety the set of solutions of a system of polynomial equations over a given field.*

Theorem 2.3.3 (Hilbert Basis Theorem [25])**.** *Every ideal $I \subseteq k[x_1, x_2, ..., x_n]$ has a finite generating set. Meaning $I = \langle g_1, g_2, ...g_t \rangle$ for some $g_1, g_2, ..., g_t \in I$.*

A term is the product of a coefficient and a monomial. The leading monomial of a polynomial P is denoted by $LM(p)$ and the least common multiple of two monomials x_a and x_b is denoted by $LCM(x_a, x_b)$. Using this notation, S-polynomials are constructed as follows:

$$S(p,q) = \frac{LCM(LM(p), LM(q))}{LT(p)} \cdot p - \frac{LCM(LM(p), LM(q))}{LT(q)} \cdot q \qquad (7)$$

To determine what the leading monomial of a polynomial is, there must be a predetermined monomial order imposed. Consider two monomials α and β under a monomial ordering where $\alpha > \beta$. This implies that for any other monomial γ, $\alpha + \gamma > \beta + \gamma$ and $\alpha \cdot \gamma > \beta \cdot \gamma$. This concept is especially important for calculating Gröbner bases because using a different monomial orderings may result in a different basis.

There are several different monomial orderings. To illustrate the differences consider the set of monomials $M = \{6xy^2z, 4z^2, 5x^3, x^2z^2\}$ with $x > y > z$ under each of the following orderings

1. **Lexicographic ordering**, also referred to as dictionary ordering, orders monomials based on the exponents of each individual variable. First all terms with some variable x_1 are ordered by decreasing exponent, then by the exponent of another variable x_2 in the case of a tie. This process is repeated until there are no monomials left to be sorted. For M lexicographic ordering means $5x^3 > x^2z^2 > 6xy^2z > 4z^2$.

2. **Graded lexicographic ordering** sorts monomials by total degree followed by lexicographic ordering in the case of a tie. Graded lexicographic ordering on M means $x^2z^2 > 6xy^2z > 5x^3 > 4z^2$.

3. **Graded reverse lexicographic order** sorts monomials by total degree followed by reverse lexicographic order in the case of a tie. For monomials where $x_1 > x_2 > ... > x_n$, when there is a tie for overall degree of a monomial, the term with the lowest exponent on the smallest variable (starting with x_n) is prioritized. Thus graded reverse lexicographic ordering of M means $6xy^2z > x^2z^2 > 5x^3 > 4z^2$.

Now, we have all of the pieces we need to define a Gröbner basis.

Definition 2.3.4 (Gröbner basis)**.** *A set G of polynomials is a Gröbner basis for an ideal I if the following criteria are satisfied:*

1. *I is generated by the elements of G, meaning $I = \langle G \rangle$*

2. *The leading terms of G generate the ideal of leading terms of the polynomials in I, meaning $\langle LT(G)\rangle = \langle LT(I)\rangle$.*

To obtain a basis that satisfies the definition of a Gröbner basis, it is possible to take any basis for an ideal in a polynomial ring over a field and use an algorithm developed by Bruno Buchberger to construct a basis with the necessary properties.

Algorithm 2.1 Buchberger's Algorithm

```
1: H: a finite set of polynomials
2: procedure BUCHBERGERS(H)
3:     while G ≠ H do
4:         H ← G
5:         for each pair (p, q), p ≠ q, for p, q ∈ G do
6:             S ← S(p, q)  ▷ eliminate the leading term of polynomials p and q
7:             r ← remainder of S after division by G
8:             if r ≠ 0 then
9:                 G := G ∪ {S}
10:            end if
11:        end for
12:    end while
13:    return G
14: end procedure
```

Theorem 2.3.5 (Buchberger's Criterion)**.** *A set $g = (g_1, g_2, ..., g_s)$ is a Gröbner basis if and only if for all pairs g_i and g_j where $i \neq j$ the remainder of the division of $S(g_i, g_j)$ by g equals zero.*

To illustrate Buchberger's algorithm, consider the following example:

Example 2.3.6. *Let the input basis $G = -4xy - 6y^2, x + 1$ with the input monomials $\{-4xy, 6y^2\}$ and $\{x, 1\}$ under lexicographic ordering.*

Then compute the s-polynomial of the basis elements $S(-4xy - 6y^2, x + 1) = (3y^2/2) - y$. Only one s-polynomial calculation is necessary because there are two basis elements.

After performing division by G, the remainder from division is $(3y^2/2) - y$.

So after the first iteration of Buchbergers, the enlarged basis is $G = \{-4xy - 6y^2, x + 1, (3y^2/2) - y\}$.

Since $g \neq h$, repeat the steps again.

Now, the s-polynomial must be calculated for each pair of monomials involving the new element. So, the s-polynomials of the new basis elements $S(-4xy - 6y^2, (3y^2/2) - y) = 0$ and $S(x + 1, (3y^2/2) - y) = 0$.

The remainder from dividing both by G is 0. Thus, the Gröbner basis is $G = \{-4xy - 6y^2, x + 1, (3y^2/2) - y\}$.

Corollary 2.3.7 (Existence of Gröbner Bases)**.** *Fix a monomial order. Then every nonzero ideal $I \subseteq k[x_1, x_2, ..., x_n]$ has a Gröbner basis. That is, there exist $g_1, g_2, ...g_t \in I$, such that $I = \langle g_1, g_2, ...g_t \rangle$ and $\langle LT(I) \rangle = \langle LT(g_1), LT(g_2), ...LT(g_t) \rangle$.*

The properties of Gröbner bases allow simple arithmetic solutions for solving systems of linear equations, which is what makes it applicable to a wide range of cryptographic schemes, including the the LWE problem. Using Gröbner bases yields an exponential speedup over other solutions to the LWE problem [26]. The basis generated by Buchberger's algorithm has the same roots as the input polynomial, but because of the way it is generated, is easier to solve.

2.4 Arora-Ge using Gröbner Bases

Recall the example from Section 2.2 where $s = 6$, $q = 17$, $a = (7, 10, 12, 9, 16, 12, 13, 5, 5, 10, 4, 0, 15, 16)$, $e = (0, -1, -1, -1, 1, 0, 1, -1, 0, -1, -1, 0, 0, 1)$, and $b = (8, 8, 3, 2, 12, 4, 11, 12, 13, 8, 6, 0, 5, 12)$. From that instantiation, the polynomial system, p, is

$$\begin{gathered}
p[0] = 14 \cdot u^3 + 3 \cdot u^2 + 6 \cdot u + 11, \\
p[1] = 3 \cdot u^3 + 3 \cdot u^2 + 11 \cdot u + 11, \\
p[2] = 6 \cdot u^3 + 4 \cdot u^2 + 11 \cdot u + 7, \\
p[3] = 2 \cdot u^3 + 10 \cdot u^2 + 3 \cdot u + 6, \\
p[4] = u^3 + 2 \cdot u^2 + 6 \cdot u + 16, \\
p[5] = 6 \cdot u^3 + 11 \cdot u^2 + 14 \cdot u + 9, \\
p[6] = 13 \cdot u^3 + u^2 + 3 \cdot u + 11, \\
p[7] = 11 \cdot u^3 + 16 \cdot u^2 + 4 \cdot u + 16, \\
p[8] = 11 \cdot u^3 + 6 \cdot u^2 + 3 \cdot u + 8, \\
p[9] = 3 \cdot u^3 + 3 \cdot u^2 + 11 \cdot u + 11, \\
p[10] = 4 \cdot u^3 + 16 \cdot u^2 + 14 \cdot u + 6, \\
p[11] = 8 \cdot u^3 + 9 \cdot u^2 + 12 \cdot u + 1, \\
p[12] = u^3 + 2 \cdot u^2 + 6 \cdot u + 16, \\
p[13] = 9 \cdot u^3 + 8 \cdot u^2 + 5 \cdot u + 16
\end{gathered}$$

To use Gröbner bases instead of linearization, calculate the ideal of the polynomials in p. Then perform Buchbergers algorithm in that ideal to obtain the Gröbner basis $u + 11$. The nullspace of the Gröbner basis must contains the secret, and since the secret can lives in $\mathbb{Z}/17\mathbb{Z}$, the only possible guess is 6.

Consider another instance of LWE where $n = 3$, $e \in \{-1, 0, 1\}$, prime $q = 1237$, and secret, $s = [178, 720, 595]$. Let a and e be sampled a uniformly random distribution such that

$$a = \begin{pmatrix} 573 & 1065 & 953 \\ 146 & 478 & 12 \\ 269 & 221 & 276 \\ 204 & 1226 & 1044 \\ 508 & 983 & 696 \end{pmatrix}, e = \begin{pmatrix} -1 \\ -1 \\ 1 \\ 1 \\ 0 \end{pmatrix}$$

Each entry of b is calculated by taking $a_i \cdot secret + e_i$. For this case:

$$b = \begin{pmatrix} 908 \\ 2 \\ 123 \\ 148 \\ 44 \end{pmatrix}$$

Then to construct the system of equations with the secret as part of the nullspace, take each $b_i - (a_{i0} \cdot s1 + a_{i1} \cdot s2 + a_{i2} \cdot s3 - e_j)$ for each row i and each possible error, e_j. For each i, take the product all polynomials for that row with each of the different e_j. From the above a, e and b, we obtain the following system of equations:

$$\begin{cases} 339 \cdot s1^3 + 718 \cdot s1^2 \cdot s2 + 648 \cdot s1 \cdot s2^2 + 667 \cdot s2^3 + 1128 \cdot s1^2 \cdot s3 + 644 \cdot s1 \cdot s2 \cdot s3 + 456 \cdot s2^2 \cdot s3 + 244 \cdot s1 \cdot s3^2 + 868 \cdot s2 \cdot s3^2 + 775 \cdot s3^3 + 1115 \cdot s1^2 + 369 \cdot s1 \cdot s2 + 1214 \cdot s2^2 + 149 \cdot s1 \cdot s3 + 672 \cdot s2 \cdot s3 + 900 \cdot s3^2 + 123 \cdot s1 + 397 \cdot s2 + 224 \cdot s3 + 1033 \\ \\ 156 \cdot s1^3 + 363 \cdot s1^2 \cdot s2 + 1019 \cdot s1 \cdot s2^2 + 615 \cdot s2^3 + 801 \cdot s1^2 \cdot s3 + 1195 \cdot s1 \cdot s2 \cdot s3 + 626 \cdot s2^2 \cdot s3 + 15 \cdot s1 \cdot s3^2 + 83 \cdot s2 \cdot s3^2 + 746 \cdot s3^3 + 485 \cdot s1^2 + 7 \cdot s1 \cdot s2 + 308 \cdot s2^2 + 1232 \cdot s1 \cdot s3 + 797 \cdot s2 \cdot s3 + 864 \cdot s3^2 + 868 \cdot s1 + 927 \cdot s2 + 1105 \cdot s3 + 6 \\ \\ 323 \cdot s1^3 + 465 \cdot s1^2 \cdot s2 + 1081 \cdot s1 \cdot s2^2 + 201 \cdot s2^3 + 424 \cdot s1^2 \cdot s3 + 338 \cdot s1 \cdot s2 \cdot s3 + 893 \cdot s2^2 \cdot s3 + 1157 \cdot s1 \cdot s3^2 + 785 \cdot s2 \cdot s3^2 + 713 \cdot s3^3 + 564 \cdot s1^2 + 683 \cdot s1 \cdot s2 + 476 \cdot s2^2 + 394 \cdot s1 \cdot s3 + 618 \cdot s2 \cdot s3 + 593 \cdot s3^2 + 356 \cdot s1 + 527 \cdot s2 + 563 \cdot s3 + 296 \\ \\ 1104 \cdot s1^3 + 258 \cdot s1^2 \cdot s2 + 168 \cdot s1 \cdot s2^2 + 94 \cdot s2^3 + 141 \cdot s1^2 \cdot s3 + 385 \cdot s1 \cdot s2 \cdot s3 + 787 \cdot s2^2 \cdot s3 + 285 \cdot s1 \cdot s3^2 + 876 \cdot s2 \cdot s3^2 + 850 \cdot s3^3 + 435 \cdot s1^2 + 135 \cdot s1 \cdot s2 + 533 \cdot s2^2 + 232 \cdot s1 \cdot s3 + 36 \cdot s2 \cdot s3 + 1103 \cdot s3^2 + 325 \cdot s1 + 413 \cdot s2 + 499 \cdot s3 + 704 \\ \\ 748 \cdot s1^3 + 115 \cdot s1^2 \cdot s2 + 561 \cdot s1 \cdot s2^2 + 525 \cdot s2^3 + 805 \cdot s1^2 \cdot s3 + 432 \cdot s1 \cdot s2 \cdot s3 + 1129 \cdot s2^2 \cdot s3 + 275 \cdot s1 \cdot s3^2 + 481 \cdot s2 \cdot s3^2 + 710 \cdot s3^3 + 1179 \cdot s1^2 + 58 \cdot s1 \cdot s2 + 604 \cdot s2^2 + 406 \cdot s1 \cdot s3 + 1034 \cdot s2 \cdot s3 + 1145 \cdot s3^2 + 289 \cdot s1 + 474 \cdot s2 + 844 \cdot s3 + 1024 \end{cases}$$

Though this example is nowhere near cryptographic size, finding the solution to this system of equations is still computationally complex. Using Gröbner bases can simplify the process. After generating an ideal for the system, and using functionality in SageMath [27] to create a Gröbner basis for that ideal, we obtain the basis {s1 + 1059, s2 + 517, s3 + 642}. The nullspace of that Gröbner basis should be the secret, and indeed it is. Thus, the correct secret s1 = 178, s2 = 720, s3 = 595 was recovered using Gröbner bases.

Thus, using Gröbner bases works as a replacement for the linearizating step in Arora-Ge. In fact, the work in [26] suggests that using Gröbner bases improves the complexity of the attack when $n \leq 5000$. Further research is needed to verify if the benefits of Gröbner bases beyond $n = 5000$.

3 Related Work

In [28] a new attack on PLWE is shown to be at least as hard as RLWE problems for power of 2 cyclotomic number fields, and for cyclotomic number fields in general with a small cost in terms of error growth.

Their work considers the ring of integers $\mathcal{O}_K$ in a number field K of degree n and a prime number q with the following properties:

1. q splits completely in K, and $q \nmid [R : \mathbb{Z}[\beta]]$
2. K is Galois over $\mathbb{Q}$
3. the ring of integers of K is generated over $\mathbb{Z}$ by β, $\mathcal{O}_K = \mathbb{Z}[\beta] = \mathbb{Z}[x]/(f(x))$ with $f'(\beta) \pmod q$ "small"
4. the transformation between the Minkowski embedding of K and the power basis representation of K is given by a scaled orthogonal matrix.
5. let $f \in \mathbb{Z}[x]$ be the minimal polynomial for β. Then $f(1) \equiv 0 \pmod q$
6. q can be chosen suitably large.

Specifically, they consider a family of polynomials $f(x) = X^n + (k-1)pX + p$, where p is a prime less than n, and k is chosen such that $1 + kp = q$ with q prime and $q > n$. The polynomial is constructed to be Eisenstein at p and, therefore, irreducible. They also note that there are infinitely many values of k that give a prime q by Dirichlet's theorem about primes in arithmetic progressions.

For particularly chosen error parameters, the authors in [28] solve non-dual decision-RLWE, a variant of LWE that has reductions to the search-LWE problem, given 20 samples, with a success rate ranging from 10% to 80%. Their attack requires that the defining polynomial f of P, as defined in Section 1, and the modulus q satisfy the relation $f(1) \equiv 0 \pmod q$.

Work in [18] extends the attack to search-LWE and generalizes so that $f(1) \equiv 0 \pmod q$ is not a necessary condition. The attack works modulo every modulus q', as long as the same error parameters are used (or smaller ones). Assuming that the highest $\lceil n/k \rceil$ error terms round to zero, the attack only requires k samples to recover the secret s using simple linear algebra with a 100% success rate.

4 Results

The results in [28] and [18] suggest that there is a family of polynomials that are susceptible to attack under both search and decision-LWE.

Theorem 4.0.1 (Irredicibility over infinite primes)**.** *An irreducible polynomial $f \in \mathbb{Q}[x]$ such that $deg(f) = l$ with prime l is irreducible mod q for an infinite set of primes q.*

Proof. Consider an irreducible polynomial $f \in \mathbb{Q}[x]$ such that $deg(f) = l$ with prime l. From Galois theory there exists a splitting field $K/\mathbb{Q}$ of f whose Galois group, $\mathrm{Gal}(K/\mathbb{Q})$, can be interpreted as permutations of the roots of f. Thus, $\mathrm{Gal}(K/\mathbb{Q}) \leq S_l$, where S_l is the symmetric group on l elements.

Since $deg(f) = l$, there is a sub-extension, $H = \mathbb{Q}(\alpha_1)$, of the splitting field K where a single root of f, namely α_1, is adjoined to the field. Field theory says this sub-extension H is degree $l = deg(f)$. Thus, the index of the subgroup H in $\mathrm{Gal}(K/\mathbb{Q})$ is l and $|H| \cdot l = \mathrm{Gal}(K/\mathbb{Q})$. Recall that l is a prime factor, which means that by Cauchy's Theorem that there is an element $\lambda \in \mathrm{Gal}(K/\mathbb{Q}) \leq S_l$ of order l. This means λ must be an l-cycle.

The Frobenius Density Theorem describes the relationship between cycle decomposition of elements of $\mathrm{Gal}(K/\mathbb{Q})$ and factorization of f modulo primes. To be more specific, let S be the set of primes q not dividing the discriminant of $f(x)$ for which $f(x)$ modulo q is irreducible. Then, the Frobenuius Density Theorem says S has natural density equal to $1/\#\mathrm{Gal}(K/\mathbb{Q})$ times the number of $\sigma \in G$ where σ is an indecomposible l-cycle. The work above shows that there is at least one l-cycle, λ, thus this the density is greater than zero. Since the set of all primes is infinite, a nonzero density of this family of primes guarantees that the polynomial is irreducible over an infinite set of primes. ■

As a consequence of Theorem 1.5.6, under each prime where a given polynomial f is irreducible, P_q is a field. Thus it is possible to use classical algorithms, such as the one discussed earlier, to compute Gröbner bases, for polynomial rings over P_q. So, PLWE instantiated with P_q is susceptible to the modified Arora-Ge with Gröbner Bases attacks.

4.1 Experiment

Consider the family of polynomials from above of the form $f(x) = X^n + (k-1)pX + p$ irreducible under some modulus q. The following experiments were performed using one example of a polynomial and prime pairing that fit the criteria outlined in the previous section.

Samples	Percentage correct	Time per run (sec)
4	100%	0.7083
3	100%	0.5473
2	94%	0.5673

Table 1: Polynomial $x^7 \cdot 60x + 5$ with modulus 769, $n = 6$, and errors of degree 1 with coefficients of 1. 100 trials were run for each number of samples.

Samples	Percentage correct	Time per run (sec)
4	100%	14.1642
3	100%	11.4584
2	94%	10.7105

Table 2: Polynomial $x^7 \cdot 60x + 5$ with modulus 769, $n = 6$, and errors of degree 2 with coefficients of 1. 100 trials were run for each number of samples.

Samples	Percentage correct	Time per run (sec)
4	100%	140.7105
3	90%	127.6386
2	90%	142.1697

Table 3: Polynomial $x^7 \cdot 60x + 5$ with modulus 769, $n = 6$, and errors of degree 1 with coefficients up to 2. 10 trials were run for each number of samples.

4.2 Discussion

In the above experiments, a high percentage of successfully recovering the secret key was obtained with a small number of samples. As the number of samples increased, so did the percentage of correct guesses. Generally, the time increased as the samples increased as well, except in the case of Table 3 where three samples showed an significantly lower time than both higher and lower number of samples.

5 Conclusion

5.1 Limitations and Future Research

The modified Arora-Ge attack implemented in SageMath utilized a toy implementation of Buchberger's algorithm [29] which may have negatively impacted the performance. The complexity of Buchbergers algorithm is known to be exponential in the worst case but there is extensive research on improvements that can be made in specific cases. [30] A future direction of reserach could include optimizing the coded attack in terms of Buchberger's algorithm or in other areas of the code.

The secret key sizes and error distribtutions considered in the experiments presented here are much smaller than practical cryptographic size. Though generally, larger keys are considered more secure, different recommendations exist in the literature. Work by Piekert and Lindner [31] suggests that a key size of around $n = 320$ to achieve "high security." Scaling the presented attack to cryptographic size for both secret size and error distribution would be another interesting avenue for future research.

5.2 Implications

This research demonstrates that the search-PLWE is tractable when instantiated by randomly sampling from a specific family of polynomials. Though this family is vulnerable to attack, this research by no means undermines the overall security of LWE. Other instantiations, specifically prime cyclotomics, have been shown to be invulnerable to attack. [32] For certain instantiations, the hardness guarantees discussed earlier hold, and no classical or quantum algorithm to break the encryption system has yet been found.

References

[1] J. F. Dooley, *History of Cryptography and Cryptanalysis: Codes, Ciphers, and Their Algorithms*. Springer, Aug. 23, 2018, 307 pp., Google-Books-ID: q61qDwAAQBAJ, ISBN: 978-3-319-90443-6.

[2] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," p. 15,

[3] Jeremy Fleming. "Director GCHQ's speech at national memorial arboretum centenary event." (), [Online]. Available: https://www.gchq.gov.uk/speech/nma-speech (visited on 03/16/2022).

[4] E. B. Barker and Q. H. Dang, "Recommendation for key management part 3: Application-specific key management guidance," National Institute of Standards and Technology, NIST SP 800-57Pt3r1, Jan. 2015, NIST SP 800–57Pt3r1. DOI: 10.6028/NIST.SP.800-57Pt3r1. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-57Pt3r1.pdf (visited on 03/08/2022).

[5] N. Koblitz, "Elliptic curve cryptosystems," p. 7,

[6] V. S. Miller, "Use of elliptic curves in cryptography," in *Advances in Cryptology — CRYPTO '85 Proceedings*, H. C. Williams, Ed., vol. 218, Series Title: Lecture Notes in Computer Science, Berlin, Heidelberg: Springer Berlin Heidelberg, 1986, pp. 417–426, ISBN: 978-3-540-16463-0. DOI: 10.1007/3-540-39799-X_31. [Online]. Available: http://link.springer.com/10.1007/3-540-39799-X_31 (visited on 03/16/2022).

[7] A. Abbas, S. Andersson, A. Asfaw, *et al.* "Learn quantum computation using qiskit." (2020), [Online]. Available: http://community.qiskit.org/textbook.

[8] A. E.
bibinitperiod T. Hosgood, *Introduction to Quantum Information Science*. [Online]. Available: https://qubit.guide/ (visited on 03/16/2022).

[9] "Shor's algorithm." (), [Online]. Available: https://qiskit.org/textbook/ch-algorithms/shor.html (visited on 03/07/2022).

[10] T. Review. "How quantum computer could break 2,048-bit RSA encryption in 8 hours." (), [Online]. Available: https://cacm.acm.org/news/237303-how-quantum-computer-could-break-2048-bit-rsa-encryption-in-8-hours/fulltext (visited on 03/11/2022).

[11] Jerry Chow, Oliver Dial, and Jay Gambetta. "IBM quantum breaks the 100-qubit processor barrier," IBM Research Blog. (), [Online]. Available: https://research.ibm.com/blog/127-qubit-quantum-processor-eagle (visited on 03/16/2022).

[12] U. Skosana and M. Tame, "Demonstration of shor's factoring algorithm for n $$=$$ = 21 on IBM quantum processors," *Scientific Reports*, vol. 11, no. 1, p. 16 599, Aug. 16, 2021, Number: 1 Publisher: Nature Publishing Group, ISSN: 2045-2322. DOI: 10.1038/s41598-021-95973-w. [Online]. Available: https://www.nature.com/articles/s41598-021-95973-w (visited on 03/16/2022).

[13] Oded Regev, "On lattices, learning with errors, random linear codes, and cryptography," *Journal of the ACM*, vol. 56, no. 6, pp. 1–40, Sep. 8, 2009. DOI: https://doi.org/10.1145/1568318.1568324. [Online]. Available: https://dl.acm.org/doi/abs/10.1145/1568318.1568324 (visited on 03/16/2022).

[14] V. Lyubashevsky, C. Peikert, and O. Regev, "On ideal lattices and learning with errors over rings," 230, 2012. [Online]. Available: http://eprint.iacr.org/2012/230 (visited on 03/16/2022).

[15] D. S. Dummit and R. M. Foote, *Abstract Algebra*. John Wiley & Sons, Jul. 14, 2003, 944 pp., Google-Books-ID: KIGbCgAAQBAJ, ISBN: 978-0-471-43334-7.

[16] R. Hines, "Groebner bases and applications," p. 7,

[17] D. Stehlé, R. Steinfeld, K. Tanaka, and K. Xagawa, "Efficient public key encryption based on ideal lattices," in *Advances in Cryptology – ASIACRYPT 2009*, M. Matsui, Ed., Berlin, Heidelberg: Springer, 2009, pp. 617–635, ISBN: 978-3-642-10366-7. DOI: 10.1007/978-3-642-10366-7_36.

[18] W. Castryck, I. Iliashenko, and F. Vercauteren, "Provably weak instances of ring-LWE revisited," 239, 2016. [Online]. Available: http://eprint.iacr.org/2016/239 (visited on 03/16/2022).

[19] D. A. Cox, J. Little, and D. O'Shea, "Gröbner bases," in *Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra*, ser. Undergraduate Texts in Mathematics, D. A. Cox, J. Little, and D. O'Shea, Eds., Cham: Springer International Publishing, 2015, pp. 49–119, ISBN: 978-3-319-16721-3. DOI: 10.1007/978-3-319-16721-3_2. [Online]. Available: https://doi.org/10.1007/978-3-319-16721-3_2 (visited on 03/07/2022).

[20] D. Khurana, "Lecture 10: Encryption from learning with errors," Lecture Notes, Lecture Notes, University of Illinois, Urbana Champaign, Oct. 5, 2019.

[21] Z. Brakerski, A. Langlois, C. Peikert, O. Regev, and D. Stehlé, "Classical hardness of learning with errors," *arXiv:1306.0281 [cs]*, Jun. 2, 2013. arXiv: 1306.0281. [Online]. Available: http://arxiv.org/abs/1306.0281 (visited on 03/07/2022).

[22] S. Arora and R. Ge, "New algorithms for learning in presence of errors," in *Automata, Languages and Programming*, L. Aceto, M. Henzinger, and J. Sgall, Eds., vol. 6755, Series Title: Lecture Notes in Computer Science, Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 403–415, ISBN: 978-3-642-22005-0 978-3-642-22006-7. DOI: 10.1007/978-3-642-22006-7_34. [Online]. Available: http://link.springer.com/10.1007/978-3-642-22006-7_34 (visited on 03/07/2022).

[23] V. Vaikuntanathan, "Lattices, learning with errors, post-quantum cryptography," UC Berkeley, 2020.

[24] B. Buchberger, "Bruno buchberger's PhD thesis 1965: An algorithm for finding the basis elements of the residue class ring of a zero dimensional polynomial ideal," *Journal of Symbolic Computation*, vol. 41, no. 3, pp. 475–511, Mar. 2006, ISSN: 07477171. DOI: 10.1016/j.jsc.2005.09.007. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0747717105001483 (visited on 03/08/2022).

[25] D. Hilbert, "Ueber die Theorie der algebraischen Formen," *Mathematische Annalen*, vol. 36, no. 4, pp. 473–534, Dec. 1, 1890, ISSN: 1432-1807. DOI: 10.1007/BF01208503. [Online]. Available: https://doi.org/10.1007/BF01208503 (visited on 03/16/2022).

[26] M. Albrecht, C. Cid, J.-C. Faugère, R. Fitzpatrick, and L. Perret, "On the complexity of the arora-ge algorithm against LWE," presented at the SCC 2012 – Third international conference on Symbolic Computation and Cryptography, Jul. 11, 2012, p. 93. [Online]. Available: https://hal.inria.fr/hal-00776434 (visited on 03/16/2022).

[27] "SageMath mathematical software system," SageMath Mathematical Software System. (), [Online]. Available: https://www.sagemath.org/ (visited on 03/16/2022).

[28] Y. Elias, K. E. Lauter, E. Ozman, and K. E. Stange, "Provably weak instances of ring-LWE," *arXiv:1502.03708 [cs, math]*, Aug. 8, 2015. arXiv: 1502.03708. [Online]. Available: http://arxiv.org/abs/1502.03708 (visited on 03/16/2022).

[29] "Educational versions of groebner basis algorithms — sage 9.5 reference manual: Polynomials." (), [Online]. Available: https://doc.sagemath.org/html/en/reference/polynomial_rings/sage/rings/polynomial/toy_buchberger.html (visited on 04/17/2022).

[30] B. Buchberger, "Gröbner bases: A short introduction for systems theorists," in *Computer Aided Systems Theory — EUROCAST 2001*, R. Moreno-Díaz, B. Buchberger, and J. Luis Freire, Eds., red. by G. Goos, J. Hartmanis, and J. van Leeuwen, vol. 2178, Series Title: Lecture Notes in Computer Science, Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 1–19, ISBN: 978-3-540-42959-3 978-3-540-45654-4. DOI: 10.1007/3-540-45654-6_1. [Online]. Available: http://link.springer.com/10.1007/3-540-45654-6_1 (visited on 04/17/2022).

[31] R. Lindner and C. Peikert, "Better key sizes (and attacks) for LWE-based encryption," in *Topics in Cryptology – CT-RSA 2011*, A. Kiayias, Ed., vol. 6558, Series Title: Lecture Notes in Computer Science, Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 319–339, ISBN: 978-3-642-19073-5 978-3-642-19074-2. DOI: 10.1007/978-3-642-19074-2_21. [Online]. Available: http://link.springer.com/10.1007/978-3-642-19074-2_21 (visited on 04/17/2022).

[32] C. Peikert, "How (not) to instantiate ring-LWE," 351, 2016. [Online]. Available: http://eprint.iacr.org/2016/351 (visited on 04/17/2022).