Attacks on Integer-RLWE

Alessandro Budroni[1], Benjamin Chetioui[1], and Ermes Franch[1]

Department of Informatics, University of Bergen, Norway[1]
{alessandro.budroni,benjamin.chetioui,ermes.franch}@uib.no

Abstract. In 2019, Gu Chunsheng introduced Integer-RLWE, a variant of RLWE devoid of some of its efficiency flaws. Most notably, he proposes a setting where n can be an arbitrary positive integer, contrarily to the typical construction $n = 2^k$. In this paper, we analyze the new problem and implement the classical meet-in-the-middle and lattice-based attacks. We then use the peculiarity of the construction of n to build an improved lattice-based attack in cases where n is composite with an odd divisor. For example, for parameters $n = 2000$ and $q = 2^{33}$, we reduce the estimated complexity of the attack from 2^{288} to 2^{164}. We also present reproducible experiments confirming our theoretical results.

Keywords: Post-quantum cryptography · Meet-in-the-middle · Lattice-based attack · I-RLWE

1 Introduction

With the advent of quantum computers, cryptographers have begun a consistent search for new trapdoor functions to use as building blocks for public-key cryptographic protocols that are resistant to quantum attacks.

In 2006, Regev introduced the Learning With Errors (LWE) [18] problem, one of the most important candidate trapdoors in post-quantum cryptography today. This problem has gained the trust of researchers thanks to its simplicity and its connection to lattice theory, which has been studied for years and provides us with useful security estimates. However, cryptosystems based on LWE present the disadvantage of having large public key sizes. In order to overcome this problem, Lyubashevsky, Peikert and Regev introduced Ring-LWE (RLWE) in 2010 [17], a related problem that allows smaller key sizes and more efficient encryption and decryption.

Informally, let $R = \mathbb{Q}[x]/(x^n + 1)$ and let $R_q = R/qR$, for an integer $n > 1$ and a prime q. The *Search* RLWE problem consists in finding the secret $\mathbf{s} \in R_q$ given samples of the form $(\mathbf{a}, \mathbf{b} = \mathbf{as} + \mathbf{e}) \in R_q \times R_q$, where $\mathbf{e} \in R_q$ is a "small" polynomial drawn from a certain distribution. Another variant of the problem is the *Decision* RLWE, which consists in distinguishing the pairs $(\mathbf{a}, \mathbf{b} = \mathbf{as} + \mathbf{e}) \in R_q \times R_q$ from pairs drawn uniformly at random from $R_q \times R_q$.

However, efficiency varies over different polynomial rings in RLWE and a dedicated optimization is required for each one of them. To overcome this inconvenience, Gu Chunsheng introduced a variant of RLWE named Integer-RLWE

(I-RLWE) [12]. In this new problem the variable x in RLWE is substituted with a prime q and the space of keys R_q is substituted with $\mathbb{Z}_p$, i.e. the set of integers modulo $p = q^n + 1$. The samples are of the form $(a, b = as + e) \in \mathbb{Z}_p \times \mathbb{Z}_p$, where $s = \sum_{i=0}^{n-1} s_i q^i$ and $e = \sum_{i=0}^{n-1} e_i q^i$ such that s_i and e_i are "small".

In his work, Gu also presented a public-key encryption protocol based on I-RLWE. It is therefore important to analyze this problem and gain a better understanding of the security it offers.

It is worth mentioning that a similar work has been done by Aggarwal et al. [1], who introduced an integer-version of the NTRU protocol, and by Beunardeau et al. [6] and de Boer et al. [7], who cryptanalyzed it. Moreover, a module version of I-RLWE is used in ThreeBears [13], a candidate protocol in the NIST Post-Quantum Standardization Process.

1.1 Contribution

In this paper, we analyze the complexity of the I-RLWE problem.

We provide some background and notation in Section 2. In Section 3 we adapt two standard attacks to this problem, namely a meet-in-the-middle attack and a lattice-based attack. These two attacks are straightforward to adapt to the problem, thus providing an upper bound for the acceptable complexity of further attacks with minimal effort; studying these attacks is a natural choice. We adapt the meet-in-the-middle attack of Cheon et al. on Decision LWE [10] to Search I-RLWE, and analyze its complexity. Likewise, we produce a lattice-based attack and follow the analysis of Alkim et al. [4] to determine its complexity.

In his work [12], Gu introduces a setting in which $q = 2^t$, instead of a prime, and n can be any positive integer, instead of $n = 2^k$. We exploit this setting to construct a new lattice-based attack for cases where n is neither prime nor a power of two and q is an arbitrary positive integer. Together with the outline of the attack, we show in Section 4 how these weak choices of n lead to a drastic drop in the estimated security of I-RLWE. Furthermore, we provide experiments supporting our theoretical estimates in Section 5. Finally we give our conclusions in Section 6.

2 Preliminaries and Notation

We denote the set of the real, rational and integer numbers with $\mathbb{R}, \mathbb{Q}, \mathbb{Z}$ respectively. Bold lower case letters represent vectors. For a given vector $\mathbf{v}$, v_j represents its j-th component. For a positive integer p, we write $\mathbb{Z}_p = \mathbb{Z}/p\mathbb{Z}$. Furthermore, the notation $[a]_p \in \{0, ..., p-1\}$ indicates $a \bmod p$ and, similarly, $[\mathbf{v}]_p$ is the vector composed by the entries of the integer vector $\mathbf{v}$ reduced modulo p. The notation $\|\mathbf{v}\|$ denotes the Euclidean norm of $\mathbf{v}$. Matrices are denoted with upper case bold $\mathbf{M}$.

Let q be an odd prime and let $p = q^n + 1$, for $n > 1$ integer. Given $a \in \mathbb{Z}_p \backslash \{p-1\}$, let a' be the integer representative of a in $\{0, ..., p-2\}$. We denote with $\mathbf{a} = (a_0, a_1, ..., a_{n-1})$ the vector of its components in base q. i.e. $a' = \sum_{i=0}^{n-1} a_i q^i$.

Similarly, if we represent $a \neq \frac{p}{2}$ with the integer $a' \in \left\{-\frac{p}{2}+1, ..., \frac{p}{2}-1\right\}$, then we can uniquely write $a' = \sum_{i=0}^{n-1} a_i q^i$, with $a_i \in \left\{-\frac{q-1}{2}, ..., \frac{q-1}{2}\right\}$. Hence we will write $\mathbf{a} = (a_0, a_1, ..., a_{n-1}) \in \left\{-\frac{q-1}{2}, ..., \frac{q-1}{2}\right\}^n$.

We use the symbol $\approx_B$ to denote the reflexive and symmetric relation between two vectors $\mathbf{x} \approx_B \mathbf{y}$ iff $\|\mathbf{x}-\mathbf{y}\|_\infty \leq B$ for some positive integer $B < \frac{q}{2}$. In a natural way we can extend this relation to $x, y \in \mathbb{Z}_p$ applying the relation above to the vectors of the corresponding components in base q.

2.1 Discrete Gaussian Distributions

In the following we write $x \sim D$ to mean that the random variable x follows the distribution D. Let $\rho_{0,\sigma}(x)$ be the probability distribution function of the Gaussian distribution $N(0,\sigma)$ with mean 0 and variance σ^2. We denote with $D_{\mathbb{Z},\sigma}$ the discrete Gaussian distribution on $\mathbb{Z}$ with mean 0 and variance σ^2 that assigns to each $a \in \mathbb{Z}$ the probability

$$\frac{\rho_{0,\sigma}(a)}{\sum_{d\in\mathbb{Z}} \rho_{0,\sigma}(d)} = \frac{\exp(-\pi a^2/2\sigma^2)}{\sum_{d\in\mathbb{Z}} \exp(-\pi d^2/2\sigma^2)}.$$

Given n independent random variables $x_1, ..., x_n \sim D_{\mathbb{Z},\sigma}$, we assume $y = \sum_{i=1}^n x_i$ follows the distribution $D_{\mathbb{Z},\sigma\sqrt{n}}$. This is a common assumption in this field and it comes from the approximation of the discrete Gaussian distribution with the continuous one. With the notation $\mathbf{v} \leftarrow D_{\mathbb{Z}^n,\sigma}$ we indicate a vector in $\mathbb{Z}^n$ with entries sampled independently at random from $D_{\mathbb{Z},\sigma}$.

Furthermore, we denote with $U_{\mathbb{Z}_q}$ the uniform distribution over $\mathbb{Z}_q$ and, similarly, $\mathbf{v} \leftarrow U_{\mathbb{Z}_q^n}$ is a vector in $\mathbb{Z}_q^n$ with entries sampled independently and uniformly at random from $\mathbb{Z}_q$.

2.2 Lattices

In this subsection we recall some important definitions and notions of lattice theory. For a more detailed resource on this topic, we refer the reader to [15].

A **lattice** is a discrete additive subgroup of $\mathbb{R}^n$. Let $\mathbf{b}_1, ..., \mathbf{b}_m \in \mathbb{R}^n$ be a set of linearly independent vectors. We define the lattice generated by $\mathbf{b}_1, ..., \mathbf{b}_m$ as

$$\mathcal{L}(\mathbf{b}_1, ..., \mathbf{b}_m) = \left\{ \mathbf{v} \in \mathbb{R}^n : \mathbf{v} = \sum_{i=1}^m \alpha_i \mathbf{b}_i,\ \alpha_i \in \mathbb{Z} \right\}.$$

A *basis* is any set of linearly independent vectors that generates the lattice as a $\mathbb{Z}$-module and the *dimension* is the number of vectors in a basis. Let $\mathbf{B}$ a matrix whose rows form a basis of $\mathcal{L}$, we then define the volume of $\mathcal{L}$ as $\mathrm{Vol}(\mathcal{L}) = \sqrt{\det(\mathbf{B}^T\mathbf{B})}$. Unless differently specified, we consider *full-rank* lattices through this paper — that is, the case when $m = n$.

Definition 1. *Let $\mathbf{b}_1, ..., \mathbf{b}_n \in \mathbb{R}^n$ be a set of linearly independent vectors. We denote with $\mathbf{b}_1^*, ..., \mathbf{b}_n^*$ the **Gram-Schmidt Orthogonalization** of $\mathbf{b}_1, ..., \mathbf{b}_n$ defined as follows:*

$$\mathbf{b}_1^* = \mathbf{b}_1, \qquad \mathbf{b}_i^* = \mathbf{b}_i - \sum_{j=1}^{i-1} \frac{\langle \mathbf{b}_i, \mathbf{b}_j^* \rangle}{\|\mathbf{b}_j^*\|^2} \mathbf{b}_j^*, \quad \text{for } 1 < i \leq n.$$

Definition 2. *Given a basis of a lattice $\mathcal{L}$ and a gap factor $\alpha \geq 1$, the **unique Shortest Vector Problem** (uSVP$_\alpha$) is to find (if it exists) the unique non-zero $\mathbf{v} \in \mathcal{L}$ such that any $\mathbf{u} \in \mathcal{L}$ with $\|\mathbf{u}\| \leq \alpha \|\mathbf{v}\|$ is an integral multiple of $\mathbf{v}$.*

Estimating the complexity to solve uSVP is a central problem in lattice-based cryptography. The following, known as *Gaussian Heuristic*, gives us an estimate of the length of the shortest vector in a random lattice.

Heuristic 1. *Let $\mathcal{L}$ be a full-rank lattice of dimension n and let $\mathbf{v} \in \mathcal{L}$ be a shortest non-zero vector. Then*

$$\|\mathbf{v}\| \approx \sqrt{\frac{n}{2\pi e}} \cdot \mathrm{Vol}(\mathcal{L})^{1/n}.$$

2.3 Integer Ring-Learning With Errors

Let q, n be two positive integers such that q is prime and $q > n^3$, and let $p = q^n + 1$.

Definition 3. *Let $\mathbf{s} \leftarrow D_{\mathbb{Z}^n,\sigma}$ be secret. Given an arbitrary number of samples of the form*

$$(a, b = as + e \bmod p) \in \mathbb{Z}_p \times \mathbb{Z}_p, \tag{1}$$

*where $a \leftarrow U_{\mathbb{Z}_p}$ and $\mathbf{e} \leftarrow D_{\mathbb{Z}^n,\sigma}$, the **Search Integer-RLWE** problem is to retrieve the secret s.*

Definition 4. *Let $\mathbf{s} \leftarrow D_{\mathbb{Z}^n,\sigma}$ be secret. The **Decision Integer-RLWE** problem is to distinguish with non-negligible advantage between an arbitrary number of samples of the form*

$$(a, b = as + e \bmod p) \in \mathbb{Z}_p \times \mathbb{Z}_p, \tag{2}$$

where $a \leftarrow U_{\mathbb{Z}_p}$ and $\mathbf{e} \leftarrow D_{\mathbb{Z}^n,\sigma}$, and the same number of samples drawn uniformly at random from $\mathbb{Z}_p \times \mathbb{Z}_p$.

In Section 3, we will consider n to be a power of 2 and $\sigma = \sqrt{n}$, as suggested by Gu [12] in the original definition. However, we will exploit a relaxation on n claimed in Remark 4.1 of [12] to build a more efficient attack in Section 4. Furthermore, the notation I-RLWE will refer to Search I-RLWE, which is the version of the problem that we address.

3 Standard Attacks

3.1 Meet-in-the-Middle attack

A classical meet-in-the-middle (MITM) attack on LWE was previously described [10]. Due to the connection I-RLWE has to the aforementioned problem, we follow the exact same methodology to perform our attack. We also draw inspiration from the work of de Boer et al. on the AJPS Mersenne-Based Cryptosystem [7].

Consider an I-RLWE sample $(a, b = as + e \bmod p)$. Let $v = s \bmod q^{n/2}$ and $w = s - v$. For the MITM approach, we consider the noisy relation

$$aw \approx_B b - av$$

We start by building a table

$$\mathcal{T} = \left\{ (av, v) \ : \ \mathbf{v} = (\mathbf{x}, \mathbf{y}), \mathbf{x} \in \{-B, \dots, B\}^{n/2}, \mathbf{y} \in \{0\}^{n/2} \right\} \subset \mathbb{Z}_p \times \mathbb{Z}_p$$

where B parameterizes the probability of finding the right secret depending on n. The probability that a given component of $\mathbf{s}$ falls in the range $\{-B, \dots, B\}$ is given by $P_B = \mathbb{P}(x \in \{-B, \dots, B\} \ : \ x \sim N(0, \sigma))$. It follows that the probability of all the components of $\mathbf{s}$ and $\mathbf{e}$ to fall in the range $\{-B, \dots, B\}$ is P_B^{2n}.

The second part of the MITM attack consists in an exhaustive search for y such that $\mathbf{y} \in \left\{ (\mathbf{y}_1, \mathbf{y}_2) : \ \mathbf{y}_1 \in \{0\}^{n/2}, \mathbf{y}_2 \in \{-B, \dots, B\}^{n/2} \right\}$, and $b - ay \in \mathbb{Z}_p$ is close to the first component of values in $\mathcal{T}$. If such a case occurs for a given y and a given key-value pair $(az, z) \in \mathcal{T}$, then we set $s' = z + y$, and we compute $e' = [b - as']_p$. Finally, if we have $e' \approx_B 0$, then s' is a likely candidate for s.

The difficult component of this attack lies in determining an efficient search algorithm to find an element in $\mathcal{T}$ that is close to $[b - ay]_p$, as is the case for the same attack on LWE.

We achieve this by applying the Noisy Collision Search described by Cheon et al. [10], with some slight adjustments to fit our problem. As such, the below description is directly adapted from their approach.

Noisy Collision Search In order to efficiently split the search space, Cheon et al. propose a locality sensitive hashing function sgn : $\mathbb{Z}_q \to \{0, 1\}$ defined as $\mathrm{sgn}(x) = 1$ for $x \in \{0, \dots, \frac{q}{2} - 1\}$ and 0 otherwise. For $y \in \mathbb{Z}_p$, if there exists $t \in \mathbb{Z}_p$ such that $y \approx_B t$, then it is guaranteed that $\mathrm{sgn}(y_i) = \mathrm{sgn}(t_i)$ if $y_i \in V_B = \{-\frac{q-1}{2} + B, \dots, -B - 1\} \cup \{B, \dots, \frac{q-1}{2} - B\}$ at a given index i.

To deal with the case when $y_i \notin V_B$, Cheon et al. define a function sgn' : $\mathbb{Z}_q \to \{0, 1, \times\}$ that returns $\mathrm{sgn}(y)$ if $y \in V_B$, and $\times$ otherwise. $\times$ indicates that the result may be either a 1 or a 0. It thus follows naturally that for any given $y \in \mathbb{Z}_p$, for any $t \in \mathbb{Z}_p$ such that $y \approx_B t$, $\mathrm{sgn}(y_i) = \mathrm{sgn}(t_i)$ for all $i \in \{i \mid y_i \in V_B\}$.

Meet-in-the-Middle Algorithm Our proposal makes use of two sub-algorithms described in the work of Cheon et al., namely Preprocess and Search [10]. We note that in our case, $m = n$ and otherwise perform slight adjustments so as to fit them to the Search version of our problem. The two algorithms detailed below are thus nearly taken verbatim from the aforementioned paper, where the only changes pertain to the content of $\mathcal{T}$ and $\mathcal{H}$ as well as the accumulation of the results of Search in a list L. We define sgn($\mathbf{x}$) (respectively sgn'($\mathbf{x}$)) to denote the application of sgn (respectively sgn') to each of the components of $\mathbf{x}$.

- Preprocess: On input $\mathcal{T} \subset \mathbb{Z}_p \times \mathbb{Z}_p$
  1. Initialize an empty hash table $\mathcal{H}$ with 2^n (empty) linked lists with indexes in $\{0,1\}^n$.
  2. For each $(t, z) \in \mathcal{T}$,
     (a) append (t, z) into the linked list indexed sgn($\mathbf{t}$).
  3. Return non-empty linked lists $\mathcal{H}$.

- Search: On input a hash table $\mathcal{H}$, a query $y \in \{x \mid \mathbf{x} \in \mathbb{Z}_q^n\}$ and a distance bound B,
  1. Initialize an empty list L.
  2. For each bin $\in \{0,1\}^n$ obtained from sgn'($\mathbf{y}$) by replacing × by 0 or 1,
     (a) If $\mathcal{H}$ has a linked list indexed bin, for each (t, z) in the list,
        i. Check whether $\|\mathbf{y} - \mathbf{t}\|_\infty \leq B$. If so, append $z + y$ to L.
  3. Return L.

Since our changes do not modify the core of the algorithms, we rely on the proof of correctness provided for the original algorithms.

In the same way, we need to adapt the MITM algorithm provided by Cheon et al. Pseudocode for this is given by Algorithm 1.

Algorithm 1: Meet-in-the-middle attack for Search I-RLWE

```
Input: A sample $(a, b) \in \mathbb{Z}_p \times \mathbb{Z}_p$
       $(n,\ q)$ such that $p = q^n + 1$
       $B \in \mathbb{Z}_q$
Output: A list $R$ of candidates for $s$
Initialize an empty list $R$
Compute $\mathcal{T} = \{(av, v)\ :\ \mathbf{v} = (\mathbf{x}, \mathbf{y}), \mathbf{x} \in \{-B, \dots, B\}^{n/2}, \mathbf{y} \in \{0\}^{n/2}\}$
Run Preprocess on input $\mathcal{T}$ to have a hash table $\mathcal{H}$
for $y \in \{x \mid \mathbf{x} \in \{-B, \dots, B\}^{n/2}\}$ do
    Concatenate the result of Search on input $(\mathcal{H}, b - ay, B)$ to $R$
return $R$
```

Since both $\mathbf{e}$ and $\mathbf{s}$ are sampled from the same distribution, we use the same B for the construction of $\mathcal{T}$ and for the Search step of the attack. We study the general complexity of the algorithm below.

Complexity Analysis According to the construction of $\mathcal{T}$, we write $N_{\mathcal{T}} = |\mathcal{T}| = (2B+1)^{n/2}$. We assume that the insertion of an element into a linked list has complexity $O(1)$. Now, for each element in $\mathcal{T}$, Preprocess needs to call sgn n times. It follows that the time cost of Preprocess is $N_{\mathcal{T}} \cdot n$.

Since the core of the algorithm didn't change, we rely on the proof of Lemma 3 in the work of Cheon et al. [10], which determines that Search performs around $2^{4nB/q}$ lookups in $\mathcal{T}$. Each one of these lookups returns a list of elements. We are interested in counting the average number of elements contained in one of the linked lists of $\mathcal{H}$.

Proposition 1. *Suppose that for $(t, z) \in \mathcal{T}$, t comes from a uniform distribution over $\mathbb{Z}_q^n$. Then, the average length of a given linked list in $\mathcal{H}$ is $\frac{N_{\mathcal{T}}}{2^n}$.*

Search thus finds $O(2^{4nB/q} \cdot \frac{N_{\mathcal{T}}}{2^n})$ elements. Finally, it must compute $\| \cdot \|_\infty$ for each of them, which has $O(n)$ cost.

We summarize these results in Table 1.

Table 1. Time cost for noisy search

Preprocess	Search (per query)
$N_{\mathcal{T}} \cdot n$	$O(2^{4nB/q} \cdot \frac{N_{\mathcal{T}}}{2^n} \cdot n)$

The full MITM algorithm also consists of two phases. We denote by T_{pre} the time complexity of the whole preprocessing phase (i.e. the building of $\mathcal{T}$ and the call to Preprocess), and by T_{search} the time complexity of the whole search phase, and give a cost estimation for them below:

- T_{pre} consists of roughly $N_{\mathcal{T}} \cdot \frac{n}{2}$ operations to build $\mathcal{T}$, added to the cost of executing Preprocess, thus $T_{\text{pre}} = N_{\mathcal{T}} \cdot (n + \frac{n}{2})$;
- T_{search} consists of $N_{\mathcal{T}}$ queries to Search, thus $T_{\text{search}} = O(N_{\mathcal{T}} \cdot 2^{4nB/q} \cdot \frac{N_{\mathcal{T}}}{2^n} \cdot n)$.

Choice of B The choice of B affects both the probability of success and the complexity of the MITM algorithm, where a higher accuracy necessarily means a higher complexity. We can use the empirical rule of the normal distribution to determine a good value for B. Take for example $n = 256$; according to the construction of I-RLWE, we have $\sigma = \sqrt{n} = 16$. The empirical rule cited above states that, if we set $B = 3\sigma$, $P_B = \mathbb{P}(x \in \{-B, \ldots, B\} \, : \, x \sim N(0, \sigma)) \approx 0.9973$.

In that setting, the probability that $\|(\mathbf{s}, \mathbf{e})\|_\infty \leq B$ (i.e. that the algorithm succeeds) is about $0.9973^{512} \approx 0.25$. On the other hand, if we set $B = 4\sigma$, then the algorithm will find the right secret with probability about $0.9999^{512} \approx 0.95$.

3.2 Lattice-Based Attack

Generally speaking, the most successful approach to solve LWE consists of converting this problem into a hard lattice problem (e.g. uSVP) and then applying a lattice reduction algorithm that solves it [3]. This approach also provides us with estimates of the security of LWE against lattice attacks based on the complexity of such reduction algorithms. Because of its similarity and connections to LWE, it is natural to define a lattice-based attack to solve I-RLWE.

Consider an I-RLWE sample $(a, b = as + e \bmod p)$. One wants to define a lattice that, given a small enough standard deviation σ, contains the *target* vector $\mathbf{v} = (\mathbf{s}, \mathbf{e}, 1)$ as a shortest vector. Next, one applies a reduction algorithm on a basis of such a lattice in order to find $\mathbf{v}$.

Consider the following lattice:

$$\mathcal{L} = \left\{ (\mathbf{x}, \mathbf{y}, u) \in \mathbb{Z}^n \times \mathbb{Z}^n \times \mathbb{Z} : a \sum_{i=0}^{n-1} x_i q^i + \sum_{j=0}^{n-1} y_j q^j - ub \equiv 0 \bmod p \right\}. \tag{3}$$

By definition, we have that $\mathbf{v} \in \mathcal{L}$. Furthermore, its norm is expected to be $\|\mathbf{v}\| \approx \sigma\sqrt{2n}$. Let us find a basis for $\mathcal{L}$. Define $\mathbf{w}^{(i)}$ as the vector formed by the components in base q of $-aq^i \bmod p$, for $i = 0, ..., n-1$. We indicate with $\mathbf{W}$ the $n \times n$ matrix whose i-th row is the $\mathbf{w}^{(i)}$ vector. We also define the matrix:

$$\mathbf{Q} = \begin{pmatrix} q & -1 & 0 & \dots & 0 & 0 \\ 0 & q & -1 & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & q & -1 & 0 \\ 0 & 0 & \dots & 0 & q & -1 \\ 1 & 0 & \dots & 0 & 0 & q \end{pmatrix} \in \mathbb{Z}^{n \times n}.$$

Based on the above, we define the following matrix:

$$\mathbf{B} = \left(\begin{array}{c|c|c} \mathbf{I}_n & \mathbf{W} & \begin{matrix} 0 \\ \vdots \\ 0 \end{matrix} \\ \hline \mathbf{0}_{n \times n} & \mathbf{Q} & \begin{matrix} 0 \\ \vdots \\ 0 \end{matrix} \\ \hline 0 \dots 0 & b_0 \dots b_{n-1} & 1 \end{array}\right) \in \mathbb{Z}^{(2n+1)\times(2n+1)}.$$

The rows of $\mathbf{B}$ form a basis for $\mathcal{L}$ and $\mathrm{Vol}(\mathcal{L}) = |\det(\mathbf{B})| = p$.

Success Condition and Complexity The best reduction algorithm known in practice is the Block-Korkine-Zolotarev (BKZ) algorithm [9]. This finds a reduced basis by calling an SVP oracle in a smaller dimension β a polynomial number of times [14].

By taking the analysis in [4] for the case of LWE as a model, we determine the success condition as follows. The Geometric Series Assumption [8, 3] states that a BKZ-reduced basis of a lattice $\mathcal{L}$ of dimension d is such that

$$\|\mathbf{b}_i^*\| = \delta_\beta^{d-2i-1} \cdot \mathrm{Vol}(\mathcal{L})^{1/d}, \quad \text{where } \delta_\beta = \left((\pi\beta)^{1/\beta} \cdot \frac{\beta}{2\pi e} \right)^{1/2(\beta-1)}.$$

Furthermore, the BKZ algorithm will detect the unique shortest vector of the lattice if its projection onto $\mathrm{Span}\{\mathbf{b}_{d-\beta+1}^*, ..., \mathbf{b}_d^*\}$ is shorter than the norm of $\mathbf{b}_{d-\beta}^*$. Let λ be the norm of the such projected vector. Then, the attack will succeed if

$$\lambda \leq \delta_\beta^{2\beta-d-1} \cdot \mathrm{Vol}(\mathcal{L})^{1/d}.$$

In our case, we have that $d = 2n+1$ and $\mathrm{Vol}(\mathcal{L}) = p \approx q^n$. The projection of our target vector has expected norm $\sigma\sqrt{\beta}$. So, in order to succeed with the attack, one must choose β to be such that

$$\sigma\sqrt{\beta} \leq \delta_\beta^{2(\beta-n-1)} q^{1/2}. \tag{4}$$

Since the complexity of BKZ is mostly ruled by the calls to the SVP oracle in dimension β, we only take the estimated complexity of this sub-routine into consideration. In the literature, there are two main branches for SVP oracle implementations: lattice sieving and lattice enumeration. Thanks to recent developments [5, 11, 16], lattice sieving took an asymptotic advantage over lattice enumeration. For this reason, we will consider only the estimated complexity provided by lattice sieving, that is $\approx 2^{0.292\beta}$.

As in the literature for LWE and RLWE, we use the above estimate to determine the theoretical security of I-RLWE for select parameters.

Remark 1. *From the complexity estimates given above, it follows that the lattice-based attack outlined here is more efficient than the meet-in-the-middle attack.*

4 Improved Lattice-Based Attack for Weak Choices of n

In Remark 4.1 of [12], Gu claims that n can be an arbitrary positive integer instead of being of the form 2^k when choosing q of the form 2^t instead of a prime. He justifies this different setting with more efficient encryption and decryption processes in his protocol. In this subsection we introduce a new lattice-based attack that exploits the fact that n is nor a prime, nor a power of 2.

Consider the following two lemmas.

Lemma 1. *Let $n \in \mathbb{Z}^+$ such that $n = \hat{n}k$ and let q be a positive integer. Then $q^n + 1 \equiv 0 \bmod q^{\hat{n}} + 1$ if and only if k is odd.*

Proof. Since $n = \hat{n}k$ we can rewrite $q^n + 1$ as $(q^{\hat{n}})^k + 1$ and $q^{\hat{n}} \equiv -1 \bmod q^{\hat{n}} + 1$. It follows that:

$$q^n + 1 \equiv (q^{\hat{n}})^k + 1 \equiv (-1)^k + 1 \equiv 0 \bmod q^{\hat{n}} + 1 \Leftrightarrow k \text{ is odd.}$$

□

Note. *We believe Lemma 1 is a known result in Number Theory. However, we could not find a reference for it.*

Lemma 2. *Take $n, \hat{n}$ and q as in Lemma 1, and define $p = q^n + 1$ and $\hat{p} = q^{\hat{n}} + 1$. Let $x \in \mathbb{Z}_p \setminus \{p-1\}$ and $\mathbf{x} = (x_0, ..., x_{n-1})$ be its representation in base q. Then we have that $\hat{x} = (x \bmod \hat{p}) \in \mathbb{Z}_{\hat{p}}$ has the following representation in base q:*

$$\hat{\mathbf{x}} = (\hat{x}_0, \hat{x}_1, ..., \hat{x}_{\hat{n}-1}),$$

where $\hat{x}_i = \sum_{j=0}^{n/\hat{n}-1} (-1)^j x_{j\hat{n}+i}$, for $i = 0, ..., \hat{n} - 1$.

Proof. Trivially, $q^{\hat{n}} \equiv -1 \bmod \hat{p}$. By applying this reduction to $x = x_0 + x_1 q + x_2 q^2 + ... + x_{n-1} q^{n-1}$ we get the above representation of $\hat{x}$. □

Let $\hat{n}$ be a divisor of n such that $n/\hat{n}$ is odd. Then $\hat{p} = q^{\hat{n}} + 1$ divides $p = q^n + 1$ (Lemma 2). Consider an I-RLWE sample $(a, b = as + e \bmod p)$ and let $\hat{a} = a \bmod \hat{p}$ and $\hat{b} = b \bmod \hat{p}$. Thanks to the Chinese Remainder Theorem, we have that

$$\hat{b} = \hat{a}\hat{s} + \hat{e} \bmod \hat{p},$$

where $\hat{s}$ (resp. $\hat{e}$) $= s$ (resp. e) mod$\hat{p}$. In other words, it is possible to obtain a new instance of the I-RLWE problem in a smaller dimension $\hat{n}$ such that, thanks to Lemma 2, we have that $\hat{\mathbf{s}}, \hat{\mathbf{e}} \sim D_{\mathbb{Z}^{\hat{n}}, \hat{\sigma}}$, where $\hat{\sigma} = \sigma\sqrt{n/\hat{n}}$.

The idea of this attack is to first solve the reduced problem using the lattice attack explained in Subsection 3.2, then use Lemma 2 to perform a faster lattice attack on the original problem.

Consider the following lattice:

$$\mathcal{L}_1 = \left\{ (\mathbf{x}, \mathbf{y}, u) \in \mathbb{Z}^{\hat{n}} \times \mathbb{Z}^{\hat{n}} \times \mathbb{Z} : \hat{a} \sum_{i=0}^{\hat{n}-1} x_i q^i + \sum_{j=0}^{\hat{n}-1} y_j q^j - u\hat{b} \equiv 0 \bmod \hat{p} \right\}. \quad (5)$$

Analogously to the lattice defined in Subsection 3.2, $\mathcal{L}_1$ contains the *reduced* target vector $\hat{\mathbf{v}} = (\hat{\mathbf{s}}, \hat{\mathbf{e}}, 1)$ and its volume is $\mathrm{Vol}(\mathcal{L}_1) = \hat{p} \approx q^{\hat{n}}$. One can apply a lattice reduction algorithm to find $\hat{\mathbf{v}}$ and so the reduced secret $\hat{s}$ and error $\hat{e}$. Next, we define the following lattice:

$$\mathcal{L}_2 = \left\{ (\mathbf{x}, \mathbf{y}, \mathbf{u}) \in \mathbb{Z}^n \times \mathbb{Z}^n \times \mathbb{Z}^3 : \begin{array}{c} x - u_1\hat{s} \equiv 0 \bmod \hat{p}, \\ y - u_2\hat{e} \equiv 0 \bmod \hat{p}, \\ a\sum_{i=0}^{n-1} x_i q^i + \sum_{j=0}^{n-1} y_j q^j - u_3 b \equiv 0 \bmod p \end{array} \right\}. \quad (6)$$

This lattice contains the target vector $\mathbf{v} = (\mathbf{s}, \mathbf{e}, \mathbf{1})$, where $\mathbf{1} = (1, 1, 1)$, and, as there are more conditions on its vectors, we expect it to have a higher volume compared to the lattice defined by (3).

Writing a basis for $\mathcal{L}_2$ varies according to the relations between $\mathrm{GCD}(b, p)$, $\mathrm{GCD}(\hat{s}, \hat{p})$ and $\mathrm{GCD}(\hat{e}, \hat{p})$ since some inversions modulo p and $\hat{p}$ are required. We show how to build a basis for the attacker's best case scenario, i.e. when $\mathrm{GCD}(b, p) = \mathrm{GCD}(\hat{s}, \hat{p}) = \mathrm{GCD}(\hat{e}, \hat{p}) = 1$. We do not report the other cases for conciseness.

Consider the following matrix:

$$\mathbf{B}_2 = \left(\begin{array}{c|c|ccc}
 & & u_1 & 0 & w_1 \\
\mathbf{I}_n & \mathbf{0}_{n\times n} & \vdots & \vdots & \\
 & & u_n & 0 & w_n \\ \hline
 & & 0 & v_1 & w_{n+1} \\
\mathbf{0}_{n\times n} & \mathbf{I}_n & \vdots & \vdots & \\
 & & 0 & v_n & w_{2n} \\ \hline
0\dots0 & 0\dots0 & \hat{p} & 0 & 0 \\
0\dots0 & 0\dots0 & 0 & \hat{p} & 0 \\
0\dots0 & 0\dots0 & 0 & 0 & p
\end{array}\right) \in \mathbb{Z}^{(2n+3)\times(2n+3)},$$

where

$$u_i = q^{i-1}\hat{s}^{-1} \bmod \hat{p} \qquad i = 1, ..., n,$$

$$v_i = q^{i-1}\hat{e}^{-1} \bmod \hat{p} \qquad i = 1, ..., n,$$

$$w_i = \begin{cases} aq^{i-1}b^{-1} \bmod p & \text{if} \quad i = 1, ..., n, \\ q^{i-1}b^{-1} \bmod p & \text{if} \quad i = n+1, ..., 2n. \end{cases}$$

It's easy to check that B_2 is a basis of $\mathcal{L}_2$. In general, $\mathrm{Vol}(\mathcal{L}_2)$ is upper bounded by $p\hat{p}^2 \approx q^{n+2\hat{n}}$. This bound is reached in the aforementioned case (but not only).

4.1 Analysis and Success Condition

In order for the attack to be successful, the reduced vector $\hat{\mathbf{v}}$ must be small enough to be a shortest vector of $\mathcal{L}_1$. Using the Gaussian Heuristic, we check if $\hat{\mathbf{v}}$ is shorter than the estimated shortest vector in $\mathcal{L}_1$:

$$\|\hat{\mathbf{v}}\| \approx \hat{\sigma}\sqrt{2\hat{n}+1} = \sigma\sqrt{\frac{n}{\hat{n}}}\sqrt{2\hat{n}+1} \leq \sqrt{\frac{2\hat{n}+1}{2\pi e}} \cdot q^{1/2}.$$

Then, one gets that σ must be such that:

$$\sigma \leq \sqrt{q\frac{\hat{n}}{2n\pi e}}. \tag{7}$$

In his paper, Gu suggested $\sigma = \sqrt{n}$ and $q > n^3$. In this setting, condition (7) is satisfied.

We give a success condition on the block size β_1 for the BKZ-β_1 reduction algorithm to find the target vector $\hat{\mathbf{v}}$ using an analogous approach as in Subsection 3.2:

$$\hat{\sigma}\sqrt{\beta_1} \leq \delta_{\beta_1}^{2(\beta_1-\hat{n}-1)} \cdot q^{1/2}.$$

Similarly, the target vector $\mathbf{v}$ will be found through a BKZ-β_2 reduction on a basis of $\mathcal{L}_2$ if the block size β_2 is such that

$$\sigma\sqrt{\beta_2} \leq \delta_{\beta_2}^{2(\beta_2-n-2)} \cdot q^{\frac{n+2\hat{n}}{2n+3}}.$$

In the above expression we took $\mathrm{Vol}(\mathcal{L}_2) = p\hat{p}^2 \approx q^{n+2\hat{n}}$.

In Table 2 we show the significant advantage of using this approach over the standard lattice attack described in Subsection 3.2 for some choices of n and $\hat{n}$. The complexity, based on the required cost for performing lattice sieving, drops significantly. This allows us to conclude that n must not have odd divisors, that is to say n is either a prime or a power of 2, in line with the setting of RLWE.

Table 2. Columns 1, 2 and 3 define the parameters, with $\sigma = \sqrt{n}$. Columns 4 and 7 contain the minimum block size (β and β_2) of the BKZ subroutine required to find the target vector $\mathbf{v}$ respectively from lattice (3) and (6). Column 6 contains the minimum block size β_1 to find $\hat{\mathbf{v}}$ from reducing a basis of lattice (5). The complexities in column 5 and 8 are expressed in $\log_2$ and correspond to the lattice sieving complexity with parameter respectively β and β_2

Parameters			Standard Lattice Attack		Improved Lattice Attack		
n	$\hat{n}$	q	β	Complexity	β_1	β_2	Complexity
2000	400	2^{33}	987	288	130	561	164
1500	300	2^{32}	713	208	83	396	116
1200	240	2^{31}	559	163	< 60	304	89
1000	200	2^{30}	463	135	< 60	246	71

Remark 2. *This attack can be further improved when n has more than one odd divisor by adding more conditions in the definition of $\mathcal{L}_2$.*

Remark 3. *We remark that these choices of n remain weak for any q and not only in the setting that Gu proposes.*

5 Experiments

In order to confirm our theoretical results, we performed some practical experiments which we report in this section.

First we generated some I-RLWE samples, then we used the BKZ implementation contained in the General Sieve Kernel [2], the cutting-edge implementation at the moment of writing, in order to perform the attacks. Finally we compared the minimum block size parameter β of the BKZ reduction required to successfully retrieve the secret and the error for both approaches. For each instance, we chose $\hat{n}$ among the possible choices so that the uSVP on $\mathcal{L}_1$ is solvable with LLL.

In the table below we report the results obtained during our experiments. The I-RLWE samples that we used in our experiments can be found at https://archive.org/details/irlwesamples.

Table 3. Columns 1, 2 and 3 define the parameters, with $\sigma = \sqrt{n}$. Column 3 report the minimum block size β that allowed us to retrieve the target vector $\mathbf{v}$ through BKZ reduction on the lattice defined in (3). Similarly, columns 4 and 5 report the minimum block sizes β_1 and β_2 for lattices (5) and (6) respectively, so that the attack was successful. Note that $\beta_1 = 1$ corresponds to LLL.

Parameters			Standard Lattice Attack	Improved Lattice Attack	
n	$\hat{n}$	q	β	β_1	β_2
130	26	2^{22}	41	1	2
110	22	2^{21}	28	1	2
105	15	2^{21}	9	1	2

6 Conclusion

In this work, we adapted a meet-in-the-middle attack and a lattice-based attack from LWE to I-RLWE. The latter, as in the case of LWE and RLWE, gives us theoretical estimates regarding the security provided by I-RLWE.

We introduced a new lattice-based attack against I-RLWE when the parameter n is chosen as a composite number divisible by an odd number. This attack exploits the weakness on choice of n to build a new lattice of bigger volume, leading to a more efficient secret and error recovery through lattice reduction. We provided theoretical estimates of our attack showing how the complexity of solving I-RLWE reduces in this setting. For example, for $n = 2000$ the complexity reduces from 2^{288}, estimated with the standard lattice attack, to 2^{164}. Moreover, this gap also appears for smaller n as in the case for $n = 1000$ where the complexity drops from 2^{135} to 2^{71}. This attack likely applies to RLWE; however, this was not investigated as the setting considered here is avoided in the literature of RLWE-based protocols.

To confirm our theoretical results, we run experiments for n up to 130. Our results shows that a much smaller block-size parameter β is required in the BKZ lattice reduction algorithm in order to successfully recover the secret and the error.

We conclude remarking that choices of n as in the aforementioned case must definitely be avoided in I-RLWE, as is prescribed for RLWE.

Acknowledgments We thank Martha Norberg Hovd and Andrea Tenti for proofreading our manuscript in an early stage and providing insightful comments.

References

1. Aggarwal, D., Joux, A., Prakash, A., Santha, M.: A new public-key cryptosystem via mersenne numbers. In: Shacham, H., Boldyreva, A. (eds.) Advances in Cryptology – CRYPTO 2018. pp. 459–482. Springer International Publishing, Cham (2018). https://doi.org/10.1007/978-3-319-96878-0_16
2. Albrecht, M., Ducas, L., Herold, G., Kirshanova, E., Postlethwaite, E., Stevens, M.: The general sieve kernel and new records in lattice reduction. pp. 717–746 (04 2019). https://doi.org/10.1007/978-3-030-17656-3_25
3. Albrecht, M., Göpfert, F., Virdia, F., Wunderer, T.: Revisiting the expected cost of solving usvp and applications to lwe. In: Advances in Cryptology - ASIACRYPT 2017. pp. 297–322. Lecture Notes in Computer Science, Springer (2017). https://doi.org/10.1007/978-3-319-70694-8_11
4. Alkim, E., Ducas, L., Pöppelmann, T., Schwabe, P.: Post-quantum key exchange: A new hope. In: Proceedings of the 25th USENIX Conference on Security Symposium. p. 327343. SEC16, USENIX Association, USA (2016). https://doi.org/10.5555/3241094.3241120
5. Becker, A., Ducas, L., Gama, N., Laarhoven, T.: New directions in nearest neighbor searching with applications to lattice sieving. In: Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms. p. 1024. SODA 16, Society for Industrial and Applied Mathematics, USA (2016). https://doi.org/10.5555/2884435.2884437
6. Beunardeau, M., Connolly, A., Géraud, R., Naccache, D.: On the hardness of the mersenne low hamming ratio assumption. In: Lange, T., Dunkelman, O. (eds.) Progress in Cryptology – LATINCRYPT 2017. pp. 166–174. Springer International Publishing, Cham (2019). https://doi.org/10.1007/978-3-030-25283-0_9
7. de Boer, K., Ducas, L., Jeffery, S., de Wolf, R.: Attacks on the ajps mersenne-based cryptosystem. In: Lange, T., Steinwandt, R. (eds.) Post-Quantum Cryptography. pp. 101–120. Springer International Publishing, Cham (2018). https://doi.org/10.1007/978-3-319-79063-3_5
8. Chen, Y.: Lattice reduction and concrete security of fully homomorphic encryption. PhD thesis, l'Universit Paris Diderot (2013), https://archive.org/details/PhDChen13

9. Chen, Y., Nguyen, P.Q.: BKZ 2.0: Better lattice security estimates. In: Lee, D.H., Wang, X. (eds.) Advances in Cryptology – ASIACRYPT 2011. pp. 1–20. Springer Berlin Heidelberg, Berlin, Heidelberg (2011). https://doi.org/10.1007/978-3-642-25385-0_1
10. Cheon, J.H., Hhan, M., Hong, S., Son, Y.: A hybrid of dual and meet-in-the-middle attack on sparse and ternary secret lwe. IEEE Access **7**, 89497–89506 (2019). https://doi.org/10.1109/ACCESS.2019.2925425
11. Ducas, L.: Shortest vector from lattice sieving: A few dimensions for free. In: Nielsen, J.B., Rijmen, V. (eds.) Advances in Cryptology – EUROCRYPT 2018. pp. 125–145. Springer International Publishing, Cham (2018). https://doi.org/10.1007/978-3-319-78381-9_5
12. Gu, C.: Integer version of ring-lwe and its applications. In: Meng, W., Furnell, S. (eds.) Security and Privacy in Social Networks and Big Data - 5th International Symposium, SocialSec 2019, Copenhagen, Denmark, July 14-17, 2019, Revised Selected Papers. Communications in Computer and Information Science, vol. 1095, pp. 110–122. Springer (2019). https://doi.org/10.1007/978-981-15-0758-8_9
13. Hamburg, M.: Threebears. Technical report, National Institute of Standards and Technology (2017), https://csrc.nist.gov/Projects/post-quantum-cryptography/round-2-submissions
14. Hanrot, G., Pujol, X., Stehl, D.: Terminating BKZ. Cryptology ePrint Archive, Report 2011/198 (2011), https://eprint.iacr.org/2011/198
15. Hoffstein, J., Pipher, J., Silverman, J.H.: An Introduction to Mathematical Cryptography. Springer Publishing Company, Incorporated, 2nd edn. (2014). https://doi.org/10.1007/978-1-4939-1711-2
16. Laarhoven, T., Mariano, A.: Progressive lattice sieving. In: Lange, T., Steinwandt, R. (eds.) Post-Quantum Cryptography. pp. 292–311. Springer International Publishing, Cham (2018). https://doi.org/10.1007/978-3-319-79063-3_14
17. Lyubashevsky, V., Peikert, C., Regev, O.: On ideal lattices and learning with errors over rings. In: Gilbert, H. (ed.) Advances in Cryptology – EUROCRYPT 2010. pp. 1–23. Springer Berlin Heidelberg, Berlin, Heidelberg (2010). https://doi.org/10.1145/2535925
18. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. In: Proceedings of the Thirty-seventh Annual ACM Symposium on Theory of Computing. pp. 84–93. STOC '05, ACM, New York, NY, USA (2005). https://doi.org/10.1145/1060590.1060603