Three's Compromised Too:
Circular Insecurity for Any Cycle Length from (Ring-)LWE

Navid Alamati[*] Chris Peikert[†]

June 3, 2016

Abstract

Informally, a public-key encryption scheme is *k-circular secure* if a cycle of k encrypted secret keys $(\mathsf{Enc}_{pk_1}(sk_2), \mathsf{Enc}_{pk_2}(sk_3), \ldots, \mathsf{Enc}_{pk_k}(sk_1))$ is indistinguishable from encryptions of zeros. Circular security has applications in a wide variety of settings, ranging from security of symbolic protocols to fully homomorphic encryption. A fundamental question is whether standard security notions like IND-CPA/CCA imply k-circular security.

For the case $k = 2$, several works over the past years have constructed counterexamples—i.e., schemes that are CPA or even CCA secure but not 2-circular secure—under a variety of well-studied assumptions (SXDH, decision linear, and LWE). However, for $k > 2$ the only known counterexamples are based on strong general-purpose obfuscation assumptions.

In this work we construct k-circular security counterexamples for any $k \geq 2$ based on (ring-)LWE. Specifically:

- for any constant $k = O(1)$, we construct a counterexample based on n-dimensional (plain) LWE for $\mathrm{poly}(n)$ approximation factors;
- for any $k = \mathrm{poly}(\lambda)$, we construct one based on degree-n ring-LWE for at most subexponential $\exp(n^\varepsilon)$ factors.

Moreover, both schemes are k'-circular insecure for $2 \leq k' \leq k$.

Notably, our ring-LWE construction does not immediately translate to an LWE-based one, because matrix multiplication is not commutative. To overcome this, we introduce a new "tensored" variant of LWE which provides the desired commutativity, and which we prove is actually equivalent to plain LWE.

[*] Computer Science and Engineering, University of Michigan. Email: `alamati@umich.edu`

[†] Computer Science and Engineering, University of Michigan. Email: `cpeikert@umich.edu`. This material is based upon work supported by the National Science Foundation under CAREER Award CCF-1054495 and CNS-1606362, and by the Alfred P. Sloan Foundation. The views expressed are those of the authors and do not necessarily reflect the official policy or position of the National Science Foundation or the Sloan Foundation.

1 Introduction

Classical security definitions for encryption, like semantic security [GM82], only consider messages that the *attacker itself* can generate. In certain contexts, however, a system must encrypt *secret keys*, which are unknown to the attacker, under corresponding public keys. Prominent examples of this include the anonymous credential scheme of Camenisch and Lysyanskaya [CL01], methods for proving the computational soundness of symbolic protocols [ABHS05], password managers and disk encryption utilities, and Gentry's "bootstrapping" technique for obtaining (unbounded) fully homomorphic encryption [Gen09b, Gen09a].

For these reasons, the notions of *circular* and, more generally, *key-dependent message (KDM)* security have attracted much attention in recent years. Informally, a public-key cryptosystem is k-circular secure if an *encryption cycle* $(\mathsf{Enc}_{pk_1}(sk_2), \mathsf{Enc}_{pk_2}(sk_3), \ldots, \mathsf{Enc}_{pk_k}(sk_1))$ is indistinguishable from encryptions of "junk" messages. KDM security considers a broader setting in which (adversarially specified) functions of the secret keys may be encrypted under any of the public keys.

Early positive results on circular/KDM security go back to Black *et al.* [BRS02] and [CL01], who proposed KDM-secure schemes in the random oracle model. Several years later, Boneh *et al.* [BHHO08] were the first to give a cryptosystem in the standard model with a proof of KDM-security (for affine functions) under a well-studied assumption, namely, Decision Diffie-Hellman (DDH). This was soon followed by constructions based on the learning with errors (LWE) [ACPS09] and quadratic residuosity [BG10] assumptions; constructions for richer notions like identity-based encryption [AP12]; and "KDM amplification" transforms that extended the class of functions far beyond affine ones [BHHI10, BGK11, MTY11, App11].

Despite all this progress, a very basic yet still unresolved question about circular/KDM security—especially in light of the fact that almost all the systems cited above are *specially designed* to obtain it—is:

Do classical security notions like IND-CPA or IND-CCA imply k-circular security?

For $k = 1$ there are trivial counterexamples, but for $k \geq 2$ the question is much more interesting, and has been studied extensively in recent years. To date there is a significant gap between what is known for the cases $k = 2$ and $k > 2$.

The case $k = 2$. In this setting there are several negative results based on well-studied assumptions. The first counterexamples were presented by Acar *et al.* [ABBC10] and Cash *et al.* [CGH12], who respectively gave schemes that are CPA secure but *not* 2-circular secure, and schemes that are CPA/CCA secure but not even *weakly* two-circular secure. (Weak circular security refers to the secrecy of other encrypted messages in the presence of an encryption cycle.) In both works, CPA/CCA security was under the SXDH assumption for groups with asymmetric bilinear pairings.

Most recently, Bishop *et al.* [BHW15] gave additional counterexamples for $k = 2$, based on the decision linear and LWE assumptions. In addition, they introduced the useful notion of a *cycle tester*, which simplifies and modularizes the construction of counterexamples. For example, they showed how to combine a k-cycle tester with any CPA/CCA-secure cryptosystem to obtain CPA/CCA-secure schemes that are not k-circular secure. (However, all their *concrete* cycle testers were for $k = 2$.)

The case $k > 2$. For larger values of k, the relationship between CPA/CCA and circular security remained open for many years. Intuitively, constructing a counterexample for this case is more difficult because encryption must set up a relation among k ciphertexts that can be efficiently detected; bilinear maps make this possible for $k = 2$, but seem less useful for $k > 2$. Indeed, the only negative results are two recent concurrent and independent works of Koppula *et al.* [KRW15] and Marcedone and Orlandi [MO14], which used strong *obfuscation* assumptions to construct, for any k, encryption schemes that are CPA secure but k-circular insecure. More specifically, the counterexample in [KRW15] is based on indistinguishability obfuscation (iO) for arbitrary circuits (e.g., the candidate construction proposed in [GGH^{+}13]), whereas [MO14] used the even stronger assumption of virtual black box (VBB) obfuscation for a certain large enough class of functions. (Later, following [KRW15], the authors of [MO14] refined their scheme to rely only on iO.) Separately, Koppula *et al.* also showed that any k-circular security counterexample can be generically transformed into one that is not even *weakly* circular secure, because an encryption cycle implicitly reveals all the secret keys.

In summary, for $k = 2$ we have circular-security counterexamples under a reasonably wide variety of well-studied assumptions, whereas for $k > 2$ the available evidence is weaker, since it is based on the more speculative assumption that secure iO exists. In particular, up to this point we do not have a candidate iO scheme with a proof of security under simple, plausible, and concrete assumptions. This stands in contrast to well-studied problems like those relating to bilinear pairings or (ring-)LWE, the latter of which are provably hard assuming the *worst-case* hardness of certain lattice problems [Reg05, Pei09, BLP^{+}13, LPR10].

1.1 Contributions

Our main contributions are k-circular security counterexamples, for *any* $k \geq 2$, based on the LWE [Reg05] and ring-LWE [LPR10] assumptions. We stress that these are the first circular security counterexamples for $k > 2$ that do not rely on general-purpose obfuscation assumptions. More specifically, we prove the following two main theorems (in what follows, λ denotes the security parameter):

Informal Theorem 1. *For any* $\mathrm{poly}(\lambda)$*-bounded* $k \geq 2$*, there exists (in the common random string model) a* k*-cycle tester based on* ring-LWE *in degree-*n *rings for* $\tilde{O}(nk)^{O(k)}$ *approximation factors. Moreover, it is also a* k'*-cycle tester for* $2 \leq k' \leq k$*.*

As example parameterizations, for any constant $k = O(1)$ we obtain a k-cycle tester based on $\mathrm{poly}(n)$ approximation factors, which are conjectured to offer $2^{\tilde{\Omega}(n)}$ hardness. For arbitrary $k = \mathrm{poly}(\lambda)$, we can obtain a k-cycle tester based on subexponential $2^{n^{\varepsilon}}$ factors for any desired constant $\varepsilon > 0$, by letting $n = \tilde{\Omega}(\lambda^{c/\varepsilon})$ be a sufficiently large polynomial in λ. For such factors, ring-LWE is conjectured to offer $2^{\tilde{\Omega}(n^{1-\varepsilon})} \geq 2^{\Omega(\lambda)}$ hardness.

Informal Theorem 2. *For any* constant $k \geq 2$*, there exists (in the common random string model) a* k*-cycle tester based on* plain LWE *in* n *dimensions for* $n^{O(k^2)}$ *approximation factors. Moreover, it is also a* k'*-cycle tester for* $2 \leq k' \leq k$*.*

We emphasize that unlike many lattice-based cryptographic schemes, the ring-LWE-based cycle tester from our first theorem does *not* appear to "mechanically" translate to plain LWE, so additional ideas are needed to prove our second theorem. In brief, this is because the ring-LWE problem is usually defined over a *commutative* ring, whereas in the plain LWE setting, the corresponding ring of n-by-n matrices is not commutative (see Section 1.2 below for further details). To overcome this obstacle, we introduce a new variant of LWE that we call *tensored LWE*, and prove that it is equivalent to plain LWE for corresponding parameters. We note, however, that this technique limits the solution to *constant* (but arbitrary) $k = O(1)$, because it induces key sizes that are exponential in k.

Finally, by combining our cycle testers with appropriate (ring-)LWE-based CPA/CCA-secure encryption schemes [Reg05, GPV08, MP12] using the generic transformations given in [KRW15, BHW15], we immediately obtain CPA/CCA-secure cryptosystems that are k-circular insecure, and (in the CPA-secure case) for which an encryption cycle even reveals all the encrypted secret keys.

Recent related work. In a concurrent and independent work, Koppula and Waters [KW16] also constructed a k-cycle tester for arbitrary (a priori bounded) k based on plain LWE; it can be easily adapted to ring-LWE using standard transformations. Like ours, their construction uses "telescoping products," but the exact way in which these are used to detect cycles differs significantly—in particular, their construction does not need secret keys to commute under multiplication (see Section 1.2 below for further details). This yields different simplicity and efficiency profiles for the schemes. Specifically, our *ring-LWE* scheme has public keys, secret keys, and ciphertexts that are all an $\Omega(n)$ factor smaller than in the ring-LWE version of their scheme, and is arguably technically simpler and more direct. However, their *plain-LWE* construction can handle any *polynomial* cycle length $k = \mathrm{poly}(\lambda)$, whereas our plain-LWE construction is restricted to any constant $k = O(1)$ due to an n^k factor in our key and ciphertext lengths, which arises from our "tensored" form of plain LWE that yields commuting secrets. In addition, their scheme does not use a common random string, whereas ours does.

1.2 Techniques

Here we give an overview of our constructions and proof techniques. To start, we give a brief exposition of the LWE-based two-cycle tester from [BHW15]. We recall that a k-cycle tester is a relaxed form of encryption scheme that does not require a decryption algorithm; it only requires an efficient algorithm that reliably detects when a k-tuple of ciphertexts forms an encryption cycle.

In the two-cycle tester from [BHW15], a secret key is the randomness used to generate a uniformly random matrix $\mathbf{S} \in \mathbb{Z}_q^{n\times m}$ along with a "trapdoor" $T_{\mathbf{S}}$, using the $\mathsf{GenTrap}$ algorithm from, e.g., [MP12]. The matrix $\mathbf{S}$ is interpreted as a matrix of *LWE secrets*, and the public key is the LWE instance $(\mathbf{A}, \mathbf{B} \approx \mathbf{S}^t\mathbf{A})$ for a uniformly random $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$.

To encrypt under a public key $(\mathbf{A}, \mathbf{B})$, we interpret the message as randomness for $\mathsf{GenTrap}$, thereby generating some $\hat{\mathbf{S}}$ with trapdoor $T_{\hat{\mathbf{S}}}$. We then choose a random short integer vector $\mathbf{r}$, let $\mathbf{v} = \mathbf{A}\mathbf{r}$, and output the two-component ciphertext

$$\big(\, \mathbf{x} \leftarrow \hat{\mathbf{S}}^{-1}[\mathbf{v}]\,,\, \mathbf{u} = \mathbf{B}\mathbf{r} \approx \mathbf{S}^t\mathbf{A}\mathbf{r} = \mathbf{S}^t\mathbf{v} \,\big) \in \mathbb{Z}^m \times \mathbb{Z}_q^m.$$

Here $\mathbf{x} \leftarrow \hat{\mathbf{S}}^{-1}[\mathbf{v}]$ denotes using the trapdoor $T_{\hat{\mathbf{S}}}$ to randomly sample a short solution to $\hat{\mathbf{S}}\mathbf{x} = \mathbf{v}$ without revealing any information about $T_{\hat{\mathbf{S}}}$, e.g., using a discrete Gaussian distribution [GPV08]. (This is used in the proof of IND-CPA security.) Notice that $\mathbf{x}$ is a short integer vector, whereas $\mathbf{u}$ is "large."

Now consider an encryption cycle for two keys, which consists of ciphertexts

$$(\, \mathbf{x}_i = \mathbf{S}_{1-i}^{-1}[\mathbf{v}_i]\,,\, \mathbf{u}_i \approx \mathbf{S}_i^t\mathbf{v}_i \,)$$

for $i \in \{0, 1\}$, where $\mathbf{S}_i$ is the (secret) matrix produced by $\mathsf{GenTrap}$ using the ith secret key as randomness. Because the $\mathbf{x}_i$ are short, we have

$$\begin{aligned}\langle \mathbf{u}_0, \mathbf{x}_1\rangle &= \mathbf{u}_0^t \cdot \mathbf{x}_1 \approx \mathbf{v}_0^t\mathbf{S}_0 \cdot \mathbf{S}_0^{-1}[\mathbf{v}_1] = \mathbf{v}_0^t \cdot \mathbf{v}_1 = \langle \mathbf{v}_0, \mathbf{v}_1\rangle \\ \langle \mathbf{u}_1, \mathbf{x}_0\rangle &= \mathbf{u}_1^t \cdot \mathbf{x}_0 \approx \mathbf{v}_1^t\mathbf{S}_1 \cdot \mathbf{S}_1^{-1}[\mathbf{v}_0] = \mathbf{v}_1^t \cdot \mathbf{v}_0 = \langle \mathbf{v}_1, \mathbf{v}_0\rangle.\end{aligned}$$

Because the inner product is commutative, testing whether $\langle \mathbf{u}_0, \mathbf{x}_1\rangle \approx \langle \mathbf{u}_1, \mathbf{x}_0\rangle \pmod{q}$ will therefore detect a two-cycle. (For ordinary ciphertexts, the approximation is unlikely to hold, because the inner products are essentially uniform and independent.)

1.2.1 Challenges Beyond Two-Cycles

Generalizing the above construction to work for cycle lengths larger than two comes with several technical challenges. One is that there does not appear to be an appropriate generalization of the inner product $\langle \cdot, \cdot \rangle$ to three or more vectors. However, a promising idea is to replace $\mathbf{v}$ with a *matrix* $\mathbf{V}$ of many columns, and likewise replace $\mathbf{x}$ with $\mathbf{X} \leftarrow \hat{\mathbf{S}}^{-1}[\mathbf{V}]$, so that $\hat{\mathbf{S}} \cdot \mathbf{X} = \mathbf{V}$. Then for, say, a 3-cycle, if we could somehow arrange for $\mathbf{V}_i = \mathbf{Z}_i \cdot \mathbf{S}_i$ for some $\mathbf{Z}_i$, we would have the "telescoping product"

$$\begin{aligned}\mathbf{U}_0^t \cdot \mathbf{X}_1 \cdot \mathbf{X}_2 &= \mathbf{V}_0^t \cdot \mathbf{S}_0 \cdot \mathbf{S}_0^{-1}[\mathbf{V}_1] \cdot \mathbf{X}_2 \\ &= \mathbf{S}_0^t \cdot \mathbf{Z}_0^t \cdot \mathbf{Z}_1 \cdot \mathbf{S}_1 \cdot \mathbf{S}_1^{-1}[\mathbf{V}_2] \\ &= \mathbf{S}_0^t \cdot \mathbf{Z}_0^t \cdot \mathbf{Z}_1 \cdot \mathbf{Z}_2 \cdot \mathbf{S}_2,\end{aligned}$$

and similarly for $\mathbf{U}_1 \cdot \mathbf{X}_2 \cdot \mathbf{X}_0$. Unfortunately, we do not see any way to generate $\mathbf{V}_i = \mathbf{Z}_i \cdot \mathbf{S}_i$ in the encryption algorithm, because $\mathbf{S}_i$ is *secret* (it can only be obtained from the ith secret key). Alternatively, we might try to obtain a more "LWE-like" *approximation* $\mathbf{V}_i \approx \mathbf{Z}_i \cdot \mathbf{S}_i$ using the public key, but then the above equations *do not even hold approximately*, because $\mathbf{V}_0$ is "large" and hence amplifies the errors too much.

1.2.2 Our Solution

With the above attempt in mind, we take a different and arguably simpler approach to LWE-based cycle testers, which resolves both of the difficulties identified above. Our approach is easiest to understand in the ring setting first. For concreteness, define $R = \mathbb{Z}[X]/(X^n + 1)$ for n a power of two, and define $R_q = R/qR = \mathbb{Z}_q[X]/(X^n + 1)$ for a suitably large modulus q.

As in [BHW15], a secret key in our system is the randomness used by (a ring variant of) $\mathsf{GenTrap}$ to produce a row vector $\mathbf{a} \in R_q^m$ with a trapdoor $T_{\mathbf{a}}$. However, here we simply take $\mathbf{a}$ to be the *public* key, rather than using it as a vector of ring-LWE secrets.

To encrypt under public key $\mathbf{a}$, as in [BHW15] we interpret the message as randomness for $\mathsf{GenTrap}$ to obtain an $\hat{\mathbf{a}} \in R_q^m$ and trapdoor $T_{\hat{\mathbf{a}}}$. We then choose an $s \in R$ from the ring-LWE error distribution, let $\mathbf{b} \approx s \cdot \mathbf{a} \in R_q^m$ (where the approximation hides ring-LWE errors), and output the ciphertext

$$\mathbf{C} \leftarrow \hat{\mathbf{a}}^{-1}[\mathbf{b}] \in R^{m \times m},$$

where $\hat{\mathbf{a}}^{-1}[\mathbf{b}]$ uses $T_{\hat{\mathbf{a}}}$ to randomly sample a short matrix $\mathbf{C}$ over R such that $\hat{\mathbf{a}} \cdot \mathbf{C} = \mathbf{b}$. Notice that in contrast with [BHW15], the ciphertext is just one short matrix—it does not contain any "large" components, which will be important for cycle testing.

Consider now an encryption cycle of, say, three secret keys, which consists of ciphertexts

$$\mathbf{C}_i \leftarrow \mathbf{a}_{i-1}^{-1}[\mathbf{b}_i], \quad \mathbf{b}_i \approx s_i \cdot \mathbf{a}_i$$

for each $i \in \mathbb{Z}_3$ (where the subscript arithmetic is modulo three). We then have the telescoping product

$$\begin{aligned}\mathbf{a}_2 \cdot \mathbf{C}_0 \cdot \mathbf{C}_1 \cdot \mathbf{C}_2 &= \mathbf{a}_2 \cdot \mathbf{a}_2^{-1}[\mathbf{b}_0] \cdot \mathbf{C}_1 \cdot \mathbf{C}_2 \\ &\approx s_0 \cdot \mathbf{a}_0 \cdot \mathbf{a}_0^{-1}[\mathbf{b}_1] \cdot \mathbf{C}_2 \\ &\approx s_0 \cdot s_1 \cdot \mathbf{a}_1 \cdot \mathbf{a}_1^{-1}[\mathbf{b}_2] \\ &\approx s_0 \cdot s_1 \cdot s_2 \cdot \mathbf{a}_2,\end{aligned}$$

where the approximations hold because all the s_i and $\mathbf{C}_i$ are short. Similarly,

$$\mathbf{a}_0 \cdot \mathbf{C}_1 \cdot \mathbf{C}_2 \cdot \mathbf{C}_0 \approx s_1 \cdot s_2 \cdot s_0 \cdot \mathbf{a}_0.$$

Now because the ring R is commutative, the above right-hand sides are almost identical, except for the different public keys $\mathbf{a}_0, \mathbf{a}_2$. But this issue is easily addressed: the $\mathsf{GenTrap}$ algorithm comes in a version that takes a vector over R_q as a public parameter, and outputs an $\mathbf{a}$ having that vector as its *prefix*. Therefore, our cycle tester just checks whether the first entries of the above products (corresponding to the common prefix of $\mathbf{a}_0, \mathbf{a}_2$) are approximately equal. More precisely, the difference should be smaller than some bound that depends on the maximum cycle length k we want to be able to detect; this induces our choice of the modulus q. Finally, notice that the tester also works equally well for cycles of length k' for $2 \leq k' \leq k$.

1.2.3 Adapting to Plain LWE

There is a standard mechanical translation of cryptosystems from ring-LWE to plain LWE, which replaces every uniformly random $a \in R_q$ with a uniformly random matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times n}$, and every error term $s \in R$ with a matrix $\mathbf{S} \in \mathbb{Z}^{n \times n}$ whose entries are drawn independently from the LWE error distribution. However, when this translation is applied to the above scheme, it is easy to see that the cycle tester does not work, because the error matrices $\mathbf{S}_i$ are unlikely to commute with each other under multiplication.

We resolve this difficulty by introducing a new *tensoring* technique that guarantees commutativity. (We believe that the technique will find additional applications.) The central fact we use is that the tensor product of square n-dimensional matrices obeys the following special case of the *mixed-product property*:

$$\mathbf{S}_1 \otimes \mathbf{S}_2 = (\mathbf{S}_1 \otimes \mathbf{I}_n) \cdot (\mathbf{I}_n \otimes \mathbf{S}_2) = (\mathbf{I}_n \otimes \mathbf{S}_2) \cdot (\mathbf{S}_1 \otimes \mathbf{I}_n) \in \mathbb{Z}^{n^2 \times n^2}.$$

In particular, the matrices $\mathbf{S}_1 \otimes \mathbf{I}_n$ and $\mathbf{I}_n \otimes \mathbf{S}_2$ commute under multiplication. (Naturally, the above equations generalize to the tensor product of any $k > 2$ matrices.)

We apply the above facts in our plain-LWE cycle tester as follows. When encrypting to the ith public key, we use an LWE secret matrix

$$\mathbf{S}'_i = \underbrace{\mathbf{I}_n \otimes \cdots \otimes \mathbf{I}_n}_{i \text{ terms}} \otimes \mathbf{S}_i \otimes \underbrace{\mathbf{I}_n \otimes \cdots \otimes \mathbf{I}_n}_{k-i-1 \text{ terms}} \in \mathbb{Z}^{n^k \times n^k},$$

where $\mathbf{S}_i \in \mathbb{Z}^{n \times n}$ has entries drawn from the error distribution. By the above, these $\mathbf{S}_i$ all commute with each other under multiplication, allowing us to conclude that (certain entries of) the telescoping products are approximately equal. Also notice that it is not necessary for all the $\mathbf{S}_i$ to appear in the final product, so the same cycle tester also detects k'-cycles for $2 \leq k' \leq k$.

In order for all this to work, the public key matrices $\mathbf{A}_i$ must have n^k rows, which is why our construction is limited to constant $k = O(1)$. Of course, it is not immediately obvious whether LWE is actually hard for such highly structured secret matrices $\mathbf{S}'_i$. Fortunately, we prove that this form of the problem is *equivalent* to n-dimensional LWE with the same error distribution, up to a polynomial factor in the number of samples given to the attacker. Known worst-case hardness theorems for LWE are essentially agnostic to the number of samples, so the reduction's lossiness in this respect is of little concern.

2 Preliminaries

For a positive integer t we let $[t] = \{0, \ldots, t-1\}$. The primary security parameter is denoted λ.

Tensor products. The *tensor* (or *Kronecker*) product $\mathbf{A} \otimes \mathbf{B}$ of an m_1-by-n_1 matrix $\mathbf{A}$ with an m_2-by-n_2 matrix $\mathbf{B}$, both over a common ring $\mathcal{R}$, is the $m_1 m_2$-by-$n_1 n_2$ block matrix consisting of m_2-by-n_2 blocks, whose (i,j)th block is $a_{i,j} \cdot \mathbf{B}$, where $a_{i,j}$ denotes the (i,j)th entry of $\mathbf{A}$. Equivalently, we can view $\mathbf{A} \otimes \mathbf{B}$ as having rows indexed by $[m_1] \times [m_2]$ and columns indexed by $[n_1] \times [n_2]$, where the $((i_1,i_2),(j_1,j_2))$th entry is $a_{i_1,j_1} \cdot b_{i_2,j_2}$. This corresponds to the previous definition by "flattening" the row and column index sets using the bijection that maps $(k_1,k_2) \in [\ell_1] \times [\ell_2]$ to $k_1 \cdot \ell_2 + k_2 \in [\ell_1 \ell_2]$.

We extensively use the *mixed-product property* of tensor products, which says that

$$(\mathbf{A} \otimes \mathbf{B}) \cdot (\mathbf{C} \otimes \mathbf{D}) = (\mathbf{AC}) \otimes (\mathbf{BD})$$

for any matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ of compatible dimensions. In particular,

$$(\mathbf{A} \otimes \mathbf{B}) = (\mathbf{A} \otimes \mathbf{I}_{\text{height}(\mathbf{B})}) \cdot (\mathbf{I}_{\text{width}(\mathbf{A})} \otimes \mathbf{B}) = (\mathbf{I}_{\text{height}}(\mathbf{A}) \otimes \mathbf{B}) \cdot (\mathbf{A} \otimes \mathbf{I}_{\text{width}}(\mathbf{B})).$$

Subgaussians. For analyzing error growth in our schemes it will be convenient to use the notion of *subgaussian* random variables and matrices. We say that a real random variable X (or its distribution) is subgaussian with parameter s if for all $t \in \mathbb{R}$, the (scaled) moment-generating function satisfies[1]

$$\mathbb{E}[\exp(2\pi t X)] \leq (1 + \operatorname{negl}(\lambda)) \cdot \exp(\pi s^2 t^2).$$

More generally, we say that a random matrix (over vector) $\mathbf{X}$ is subgaussian with parameter s if $\mathbf{u}^t \mathbf{X} \mathbf{v}$ is subgaussian with parameter s for all unit vectors $\mathbf{u}, \mathbf{v}$. It follows immediately from the definitions that a $\operatorname{poly}(\lambda)$-dimensional matrix made up of independent subgaussian entries, or of independent subgaussian rows or columns, with common parameter s is itself subgaussian with parameter s.

The largest singular value, also known as *spectral norm*, of a matrix $\mathbf{X}$ is defined as $s_1(\mathbf{X}) := \max_{\mathbf{u} \neq \mathbf{0}} \|\mathbf{Xu}\| / \|\mathbf{u}\|$. It is clear that the spectral norm is sub-additive and sub-multiplicative: $s_1(\mathbf{X} + \mathbf{Y}) \leq s_1(\mathbf{X}) + s_1(\mathbf{Y})$ and $s_1(\mathbf{XY}) \leq s_1(\mathbf{X}) \cdot s_1(\mathbf{Y})$. We use the following standard fact about subgaussian matrices; see [Ver12] for a proof.

Proposition 2.1. *For a subgaussian matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$ with parameter s, we have $s_1(\mathbf{X}) \leq s \cdot O(\sqrt{m} + \sqrt{n})$ except with probability at most $2^{-\Omega(m+n)}$.*

2.1 Cryptographic Definitions

Here we present some cryptographic definitions. The definition of k-cycle tester is from [BHW15].

Definition 2.2. Let $\Pi = (\mathsf{Setup}, \mathsf{Gen}, \mathsf{Enc})$ be a public-key encryption scheme (omitting the decryption algorithm) for message space $\mathcal{M} = \mathcal{M}_\lambda$. We say that Π is $\mathsf{IND\text{-}CPA}$ secure if every efficient adversary $\mathcal{A}$ has negligible (in λ) advantage in distinguishing the following two games for $b \in \{0, 1\}$:

1. Generate $pp \leftarrow \mathsf{Setup}(1^\lambda)$ and $(pk, sk) \leftarrow \mathsf{Gen}(pp)$.
2. Given (pp, pk) to $\mathcal{A}$, which outputs a pair of messages $(m_0, m_1) \in \mathcal{M}^2$.
3. Generate $c \leftarrow \mathsf{Enc}(pk, m_b)$ and give c to the adversary.

[1] We remark that the $1 + \operatorname{negl}(\lambda)$ factor makes this a slight relaxation of the standard definition of subgaussian; it coincides with the notion of $\operatorname{negl}(\lambda)$-subgaussian from [MP12].

Definition 2.3. Let $\Pi = (\mathsf{Setup}, \mathsf{Gen}, \mathsf{Enc})$ be a public-key encryption scheme (omitting the decryption algorithm) for message space $\mathcal{M} = \mathcal{M}_\lambda \supseteq \mathcal{S}_\lambda$, where $\mathcal{S}_\lambda$ denotes the secret-key space for security parameter λ. We say that Π is $\mathsf{IND\text{-}CIRC\text{-}CPA}^k$ secure if the following two games are computationally indistinguishable.

1. Generate $pp \leftarrow \mathsf{Setup}(1^\lambda)$ and $(pk_i, sk_i) \leftarrow \mathsf{Gen}(pp)$ for every $i \in \mathbb{Z}_k$.
2. In Game 0, let $c_i \leftarrow \mathsf{Enc}(pk_i, sk_{i-1})$ for $i \in \mathbb{Z}_k$ (where arithmetic in the subscripts is modulo k).

   In Game 1, let $c_i \leftarrow \mathsf{Enc}(pk_i, 0)$ for $i \in \mathbb{Z}_k$ (where $0 \in \mathcal{M}$ denotes some arbitrary fixed message).
3. Output $(pp, (pk_i)_{i \in \mathbb{Z}_k}, (c_i)_{i \in \mathbb{Z}_k})$.

Definition 2.4 (Cycle Tester [BHW15]). Let $\Gamma = (\mathsf{Setup}, \mathsf{Gen}, \mathsf{Enc}, \mathsf{Test})$ be a tuple of randomized algorithms for which:

- $\Pi = (\mathsf{Setup}, \mathsf{Gen}, \mathsf{Enc})$ is a public-key encryption scheme for message space $\mathcal{M} = \mathcal{M}_\lambda \supseteq \mathcal{S}_\lambda$;
- $\mathsf{Test}((pk_i, c_i)_{i \in \mathbb{Z}_k})$, given a tuple of public keys pk_i and corresponding ciphertexts c_i, outputs a bit $b \in \{0, 1\}$.

We say that Γ is a *k-cycle tester* if Π is $\mathsf{IND\text{-}CPA}$ secure, and if Test has *non-negligible* advantage in the $\mathsf{IND\text{-}CIRC\text{-}CPA}^k$ game against Π.

2.2 Learning With Errors

Definition 2.5. For positive integer dimensions n, m, modulus q, and error distribution χ over $\mathbb{Z}$, the decision-LWE$_{n,q,\chi,m}$ problem is to distinguish, with non-negligible advantage, between $(\mathbf{A}; \mathbf{b}^t = \mathbf{s}^t\mathbf{A} + \mathbf{e}^t)$ where $\mathbf{A} \leftarrow \mathbb{Z}_q^{n \times m}$, $\mathbf{s} \leftarrow \chi^n$, $\mathbf{e} \leftarrow \chi^m$, and *uniformly random* $(\mathbf{A}; \mathbf{b}^t)$ of the same dimensions.[2]

A standard instantiation of LWE is to let χ be a *discrete Gaussian* distribution (over $\mathbb{Z}$) with parameter $r = 2\sqrt{n}$, which is known to be subgaussian with parameter r (see [MP12]). For this parameterization, and for any polynomially bounded m, it is known that LWE is at least as hard as *quantumly* approximating certain "short vector" problems on n-dimensional lattices, in the worst case, to within $\tilde{O}(q\sqrt{n})$ factors [Reg05]. Classical reductions are also known for different parameterizations [Pei09, BLP$^+$13].

A standard hybrid argument shows that the *multi-secret* form of LWE—which is to distinguish

$$\begin{pmatrix} \mathbf{A} \\ \mathbf{B} = \mathbf{S}\mathbf{A} + \mathbf{E} \end{pmatrix} \in \mathbb{Z}_q^{(n+t) \times m}$$

from uniform, where $\mathbf{A} \leftarrow \mathbb{Z}_q^{n \times m}$, $\mathbf{S} \leftarrow \chi^{t \times n}$, and $\mathbf{E} \leftarrow \chi^{t \times m}$ for some desired $t, m = \text{poly}(n)$—is equivalent to the above single-secret version, up to a t factor loss in the distinguishing advantage.

Tensored form. In this work we rely on another equivalent form of LWE, which we call the *tensored* form. Let $m, t = \text{poly}(n)$ be as above, and additionally let $l, r = \text{poly}(n)$ be arbitrary. The problem is to distinguish

$$\begin{pmatrix} \mathbf{A} \\ \mathbf{B} = (\mathbf{I}_l \otimes \mathbf{S} \otimes \mathbf{I}_r) \cdot \mathbf{A} + \mathbf{E} \end{pmatrix} \in \mathbb{Z}_q^{l(n+t)r \times m}$$

from uniform, where $\mathbf{A} \leftarrow \mathbb{Z}_q^{lnr \times m}$, $\mathbf{S} \leftarrow \chi^{t \times n}$, and $\mathbf{E} \leftarrow \chi^{ltr \times m}$.

[2] Notice that the coordinates of $\mathbf{s}$ are drawn from the error distribution χ; as shown in [ACPS09], this form of the problem is equivalent (up to a small difference in m) to the one where $\mathbf{s} \leftarrow \mathbb{Z}_q^n$ is drawn uniformly at random.

Lemma 2.6. *The tensored form of LWE for parameters n, t, m, l, r is equivalent to the multi-secret form for the same n, t and $M = mlr$ samples.*

Proof. The equivalence follows simply by an appropriate (efficient and reversible) reindexing. Specifically, given a multi-secret instance $(\mathbf{A}; \mathbf{B}) \in \mathbb{Z}_q^{(n+t)\times M}$, we transform it to a tensored instance $(\mathbf{A}'; \mathbf{B}') \in \mathbb{Z}_q^{l(n+t)r\times m}$ as follows. For convenience, we construct $\mathbf{A}'$ by indexing its rows by $[l] \times [n] \times [r]$ in the standard way, and similarly for $\mathbf{B}'$. We partition $\mathbf{A}$ into m blocks, each consisting of lr columns of dimension n. We arbitrarily index these columns by $[l] \times [r]$, and arrange them into a single column indexed by $[l] \times [n] \times [r]$ in the obvious way; the matrix $\mathbf{A}'$ is made up of these m columns. Similarly, we construct $\mathbf{B}'$ from $\mathbf{B}$ by grouping each block of lr columns of dimension t into a single column vector indexed $[l] \times [t] \times [r]$. It is easy to see that if $(\mathbf{A}; \mathbf{B})$ is uniformly random, then so is $(\mathbf{A}'; \mathbf{B}')$. Furthermore, by construction and by definition of matrix multiplication it can be verified that if $\mathbf{B} = \mathbf{S}\mathbf{A} + \mathbf{E}$ for some $\mathbf{S}, \mathbf{E}$, then $\mathbf{B}' = (\mathbf{I}_l \otimes \mathbf{S} \otimes \mathbf{I}_r) \cdot \mathbf{A}' + \mathbf{E}'$, where $\mathbf{E}'$ is obtained from $\mathbf{E}$ in exactly the same way that $\mathbf{B}'$ is obtained from $\mathbf{B}$. Therefore, the transformation is a tight reduction from the multi-secret to the tensored form. Moreover, the transformation is efficiently reversible, which gives a reduction in the opposite direction. □

2.3 Lattice Trapdoors

We recall some standard facts about trapdoors and preimage sampling for cryptographic lattices; for full details, see [GPV08, MP12]. There exist efficient randomized algorithms $\mathsf{GenTrap}$, $\mathsf{SampleDom}$, and $\mathsf{SamplePre}$ having the following properties. For any positive integers n, q, there exist suitable $\bar{m} < m = O(n \log q)$ for which the following hold (the parameters $n, q, \bar{m}, m$ are implicit inputs to all the algorithms):

- $\mathsf{GenTrap}(\bar{\mathbf{A}}; \mathbf{R})$ takes some $\bar{\mathbf{A}} \in \mathbb{Z}_q^{n\times\bar{m}}$ and random coins $\mathbf{R} \in \mathcal{R}$ from a certain space $\mathcal{R}$, and outputs a matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ whose first $\bar{m}$ columns are $\bar{\mathbf{A}}$, and for which $\mathbf{R}$ serves as a "trapdoor."
- $\mathsf{SampleDom}()$ outputs a random $\mathbf{x} \in \mathbb{Z}^m$, drawn from a certain distribution D. For brevity we usually write $\mathbf{x} \leftarrow D$.
- For any $\bar{\mathbf{A}} \in \mathbb{Z}_q^{n\times\bar{m}}$ and $\mathbf{R} \in \mathcal{R}$ defining $\mathbf{A}$ as above, and any $\mathbf{u} \in \mathbb{Z}_q^n$, $\mathsf{SamplePre}(\bar{\mathbf{A}}, \mathbf{R}, \mathbf{u})$ outputs a random $\mathbf{x} \in \mathbb{Z}^m$ (drawn from a certain distribution) such that $\mathbf{A}\mathbf{x} = \mathbf{u}$.

  When $\mathbf{A}$ and $\mathbf{R}$ are clear from context, we usually write $\mathbf{A}^{-1}[\mathbf{u}]$ for the sake of brevity, and because it satisfies the identity $\mathbf{A} \cdot \mathbf{A}^{-1}[\mathbf{u}] = \mathbf{u}$. (We stress that $\mathbf{A}^{-1}[\cdot]$ denotes a *randomized algorithm*, not a formal matrix inverse.)

We extend the above notation column-wise to matrices, i.e., D^ℓ is the distribution over $\mathbb{Z}^{m\times\ell}$ in which the columns are drawn independently from D, and $\mathbf{A}^{-1}[\mathbf{B}] \in \mathbb{Z}^{m\times\ell}$ for $\mathbf{B} \in \mathbb{Z}_q^{n\times\ell}$ applies $\mathbf{A}^{-1}$ independently to each column of $\mathbf{B}$.

Proposition 2.7. *The above algorithms satisfy the following statistical properties:*

1. *For uniformly random $\mathbf{A} \leftarrow \mathbb{Z}_q^{n\times m}$ and $\mathbf{x} \leftarrow D$, the distribution of $(\mathbf{A}, \mathbf{A}\mathbf{x})$ is within negligible statistical distance of uniform.*
2. *For uniformly random $\bar{\mathbf{A}}$ and $\mathbf{R} \leftarrow \mathcal{R}$, the distribution of $\mathbf{A} = \mathsf{GenTrap}(\bar{\mathbf{A}}, \mathbf{R})$ is within negligible statistical distance of uniform.*
3. *For* any $\bar{\mathbf{A}}$ *and* any $\mathbf{R} \in \mathcal{R}$ *defining $\mathbf{A} = \mathsf{GenTrap}(\bar{\mathbf{A}}; \mathbf{R})$, the following experiments are within negligible statistical distance:*

(a) choose $\mathbf{x} \leftarrow D$ *and output* $(\mathbf{x}, \mathbf{u} = \mathbf{A}\mathbf{x})$*;*

(b) choose uniformly random $\mathbf{u} \leftarrow \mathbb{Z}_q^n$*, let* $\mathbf{x} \leftarrow \mathbf{A}^{-1}[\mathbf{u}]$*, and output* $(\mathbf{x}, \mathbf{u})$*.*

4. *For any* $\mathbf{A}$ *output by* GenTrap *(on randomness* $\mathbf{R}$*), and any* $\mathbf{u} \in \mathbb{Z}_q^n$*, the distribution* $\mathbf{A}^{-1}[\mathbf{u}]$ *is subgaussian with parameter* $\tilde{O}(m) = \tilde{O}(n \log q)$*.*

Remark 2.8. We emphasize that Item 3 of Proposition 2.7 applies for *any* (possibly adversarial) choice of the trapdoor $\mathbf{R}$, which is needed in our application because the trapdoor will indeed be provided by the adversary. Fortunately, the GenTrap and SamplePre algorithms described in [MP12] can easily be instantiated to satisfy this property. In brief, this is GenTrap produces a short random matrix $\mathbf{R} \in \mathcal{R}$ as the trapdoor, and SamplePre works for any Gaussian parameter exceeding a certain $\tilde{\Theta}(s_1(\mathbf{R}))$ bound. By defining $\mathcal{R}$ to be, say, the set of all binary matrices of appropriate dimensions, we ensure that $s_1(\mathbf{R}) \leq m$ for *every* $\mathbf{R} \in \mathcal{R}$, while also satisfying Item 2 via the leftover hash lemma.

2.4 The Ring Setting

Here we provide some background on rings, their geometry, and ring-LWE; then we recall analogous facts about trapdoors in the ring setting. For more details see [LPR10, MP12]. (This material is only used for our ring-LWE construction in Section 4, and may be safely skipped.)

For simplicity, we work in the $2n$th cyclotomic ring $R := \mathbb{Z}[X]/(X^n+1)$ for n a power of two. (However, all of our results can be adapted to arbitrary cyclotomics using the techniques from [LPR13].) The *canonical embedding* $\sigma\colon R \to \mathbb{C}^n$ maps $r \in R$ to $(\sigma_i(r))_{i \in \mathbb{Z}_{2n}^*}$, where $\sigma_i(r) = r(\omega^i)$ and $\omega = \exp(\pi\sqrt{-1}/n) \in \mathbb{C}$ is the principal complex $2n$th root of unity. (Notice that this definition is agnostic to the choice of $\mathbb{Z}[X]$-representative of $r \in \mathbb{Z}[X]/(X^n+1)$, which makes it "canonical.")

We use the canonical embedding to endow R with a geometry. Specifically, for a ring element $r \in R$ we define $\|r\| := \|\sigma(r)\|$ and $\|r\|_\infty := \|\sigma(r)\|_\infty$. We extend the norm notation to vectors and matrices by defining $\|\mathbf{x}\| = (\sum_i \|x_i\|^2)^{1/2}$ for any vector $\mathbf{x}$ over R, and $\|\mathbf{X}\|_\infty = \max \|x_{i,j}\|_\infty$ for any vector or matrix $\mathbf{X}$ over R. Finally, we define the *spectral norm* of $\mathbf{X}$ as

$$s_1(\mathbf{X}) := \sup_{\mathbf{u} \neq \mathbf{0}} \|\mathbf{X}\mathbf{u}\| / \|\mathbf{u}\|,$$

where the supremum is taken over all nonzero vectors (of appropriate dimension) over R. Clearly, the spectral norm is sub-additive and sub-multiplicative: $s_1(\mathbf{X}+\mathbf{Y}) \leq s_1(\mathbf{X}) + s_1(\mathbf{Y})$ and $s_1(\mathbf{X}\mathbf{Y}) \leq s_1(\mathbf{X}) \cdot s_1(\mathbf{Y})$. The following standard fact relates the spectral and ℓ_∞ norms.

Proposition 2.9. *For any matrix* $\mathbf{E} \in R^{l \times k}$ *we have* $s_1(\mathbf{E}) \leq \sqrt{lk} \cdot \|\mathbf{E}\|_\infty$*.*

The following standard fact bounds the coefficients of a ring element $r \in R$ by its ℓ_∞ norm.

Proposition 2.10. *For a ring element* $r \in R$*, let* $r = \sum_{j=0}^{n-1} r_j \cdot X^j \in \mathbb{Z}[X]$ *for* $r_j \in \mathbb{Z}$ *denote its canonical representative (with respect to the standard power basis of R). Then* $r_j \leq \|r\|_\infty$ *for every* j*.*

2.4.1 Ring-LWE

For an integer q, define $R_q := R/qR = \mathbb{Z}_q[X]/(X^n+1)$.

Definition 2.11. Let χ be an error distribution over R. The decision-RLWE$_{R,q,\chi,m}$ problem is to distinguish, with non-negligible advantage, between $(\mathbf{a}; \mathbf{b} = s \cdot \mathbf{a} + \mathbf{e}) \in R_q^m \times R_q^m$ where $\mathbf{a} \leftarrow R_q^m$, $s \leftarrow \chi$, $\mathbf{e} \leftarrow \chi^m$, and *uniformly random* $(\mathbf{a}; \mathbf{b})$ of the same dimensions.

For appropriate parameters, decision-RLWE problem is (quantumly) at least as hard as the $(q \cdot \mathrm{poly}(n,m))$-approximate shortest vector problem on *any* ideal lattice in R, i.e., in the worst case [LPR10]. The standard error distribution for which this theorem applies is a sufficiently wide discrete Gaussian distribution χ over R, for which

$$\Pr_{e \leftarrow \chi}[\|e\|_\infty > n^c] = \mathrm{negl}(n) \tag{2.1}$$

for some universal constant $c > 1$.

2.4.2 Trapdoors

Similarly to the plain setting, there are efficient randomized algorithms $\mathsf{GenTrap}$, $\mathsf{SampleDom}$, and $\mathsf{SamplePre}$ having the following properties. For any modulus q, there exist suitable $\bar{m} < m = \tilde{O}(\log q)$ for which the following hold (the parameters $R, q, \bar{m}, m$ are implicit inputs to all the algorithms):

- $\mathsf{GenTrap}(\bar{\mathbf{a}}; \mathbf{R})$ takes some $\bar{\mathbf{a}} \in R_q^{\bar{m}}$ and random $\mathbf{R} \in \mathcal{R}$ from a certain space $\mathcal{R}$, and outputs a vector $\mathbf{a} \in R_q^m$ whose first $\bar{m}$ components are $\bar{\mathbf{a}}$, and for which $\mathbf{R}$ serves as a "trapdoor."
- $\mathsf{SampleDom}()$ outputs a random column vector $\mathbf{x}^t \in R^m$, drawn from a certain distribution D. For brevity we usually write $\mathbf{x}^t \leftarrow D$.
- For any $\bar{\mathbf{a}} \in R_q^{\bar{m}}$ and $\mathbf{R} \in \mathcal{R}$ defining $\mathbf{a}$ as above, and any $u \in R_q$, $\mathsf{SamplePre}(\bar{\mathbf{a}}, \mathbf{R}, u)$ outputs a random column vector $\mathbf{x}^t \in R^m$ (drawn from a certain distribution) such that $\mathbf{a} \cdot \mathbf{x}^t = u$.

  We usually write $\mathbf{a}^{-1}[u]$ for the sake of brevity, and because it satisfies the identity $\mathbf{a} \cdot \mathbf{a}^{-1}[u] = u$. Moreover, D^l is the distribution over $R^{m \times l}$ in which the columns are drawn independently from D. The notation $\mathbf{a}^{-1}[\mathbf{v}] \in R_q^{m \times l}$, where $\mathbf{v} \in R_q^l$, applies $\mathbf{a}^{-1}$ to each component of $\mathbf{v}$ independently.

The following proposition follows by a standard adaptation of "plain" trapdoor constructions (e.g., [MP12]) to the ring setting, and by the regularity lemma for rings given in [LPR13].

Proposition 2.12. *The above algorithms satisfy the following statistical properties:*

1. *For uniformly random* $\mathbf{a} \leftarrow R_q^m$ *and* $\mathbf{x}^t \leftarrow D$*, the distribution of* $(\mathbf{a}, \mathbf{a} \cdot \mathbf{x}^t) \in R^{m+1}$ *is within negligible statistical distance of uniform.*
2. *For uniformly random* $\bar{\mathbf{a}}$ *and* $\mathbf{R} \leftarrow \mathcal{R}$*, the distribution of* $\mathbf{a} = \mathsf{GenTrap}(\bar{\mathbf{a}}, \mathbf{R})$ *is within negligible statistical distance of uniform.*
3. *For* any $\bar{\mathbf{a}}$ *and* any $\mathbf{R} \in \mathcal{R}$ *defining* $\mathbf{a} = \mathsf{GenTrap}(\bar{\mathbf{a}}; \mathbf{R})$*, the following experiments are within negligible statistical distance:*

   (a) choose $\mathbf{x}^t \leftarrow D$ *and output* $(\mathbf{x}, u = \mathbf{a} \cdot \mathbf{x}^t)$*;*

   (b) choose uniformly random $u \leftarrow R_q$*, let* $\mathbf{x}^t \leftarrow \mathbf{a}^{-1}[u]$*, and output* $(\mathbf{x}, u)$*.*
4. *There exists a universal constant* $c > 1$ *such that, for any* $\mathbf{a}$ *output by* $\mathsf{GenTrap}$ *(on randomness* $\mathbf{R}$*), and for any* $u \in R_q$*,*

$$\Pr\left[\left\|\mathbf{a}^{-1}[u]\right\|_\infty > n^c\right] = \mathrm{negl}(n).$$

3 LWE-Based Construction

In this section we construct, for any constant $k \geq 2$, a k-cycle tester that is IND-CPA secure based on the conjectured hardness of (plain) LWE, appropriately parameterized. The scheme involves the following parameters:

- $N := n^k$ for a positive integer n, an integer modulus q, and an error distribution χ over $\mathbb{Z}$, where n, q, χ are the parameters of the underlying LWE problem. For concreteness, we use the standard LWE error distribution χ, which is subgaussian with parameter $O(\sqrt{n})$.
- $\bar{M} < M = O(N \log q)$, where $\bar{M}, M$ are the dimensions associated with $\mathsf{GenTrap}$ for N, q.
- The secret-key and message spaces are both the randomness/trapdoor space $\mathcal{R}$ of $\mathsf{GenTrap}$ when given an N-by-$\bar{M}$ input.

Finally, each key is uniquely and arbitrarily identified with some $i \in \mathbb{Z}_k = \{0, \dots, k-1\}$, which is provided to the key-generation algorithm. The tester is defined as follows.

- $\mathsf{Setup}()$: output a uniformly random $\bar{\mathbf{A}} \leftarrow \mathbb{Z}_q^{N \times \bar{M}}$.
- $\mathsf{Gen}(i, \bar{\mathbf{A}})$: let $\mathbf{A}_i = \mathsf{GenTrap}(\bar{\mathbf{A}}; \mathbf{R}_i)$ for $\mathbf{R}_i \leftarrow \mathcal{R}$, and output $(i, \mathbf{A}_i)$ as the public key and the trapdoor $\mathbf{R}_i$ as the secret key.

  Recall from Proposition 2.7 that the first $\bar{M}$ columns of $\mathbf{A}_i$ are $\bar{\mathbf{A}}$, and that $\mathbf{A}_i$ is negligibly far from uniform over the random choice of $\bar{\mathbf{A}}$ and $\mathbf{R}_i$.
- $\mathsf{Enc}((i, \mathbf{A}_i), \mathbf{R} \in \mathcal{R})$: let $\mathbf{A} = \mathsf{GenTrap}(\bar{\mathbf{A}}; \mathbf{R})$, so that $\mathbf{R}$ is a trapdoor for $\mathbf{A}$. Choose an LWE secret matrix $\mathbf{S}_i \leftarrow \chi^{n \times n}$ and an error matrix $\mathbf{E}_i \leftarrow \chi^{N \times M}$, and output the ciphertext matrix

  $$\mathbf{C} \leftarrow \mathbf{A}^{-1}[\mathbf{S}'_i \cdot \mathbf{A}_i + \mathbf{E}_i] \in \mathbb{Z}^{M \times M},$$
  $$\text{where } \mathbf{S}'_i = (\mathbf{I}_{n^i} \otimes \mathbf{S}_i \otimes \mathbf{I}_{n^{k-i-1}}) \in \mathbb{Z}^{N \times N}.$$

  (The $\mathbf{A}^{-1}$ operation is performed using trapdoor $\mathbf{R}$.)
- $\mathsf{Test}((\mathbf{A}_i, \mathbf{C}_i)_{i \in \mathbb{Z}_k})$: given public key matrices $\mathbf{A}_i$ and ciphertexts $\mathbf{C}_i$, check whether

  $$(\mathbf{A}_{k-1} \cdot \mathbf{C}_0 \cdot \mathbf{C}_1 \cdots \mathbf{C}_{k-1} - \mathbf{A}_0 \cdot \mathbf{C}_1 \cdots \mathbf{C}_{k-1} \cdot \mathbf{C}_0) \cdot \bar{\mathbf{I}} \in (-q/4, q/4)^{N \times \bar{M}} \pmod{q}, \tag{3.1}$$

  where $\bar{\mathbf{I}} = \left(\begin{smallmatrix} \mathbf{I}_{\bar{M}} \\ \mathbf{0} \end{smallmatrix}\right) \in \mathbb{Z}^{M \times \bar{M}}$. (Notice that $\mathbf{A}_i \cdot \bar{\mathbf{I}} = \bar{\mathbf{A}}$ for every i, which we use in the analysis below.)

Remark 3.1. In Equation (3.1), the choice of products appearing in the difference is not special; the difference between any two products $\mathbf{A}_i \cdot \mathbf{C}_{i+1} \cdot \mathbf{C}_{i+2} \cdots \mathbf{C}_i$ for distinct values of $i \in \mathbb{Z}_k$ would work equally well.

Remark 3.2. The number and order of ciphertexts in an encryption cycle is also not too important. The Test algorithm naturally generalizes to work on any k' public keys and ciphertexts indexed by an ordered set $S \subseteq \mathbb{Z}_k$, for $2 \leq k' \leq k$. We simply take the difference of two products $\mathbf{A}_i \cdot \prod_{j \in S} \mathbf{C}_j$ for two distinct i, where the order of indices j cyclically follows the order of S and ends with $j = i$.

In the remainder of this section we prove the following theorem:

Theorem 3.3. *For any constant $k \geq 2$ and a sufficiently large $q = \tilde{O}(n^{3(k^2-1)/2})$, the above scheme is a k'-cycle tester for $2 \leq k' \leq k$, assuming the hardness of decision-LWE$_{n,q,\chi,M \cdot n^{k-1}}$.*

Recall that the LWE instantiation from Theorem 3.3 is at least as hard as (quantumly) approximating certain lattice problems on n-dimensional lattices, in the worst case, to within $\tilde{O}(n^{3k^2/2-1}) = \mathrm{poly}(n)$ factors, which is conjectured to be exponentially hard in n.

In Section 3.1 below we prove IND-CPA security, in Section 3.2 we show that Test almost always accepts on an encryption cycle, and in Section 3.3 we show that Test almost never accepts on a non-cycle. Together these prove Theorem 3.3.

3.1 Security

Lemma 3.4. *The tuple* $(\mathsf{Setup}, \mathsf{Gen}, \mathsf{Enc})$ *is IND-CPA secure under the LWE assumption from Theorem 3.3.*

Proof. We consider the following sequence of hybrid experiments, showing that adjacent hybrids are indistinguishable (either computationally or statistically), and that the last one does not depend on the adversary's choice of challenge message, which proves the claim. For simplicity, assume that the adversary names some target identity $i \in \mathbb{Z}_k$ at the start of the IND-CPA game. (The proof easily adapts to the case where the adversary adaptively chooses i after seeing all the public keys.)

Hybrid 1: Here the matrix $\mathbf{A}_i \in \mathbb{Z}_q^{N \times M}$ in the public key is generated uniformly at random, instead of by $\mathsf{GenTrap}$. By Item 2 of Proposition 2.7, this experiment is statistically indistinguishable from the real IND-CPA game.

Hybrid 2: Here the matrix $\mathbf{B}_i \in \mathbb{Z}_q^{N \times M}$ given as input to the $\mathbf{A}^{-1}$ operation is chosen uniformly at random, rather than as $\mathbf{B}_i = \mathbf{S}'_i \cdot \mathbf{A}_i + \mathbf{E}_i$ (as in the previous hybrid).

Using the tensored form of LWE, which by Lemma 2.6 is equivalent to the one appearing in the theorem statement, a straightforward reduction shows that this experiment is computationally indistinguishable from the previous one. Specifically, given an instance $(\mathbf{A}'; \mathbf{B}')$ of the tensored form of LWE, the reduction sets $\mathbf{A}_i = \mathbf{A}'$, $\mathbf{B}_i = \mathbf{B}'$, and finally lets $\mathbf{C}_i \leftarrow \mathbf{A}^{-1}[\mathbf{B}_i]$, using the adversary's challenge message to define $\mathbf{A}$ and compute the $\mathbf{A}^{-1}[\cdot]$ operation (using the $\mathsf{SamplePre}$ algorithm) in the usual way.

Hybrid 3: Here the matrix $\mathbf{C}_i$ is drawn from D^M, i.e., each column is independently drawn from D, instead of by invoking $\mathbf{A}^{-1}[\mathbf{B}_i]$ for a matrix $\mathbf{A}$ defined by the adversary's challenge message.

We claim that for any choice of $\bar{\mathbf{A}}$ and challenge message, this experiment is within negligible statistical distance of the previous one. This follows immediately by Item 3 of Proposition 2.7, applied across each pair of corresponding columns of $\mathbf{U}_i$ and $\mathbf{C}_i$.

Clearly, the final hybrid experiment does not depend on the adversary's choice of challenge message, so the proof is complete. □

3.2 Testing an Encryption Cycle

Lemma 3.5. *For a sufficiently large* $q = \tilde{O}(n^{3(k^2-1)/2})$*, the* Test *algorithm accepts with all but negligible probability when given an encryption* k*-cycle, i.e., in Game 0 of Definition 2.3.*

Remark 3.6. The lemma and its proof easily adapt to the case where Test is given a k'-cycle for $2 \leq k' \leq k$, as described in Remark 3.2. This is because the matrices $\mathbf{S}'_i$ commute with each other under multiplication, and the error terms are no larger in size and number.

Proof. We have $((i, \mathbf{A}_i), \mathbf{R}_i) \leftarrow \mathsf{Gen}(i, \bar{\mathbf{A}})$ and $\mathbf{C}_i \leftarrow \mathsf{Enc}((i, \mathbf{A}_i), \mathbf{R}_{i-1})$ for each $i \in \mathbb{Z}_k$, where all arithmetic in the subscripts is modulo k. Notice that when encrypting secret key $\mathbf{R}_{i-1}$ to produce $\mathbf{C}_i$, the encryption algorithm performs the $\mathbf{A}^{-1}$ operation for $\mathbf{A} = \mathbf{A}_{i-1}$. We therefore have

$$\begin{aligned} \mathbf{C}_i &\leftarrow \mathbf{A}_{i-1}^{-1}\big[\mathbf{S}'_i \cdot \mathbf{A}_i + \mathbf{E}_i\big] \in \mathbb{Z}^{M \times M} \\ \text{where} \quad \mathbf{S}'_i &= (\mathbf{I}_{n^i} \otimes \mathbf{S}_i \otimes \mathbf{I}_{n^{k-i-1}}) \\ &= (\underbrace{\mathbf{I}_n \otimes \cdots \otimes \mathbf{I}_n}_{i \text{ terms}} \otimes \mathbf{S}_i \otimes \underbrace{\mathbf{I}_n \otimes \cdots \otimes \mathbf{I}_n}_{k-i-1 \text{ terms}}) \in \mathbb{Z}^{N \times N} \end{aligned}$$

for some error matrices $\mathbf{S}_i, \mathbf{E}_i$. Notice that because each $\mathbf{S}_i$ appears in a different position in its tensor product, the mixed-product property implies that the matrices $\mathbf{S}'_i$ commute with each other under multiplication, i.e.,

$$\mathbf{S}'_i \cdot \mathbf{S}'_j = \mathbf{S}'_j \cdot \mathbf{S}'_i.$$

Now observe that in Equation (3.1), the minuend (left-hand term) of the difference expands as

$$\begin{aligned} \mathbf{L} &:= \mathbf{A}_{k-1} \cdot \mathbf{A}_{k-1}^{-1}[\mathbf{S}'_0 \cdot \mathbf{A}_0 + \mathbf{E}_0] \cdot \mathbf{C}_1 \cdots \mathbf{C}_{k-1} \\ &\approx \mathbf{S}'_0 \cdot \mathbf{A}_0 \cdot \mathbf{A}_0^{-1}[\mathbf{S}'_1 \cdot \mathbf{A}_1 + \mathbf{E}_1] \cdot \mathbf{C}_2 \cdots \mathbf{C}_{k-1} && (\text{error } \mathbf{E}_0 \cdot \mathbf{C}_1 \cdots \mathbf{C}_{k-1}) \\ &\approx \mathbf{S}'_0 \cdot \mathbf{S}'_1 \cdot \mathbf{A}_1 \cdot \mathbf{A}_1^{-1}[\mathbf{S}'_2 \cdot \mathbf{A}_2 + \mathbf{E}_2] \cdot \mathbf{C}_3 \cdots \mathbf{C}_{k-1} && (\text{error } \mathbf{S}'_0 \cdot \mathbf{E}_1 \cdot \mathbf{C}_2 \cdots \mathbf{C}_{k-1}) \\ &\cdots \\ &\approx \mathbf{S}'_0 \cdots \mathbf{S}'_{k-1} \cdot \mathbf{A}_{k-1}. && (\text{error } \mathbf{S}'_0 \cdots \mathbf{S}'_{k-2} \cdot \mathbf{E}_{k-1}) \end{aligned}$$

(We analyze the error terms below.) Similarly, the subtrahend (right-hand term) of the difference expands in the same way as

$$\begin{aligned} \mathbf{R} := \mathbf{A}_0 \cdot \mathbf{C}_1 \cdots \mathbf{C}_{k-1} \cdot \mathbf{C}_0 &\approx \mathbf{S}'_1 \cdots \mathbf{S}'_{k-1} \cdot \mathbf{S}'_0 \cdot \mathbf{A}_0 \\ &= \mathbf{S}'_0 \cdot \mathbf{S}'_1 \cdots \mathbf{S}'_{k-1} \cdot \mathbf{A}_0, \end{aligned}$$

with error terms as in the previous expansion, but with all the subscripts incremented (modulo k). Finally, observe that

$$(\mathbf{L} - \mathbf{R}) \cdot \bar{\mathbf{I}} \approx \mathbf{S}'_0 \cdots \mathbf{S}'_{k-1} \cdot (\mathbf{A}_{k-1} - \mathbf{A}_0) \cdot \bar{\mathbf{I}} = \mathbf{0},$$

where the approximation includes the errors (times $\bar{\mathbf{I}}$) from both of the above expansions.

It remains analyze the error terms from the above expansions. Recall that each $\mathbf{E}_i$ and $\mathbf{S}_i$ is made up of independent entries drawn from χ, which is subgaussian with parameter $O(\sqrt{n})$. Similarly, by Item 4 of Proposition 2.7, every $\mathbf{C}_i$ has independent subgaussian columns with parameter $\tilde{O}(M)$. Therefore, by Proposition 2.1,

$$s_1(\mathbf{E}_i) = O(\sqrt{nM}), \quad s_1(\mathbf{S}'_i) = s_1(\mathbf{S}_i) = O(n), \quad s_1(\mathbf{C}_i) = \tilde{O}(M^{3/2})$$

except with negligible probability. It follows that in the analysis of $\mathbf{L}, \mathbf{R}$ above, the spectral norm of each error matrix—and thereby the magnitude of every entry—is bounded by $\tilde{O}(n^{1/2} \cdot M^{3k/2-1})$. Taking a sufficiently large $q = \tilde{O}(n^{3(k^2-1)/2})$ ensures that every entry in the sum of the error matrices has magnitude less than $q/4$, so the tester accepts. □

3.3 Testing a Non-Cycle

Lemma 3.7. *Under the LWE assumption from Theorem 3.3, the* Test *algorithm accepts with only negligible probability when given ciphertexts that all encrypt zero, i.e., in Game 1 of Definition 2.3.*

Proof. We consider the following sequence of hybrid experiments for generating the tester's input. We show that successive hybrids are indistinguishable (either computationally or statistically), which implies that the tester's acceptance probability differs by only a negligible amount in successive hybrids. Moreover, we show that its acceptance probability in the final hybrid is exponentially small, which proves the claim.

Hybrid 1: here the public keys $\mathbf{A}_i$ are uniformly random and independent (modulo their common prefix $\bar{\mathbf{A}}$), and each ciphertext $\mathbf{C}_i$ is independently sampled from D^M.

Following the proof of Lemma 3.4, this experiment is computationally indistinguishable from the real one (under the LWE assumption), and hence the tester's acceptance probability is only negligibly different in the two experiments.

Hybrids 2, 3, . . . , $k+1$: in hybrid 2, in the cycle-test algorithm (Equation (3.1)) we replace $\mathbf{A}_{k-1} \cdot \mathbf{C}_0$ with a uniformly random $\mathbf{A}'_0$, and similarly replace $\mathbf{A}_0 \cdot \mathbf{C}_1$ with a uniformly random $\mathbf{A}'_1$ (both independent of everything else). Hybrids 3 through $k+1$ are defined similarly, so that the final cycle-test algorithm simply tests whether $(\mathbf{A}'_{k-1} - \mathbf{A}'_0) \cdot \bar{\mathbf{I}} \in (-q/4, q/4) \pmod q$ for uniformly random and independent $\mathbf{A}'_{k-1}, \mathbf{A}'_0$. Clearly, this test accepts with probability bounded by the negligible quantity $2^{-N \cdot \bar{M}} \leq 2^{-n}$.

We claim that each of these hybrids is within negligible statistical distance of the previous one. For Hybrid 2 this follows by Item 1 of Proposition 2.7: because $\mathbf{A}_{k-1}, \mathbf{A}_0$ are uniformly random, and $\mathbf{C}_0, \mathbf{C}_1$ are independent, $\mathbf{A}_{k-1} \cdot \mathbf{C}_0$ and $\mathbf{A}_0 \cdot \mathbf{C}_1$ are negligibly far from uniformly random and independent. (This is where we use the fact that $k \geq 2$.) The same argument applies for subsequent hybrids. This completes the proof. □

4 Ring-LWE Construction

In this section we present a k-cycle tester that is IND-CPA secure assuming the hardness of ring-LWE (RLWE), appropriately parameterized. The construction works very similarly to the plain LWE one from Section 4. However, it is not limited to constant $k = O(1)$, but can be instantiated for any $k = \text{poly}(\lambda)$, because it does not use the tensoring technique. The scheme involves the following parameters:

- the ring $R = \mathbb{Z}[X]/(X^n + 1)$ for power-of-two n, the standard ring-LWE error distribution χ over R, and an integer modulus q (which we instantiate below);
- $\bar{m} < m = \tilde{O}(\log q)$, where $\bar{m}, m$ are the dimensions associated with the ring-based $\mathsf{GenTrap}$ for parameters R, q;
- The secret-key and message spaces are both $\mathcal{R}$, the randomness/trapdoor space of the ring-based $\mathsf{GenTrap}$.

The construction is as follows.

- $\mathsf{Setup}()$: output a uniformly random $\bar{\mathbf{a}} \in R_q^{\bar{m}}$.

- $\mathsf{Gen}(\bar{\mathbf{a}})$: let $\mathbf{a} \leftarrow \mathsf{GenTrap}(\bar{\mathbf{a}}; \mathbf{R})$ for $\mathbf{R} \leftarrow \mathcal{R}$. Output $\mathbf{a}$ as the public key and the trapdoor $\mathbf{R}$ as the secret key.
- $\mathsf{Enc}(\mathbf{a}, \mathbf{R} \in \mathcal{R})$: let $\mathbf{v} \leftarrow \mathsf{GenTrap}(\bar{\mathbf{a}}; \mathbf{R})$ where $\mathbf{v} \in R_q^m$. Choose $s \leftarrow \chi$ and $\mathbf{e} \leftarrow \chi^m$. Output the ciphertext
$$\mathbf{C} \leftarrow \mathbf{v}^{-1}[s \cdot \mathbf{a} + \mathbf{e}] \in R^{m \times m},$$
where the $\mathbf{v}^{-1}$ operation is performed using the trapdoor $\mathbf{R}$.
- $\mathsf{Test}((\mathbf{a}_i, \mathbf{C}_i)_{i \in \mathbb{Z}_k})$: Given public keys $\mathbf{a}_i$ and ciphertexts $\mathbf{C}_i$, check whether
$$(\mathbf{a}_{k-1} \cdot \mathbf{C}_0 \cdot \mathbf{C}_1 \cdots \mathbf{C}_{k-1} - \mathbf{a}_0 \cdot \mathbf{C}_1 \cdots \mathbf{C}_{k-1} \cdot \mathbf{C}_0) \cdot \bar{\mathbf{I}} \in \mathcal{Q}^{\bar{m}} \pmod{q}, \tag{4.1}$$
where $\bar{\mathbf{I}} = \left(\begin{smallmatrix} \mathbf{I}_{\bar{m}} \\ \mathbf{0} \end{smallmatrix}\right) \in R^{m \times \bar{m}}$, and $\mathcal{Q} \subseteq R$ is the set of ring elements whose coefficients (with respect to the standard power basis) all are in $(-q/4, q/4)$.

In the remainder of this section we prove the following theorem:

Theorem 4.1. *For any $k = \mathrm{poly}(\lambda)$ and a sufficiently large $q = \tilde{O}(nk)^{O(k)}$, the above scheme is a k'-cycle tester for $2 \leq k' \leq k$, assuming the hardness of decision-RLWE$_{R,q,\chi,m}$.*

Recall that the Ring-LWE instantiation from Theorem 4.1 is at least as hard as (quantumly) approximating certain lattice problems on ideal lattices in R, in the worst case, to within $\tilde{O}(nk)^{O(k)}$ factors.

In Section 4.1 below we prove IND-CPA security, in Section 4.2 we show that Test almost always accepts on an encryption cycle, and in Section 4.3 we show that Test almost never accepts on a non-cycle. Together these prove Theorem 4.1.

4.1 Security

Lemma 4.2. *The tuple* $(\mathsf{Setup}, \mathsf{Gen}, \mathsf{Enc})$ *is IND-CPA secure under the RLWE assumption from Theorem 4.1.*

Proof. To prove that the scheme satisfies IND-CPA security, we consider the following sequence of hybrids. We show that adjacent hybrids are indistinguishable, either computationally or statistically.

Hybrid 1: Here we generate the public key $\mathbf{a}$ uniformly at random, instead of by using $\mathsf{GenTrap}$. By Item 2 of Proposition 2.12, this game is within negligible statistical distance of the real game.

Hybrid 2: Here we generate the ciphertext $\mathbf{C} = \mathbf{v}^{-1}[\mathbf{b}]$ using a uniformly random $\mathbf{b} \leftarrow R_q^m$, rather than using $\mathbf{b} = s \cdot \mathbf{a} + \mathbf{e}$.

We claim that this game is computationally indistinguishable from the previous one, under the RLWE assumption. This follows by a straightforward reduction: given input $(\mathbf{a}, \mathbf{b}) \in R_q^m \times R_q^m$, the reduction sets the public key as $\mathbf{a}$, and then computes the ciphertext $\mathbf{C} \leftarrow \mathbf{v}^{-1}[\mathbf{b}]$ using the adversary's challenge message to define $\mathbf{v}$ and compute the $\mathbf{v}^{-1}[\cdot]$ operation (using the $\mathsf{SamplePre}$ algorithm) in the usual way.

Hybrid 3: Here we draw the ciphertext matrix from D^m, instead of by invoking $\mathbf{v}^{-1}$. By Item 3 of Proposition 2.12, this hybrid is statistically indistinguishable from the previous one, for any choice of the challenge message.

Because the final hybrid does not depend on the challenge message, we have the desired result. □

4.2 Testing an Encryption Cycle

Lemma 4.3. *For a sufficiently large $q = \tilde{O}(nk)^{O(k)}$, the* Test *algorithm accepts with all but negligible probability when given an encryption k-cycle, i.e., in Game 0 of Definition 2.3.*

Proof. For input $(\mathbf{a}_i, \mathbf{C}_i)_{i \in \mathbb{Z}_k}$, we have

$$\mathbf{C}_i \leftarrow \mathbf{a}_{i-1}^{-1}[s_i \cdot \mathbf{a}_i + \mathbf{e}_i]$$

for some $s_i \leftarrow \chi$ and $\mathbf{e}_i \leftarrow \chi^m$. Moreover, by commutativity of R we have $s_i s_j = s_j s_i$ for any $i, j \in \mathbb{Z}_k$. Now for the left-hand term of Equation (4.1) we have

$$\begin{aligned}
\mathbf{l} &:= \mathbf{a}_{k-1} \cdot \mathbf{a}_{k-1}^{-1}[s_0 \cdot \mathbf{a}_0 + \mathbf{e}_0] \cdot \mathbf{C}_1 \cdots \mathbf{C}_{k-1} \\
&\approx s_0 \cdot \mathbf{a}_0 \cdot \mathbf{a}_0^{-1}[s_1 \cdot \mathbf{a}_1 + \mathbf{e}_1] \cdot \mathbf{C}_2 \cdots \mathbf{C}_{k-1} && (\text{error } \mathbf{e}_0 \cdot \mathbf{C}_1 \cdots \mathbf{C}_{k-1}) \\
&\approx s_0 \cdot s_1 \cdot \mathbf{a}_1 \cdot \mathbf{a}_1^{-1}[s_2 \cdot \mathbf{a}_2 + \mathbf{e}_2] \cdot \mathbf{C}_3 \cdots \mathbf{C}_{k-1} && (\text{error } s_0 \cdot \mathbf{e}_1 \cdot \mathbf{C}_2 \cdots \mathbf{C}_{k-1}) \\
&\cdots \\
&\approx s_0 \cdots s_{k-1} \cdot \mathbf{a}_{k-1}. && (\text{error } s_0 \cdots s_{k-2} \cdot \mathbf{e}_{k-1})
\end{aligned}$$

(We analyze the error terms below.) Similarly, for the right-hand term of Equation (4.1), we have

$$\begin{aligned}
\mathbf{r} := \mathbf{a}_0 \cdot \mathbf{C}_1 \cdots \mathbf{C}_{k-1} \cdot \mathbf{C}_0 &\approx s_1 \cdots s_{k-1} \cdot s_0 \cdot \mathbf{a}_0 \\
&= s_0 \cdots s_{k-1} \cdot \mathbf{a}_0,
\end{aligned}$$

with error terms as in the previous expansion, but with all the subscripts incremented (modulo k). Therefore,

$$(\mathbf{l} - \mathbf{r}) \cdot \bar{\mathbf{I}} \approx s_0 \cdots s_{k-1} \cdot (\mathbf{a}_{k-1} - \mathbf{a}_0) \cdot \bar{\mathbf{I}} = \mathbf{0},$$

where the approximation includes the errors from the expansions of both $\mathbf{l}$ and $\mathbf{r}$, and where we use the fact that $\mathbf{a}_i \cdot \bar{\mathbf{I}} = \bar{\mathbf{a}}$ for every $i \in \mathbb{Z}_k$.

It remains to analyze the error terms. Recall that each $\mathbf{e}_i$ is made up of independent entries from χ. Also, each secret s_i comes from χ. Lastly, each ciphertext $\mathbf{C}_i \in \mathbb{R}^{m \times m}$ is drawn as some $\mathbf{a}^{-1}[\cdot]$. Then by Equation (2.1), Proposition 2.9, and Item 4 of Proposition 2.12, we have (except with negligible probability)

$$s_1(s_i) \leq n^c, \quad s_1(\mathbf{e}_i) \leq \sqrt{m} \cdot n^c, \quad s_1(\mathbf{C}_i) \leq m \cdot n^c$$

for some universal constant $c > 1$. Let $\mathbf{e}$ denote the sum of all the error terms in the above approximations for $\mathbf{l}, \mathbf{r}$. We have

$$\|\mathbf{e}\|_\infty \leq s_1(\mathbf{e}) \leq 2k \cdot m^{k-1} \cdot n^{ck}.$$

Because $m = \tilde{O}(\log q)$, for a sufficiently large $q = \tilde{O}(nk)^{O(k)}$, Proposition 2.10 guarantees that every coefficient of every entry of $\mathbf{e}$ has the magnitude less than $q/4$, and therefore $\mathbf{e} \in \mathcal{Q}^m$ and Test accepts, as desired. □

4.3 Testing a Non-Cycle

Lemma 4.4. *Under the same RLWE assumption from Theorem 4.1, for $k \geq 2$ the* Test *algorithm accepts with only negligible probability on ciphertexts that all encrypt zero, i.e., in Game 1 of Definition 2.3.*

Proof. We consider the following sequence of hybrids. We show that adjacent hybrids are indistinguishable, either computationally or statistically. Hence, the tester's acceptance probability differs by only a negligible amount in successive hybrids.

Hybrid 1: In this hybrid, the public keys are uniformly random and independent (modulo their common prefix $\bar{\mathbf{a}}$), and each ciphertext is sampled independently from D^m. Following the proof of Lemma 4.2, this hybrid is computationally indistinguishable from real game.

Hybrids 2, 3, . . . , $k+1$: In the second hybrid, in Equation (4.1) we replace $\mathbf{a}_{k-1} \cdot \mathbf{C}_0$ with a uniformly random $\mathbf{a}'_0$ and replace $\mathbf{a}_0 \cdot \mathbf{C}_1$ with a uniformly random $\mathbf{a}'_1$. We define hybrids 3 through $k+1$ similarly. Hence, the final algorithm tests whether $(\mathbf{a}'_{k-1} - \mathbf{a}'_0) \cdot \bar{\mathbf{I}} \in \mathcal{Q}^{\bar{m}}$, where both terms in the difference are uniformly random and independent. The acceptance probability is therefore bounded by 2^{-n}.

Statistical indistinguishability of each of these hybrids from the previous one follows by Item 1 of Proposition 2.12. Therefore, the algorithm rejects on non-cycles with high probability, and proof is complete. □

References

[ABBC10] T. Acar, M. Belenkiy, M. Bellare, and D. Cash. Cryptographic agility and its relation to circular encryption. In *EUROCRYPT*, pages 403–422. 2010.

[ABHS05] P. Adão, G. Bana, J. Herzog, and A. Scedrov. Soundness of formal encryption in the presence of key-cycles. In *ESORICS*, pages 374–396. 2005.

[ACPS09] B. Applebaum, D. Cash, C. Peikert, and A. Sahai. Fast cryptographic primitives and circular-secure encryption based on hard learning problems. In *CRYPTO*, pages 595–618. 2009.

[AP12] J. Alperin-Sheriff and C. Peikert. Circular and KDM security for identity-based encryption. In *Public Key Cryptography*, pages 334–352. 2012.

[App11] B. Applebaum. Key-dependent message security: Generic amplification and completeness. In *EUROCRYPT*, pages 527–546. 2011.

[BG10] Z. Brakerski and S. Goldwasser. Circular and leakage resilient public-key encryption under subgroup indistinguishability - (or: Quadratic residuosity strikes back). In *CRYPTO*, pages 1–20. 2010.

[BGK11] Z. Brakerski, S. Goldwasser, and Y. T. Kalai. Black-box circular-secure encryption beyond affine functions. In *TCC*, pages 201–218. 2011.

[BHHI10] B. Barak, I. Haitner, D. Hofheinz, and Y. Ishai. Bounded key-dependent message security. In *EUROCRYPT*, pages 423–444. 2010.

[BHHO08] D. Boneh, S. Halevi, M. Hamburg, and R. Ostrovsky. Circular-secure encryption from decision Diffie-Hellman. In *CRYPTO*, pages 108–125. 2008.

[BHW15] A. Bishop, S. Hohenberger, and B. Waters. New circular security counterexamples from decision linear and learning with errors. In *ASIACRYPT 2015*, pages 776–800. 2015.

[BLP$^+$13] Z. Brakerski, A. Langlois, C. Peikert, O. Regev, and D. Stehlé. Classical hardness of learning with errors. In *STOC*, pages 575–584. 2013.

[BRS02] J. Black, P. Rogaway, and T. Shrimpton. Encryption-scheme security in the presence of key-dependent messages. In *Selected Areas in Cryptography*, pages 62–75. 2002.

[CGH12] D. Cash, M. Green, and S. Hohenberger. New definitions and separations for circular security. In *PKC 2012*, pages 540–557. 2012.

[CL01] J. Camenisch and A. Lysyanskaya. An efficient system for non-transferable anonymous credentials with optional anonymity revocation. In *EUROCRYPT*, pages 93–118. 2001.

[Gen09a] C. Gentry. *A fully homomorphic encryption scheme*. Ph.D. thesis, Stanford University, 2009. `http://crypto.stanford.edu/craig`.

[Gen09b] C. Gentry. Fully homomorphic encryption using ideal lattices. In *STOC*, pages 169–178. 2009.

[GGH$^+$13] S. Garg, C. Gentry, S. Halevi, M. Raykova, A. Sahai, and B. Waters. Candidate indistinguishability obfuscation and functional encryption for all circuits. In *FOCS*, pages 40–49. 2013.

[GM82] S. Goldwasser and S. Micali. Probabilistic encryption. *J. Comput. Syst. Sci.*, 28(2):270–299, 1984. Preliminary version in STOC 1982.

[GPV08] C. Gentry, C. Peikert, and V. Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *STOC*, pages 197–206. 2008.

[KRW15] V. Koppula, K. Ramchen, and B. Waters. Separations in circular security for arbitrary length key cycles. In *TCC*, pages 378–400. 2015.

[KW16] V. Koppula and B. Waters. Circular security separations for arbitrary length cycles from LWE. In *CRYPTO*. 2016. To appear.

[LPR10] V. Lyubashevsky, C. Peikert, and O. Regev. On ideal lattices and learning with errors over rings. *Journal of the ACM*, 60(6):43:1–43:35, November 2013. Preliminary version in Eurocrypt 2010.

[LPR13] V. Lyubashevsky, C. Peikert, and O. Regev. A toolkit for ring-LWE cryptography. In *EUROCRYPT*, pages 35–54. 2013.

[MO14] A. Marcedone and C. Orlandi. Obfuscation ($\to$) (IND-CPA security ! $\to$ circular security). In *SCN*, pages 77–90. 2014.

[MP12] D. Micciancio and C. Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. In *EUROCRYPT*, pages 700–718. 2012.

[MTY11] T. Malkin, I. Teranishi, and M. Yung. Efficient circuit-size independent public key encryption with KDM security. In *EUROCRYPT*, pages 507–526. 2011.

[Pei09] C. Peikert. Public-key cryptosystems from the worst-case shortest vector problem. In *STOC*, pages 333–342. 2009.

[Reg05] O. Regev. On lattices, learning with errors, random linear codes, and cryptography. *J. ACM*, 56(6):1–40, 2009. Preliminary version in STOC 2005.

[Ver12] R. Vershynin. *Compressed Sensing, Theory and Applications*, chapter 5, pages 210–268. Cambridge University Press, 2012. Available at `http://www-personal.umich.edu/~romanv/papers/non-asymptotic-rmt-plain.pdf`.