Counterexamples to New Circular Security Assumptions Underlying iO

Sam Hopkins*, Aayush Jain†, and Huijia Lin‡

Abstract. We study several strengthening of classical circular security assumptions which were recently introduced in four new lattice-based constructions of indistinguishability obfuscation: Brakerski-Döttling-Garg-Malavolta (Eurocrypt 2020), Gay-Pass (STOC 2021), Brakerski-Döttling-Garg-Malavolta (Eprint 2020) and Wee-Wichs (Eprint 2020).

We provide explicit counterexamples to the 2-*circular shielded randomness leakage* assumption w.r.t. the Gentry-Sahai-Waters fully homomorphic encryption scheme proposed by Gay-Pass, and the *homomorphic pseudorandom LWE samples* conjecture proposed by Wee-Wichs. Our work suggests a separation between classical circular security of the kind underlying un-levelled fully-homomorphic encryption from the strengthened versions underlying recent iO constructions, showing that they are not (yet) on the same footing.

Our counterexamples exploit the flexibility to choose specific implementations of circuits, which is explicitly allowed in the Gay-Pass assumption and unspecified in the Wee-Wichs assumption. Their indistinguishabilty obfuscation schemes are still unbroken. Our work shows that the assumptions, at least, need refinement. In particular, generic leakage-resilient circular security assumptions are delicate, and their security is sensitive to the specific structure of the leakages involved.

1 Introduction

Indistinguishability obfuscation ($i\mathcal{O}$) for general programs computable in polynomial time [7] enables turning programs into unintelligible ones while preserving their functionality. $i\mathcal{O}$ is a fundamental primitive and has found many applications in cryptography and beyond. As such, it is extremely important to base the feasibility of $i\mathcal{O}$ on simple and well-studied hardness assumptions, and to thoroughly understand the objects and assumptions that imply $i\mathcal{O}$. Current constructions of $i\mathcal{O}$ can be broadly categorized into two schools: those using multilinear or bilinear pairing, and those without pairing. Very recently, we have seen exciting advances on both fronts. Using pairing, Jain, Lin, and Sahai [32] constructed $i\mathcal{O}$ from four well-studied assumptions: Learning With Errors (LWE) [41], Decisional Linear assumption (DLIN) [6] over bilinear maps,

*UC Berkeley. Email: hopkins@berkeley.edu.

†UCLA, Center for Encrypted Functionalities, and NTT Research. Email: aayushjain@cs.ucla.edu.

‡UW. Email: rachel@cs.washington.edu.

Learning Pairity with Noise over general fields [29], and Pseudo-Random Generators in NC_0 [25]. Without pairing, three works [21,11,44], following [10], based $i\mathcal{O}$ on new types of circular security assumptions on integer lattices.

In this work, we focus on these recent constructions [10,21,11,44] and the new circular security assumptions they are based on. These constructions are very interesting because of their novel approaches and distinctive features. First, they are built solely on integer lattices (instead of drawing hardness from multiple cryptosystems) and therefore are possibly secure against quantum attacks. Second, their security assumptions are similar in flavor to the classical circular security heuristic [16,9], which by now has been extensively studied and widely applied, most notably to un-leveled Fully Homomorphic Encryption (FHE) using Gentry's boostrapping mechanism [22].

At the same time, the new assumptions are stronger than classical circular security in non-trivial ways. Consider the Gay-Pass assumption. Classical circular security w.r.t. a public key encryption scheme postulates that it is Chosen-Message-Attack (CPA) secure, even in the presence of an encrypted key-cycle that possibly uses other encryption schemes. The Gay-Pass assumption generalizes this blueprint to consider *leakage-resilient CPA security*: it says that if an encryption scheme is CPA secure when the adversary has access to certain leakage on the randomness of encryption, then additionally publishing an encrypted key-cycle should not harm this leakage-resilient CPA security. Concretely, their $i\mathcal{O}$ scheme assumes *Shielded Randomness Leakage (SRL) resilience* in the presence of *2-circular encryption*, w.r.t. the Gentry-Sahai-Waters FHE scheme [24] and a Packed version of Regev's encryption [41,40][1]. The work [11] proposes a variant of the Gay-Pass assumption with a key-randomness cycle. Wee and Wichs [44] take a different approach and construct $i\mathcal{O}$ based on LWE and a new conjecture, *Homomorphic Pseudorandom LWE Samples* (HPLS). Though this conjecture does not directly follow the circular security blueprint, close examination reveals a circular security flavor, involving the dual-GSW homomorphic commitment [24,26] and a Pseudo-Random Function (PRF).

Although stronger and more complex than the classical circular security, these new assumptions were formulated in a principled way – indeed, on the surface, they seem to place $i\mathcal{O}$ on qualitatively similar footing as un-leveled FHE! While exciting and encouraging, when it comes to new assumptions, it is important to be cautious and imperative to conduct cryptanalysis to develop deeper understandings. That is the purpose of our work.

Our Results We present counterexamples to the Gay-Pass and Wee-Wichs assumptions. In both cases, we consider the GSW FHE scheme and the dual-GSW homomorphic commitment scheme for evaluating *arithmetic circuits* consisting of arithmetic addition, multiplication, and multiplication by constant gates. We stress that both schemes natively support these arithmetic operations [24]. In particular, in our counterexample to the Wee-Wichs conjecture we will leverage multiplication by a large constant, $2^{-1} \bmod p$ (which is not needed for the counterexample to Gay-Pass assumption).

[1] Or alternatively, the Damgård-Jurik encryption [18,39].

– First, we show that the Gay-Pass assumption is false when instantiated with the GSW FHE scheme by presenting a concrete attack.
– Second, Wee and Wichs's HPLS conjecture is parameterized with a sampling algorithm D that takes random coins τ and produces a random LWE secret $\boldsymbol{s} \leftarrow \mathbb{Z}_p^n$ and an error vector $\boldsymbol{e}$ according to some error distribution. We show the conjecture is sensitive to the circuit implementation of D, namely, for every D, there is an arithmetic circuit C_{D} implementing it such that the HPLS conjecture instantiated with C_{D} is false. Again, we present a concrete attack.

Notably, classical circular security plausibly holds w.r.t. both the modified GSW and dual GSW schemes. Hence, our work gives the first examples that separate classical circular security and the strengthened versions of circular security underlying recent $i\mathcal{O}$ schemes, showing evidence that they are not (yet) on the same footing.

Our counterexamples exploit some flexibility in the implementation details of the Gay-Pass and Wee-Wichs assumptions. The choice of such implementation is explicitly given to the adversary in the Gay-Pass assumption and is left unspecified in the Wee-Wichs conjecture. It remains possible that other choices of implementation of circuits do result in an unbroken assumption. Nevertheless, our work shows that this will, at least, require refinement of the assumptions, and in particular that generic circular security assumptions/definitions are delicate, and their security is actually sensitive to the specific structure of the leakages involved.

Next, we describe the Gay-Pass and Wee-Wichs assumptions and our counterexamples in more detail.

Counterexample to the Gay-Pass assumption. As stated in Gay and Pass [21], the *2-circular assumption* w.r.t. two public key encryption schemes Enc^1 and Enc^2 that are Chosen-Plaintext-Attack (CPA) secure postulates that

– *Classical 2-circular security assumption w.r.t.* $\mathsf{Enc}^1, \mathsf{Enc}^2$*:* Enc^1 is (still) CPA secure – that is, honestly generated ciphertexts $\mathsf{Enc}^1_{\mathsf{pk}^1}(m^0)$ and $\mathsf{Enc}^1_{\mathsf{pk}^1}(m^1)$ for any two chosen messages m^0, m^1 are indistinguishable – when a length-two encrypted key cycle $\mathsf{Enc}^1_{\mathsf{pk}^1}(\mathsf{sk}^2)$, $\mathsf{Enc}^2_{\mathsf{pk}^2}(\mathsf{sk}^1)$ is published.

Classical circular security has been extensively studied as encrypted key cycles of different lengths naturally arise in applications such as encrypted storage system, anonymous credentials [16], and un-leveled FHE [22]. So far, though counterexamples to 2-circular security or 1-circular security for bit encryption[2] (where the key cycle has length 1 $\{\mathsf{Enc}_{\mathsf{pk}}(\mathsf{sk}_i)\}_{i\in[|\mathsf{sk}|]}$) have been constructed (see e.g. [1,28,17,43,36,33,8,34,27,45]), no attacks have been shown against any "natural" encryption schemes. Therefore, classical circular security is still commonly assumed w.r.t. natural encryption schemes such as homomorphic encryption [24,12,15], Regev's encryption [41] etc.

[2]Crafting a counterexample for 1-circular security for string encryption is trivial.

Gay and Pass extend 2-circular security to consider CPA security in the presence of the so-called shielded randomness leakage (SRL). More specifically, shielded randomness leakage is only defined w.r.t. FHE schemes with certain properties including randomness homomorphism. The leakage is captured by an oracle $\mathcal{O}_{\mathsf{SRL}}$ (described shortly below) and reveals certain information of the randomness of encryption. A public-key FHE scheme Enc^1 is *SRL-secure* if CPA security holds even if the adversary has access to $\mathcal{O}_{\mathsf{SRL}}$. Then the 2-circular SRL security assumption w.r.t. $\mathsf{Enc}^1, \mathsf{Enc}^2$ where Enc^1 is SRL secure and Enc^2 is CPA secure, states that:

- *2-circular SRL security assumption w.r.t.* $\mathsf{Enc}^1, \mathsf{Enc}^2$*:* Enc^1 is (still) SRL secure – that is, honestly generated ciphertexts $\mathsf{Enc}^1_{\mathsf{pk}^1}(m^0)$ and $\mathsf{Enc}^1_{\mathsf{pk}^1}(m^1)$ for any two chosen messages m^0, m^1 are indistinguishable, even if the adversary has access to $\mathcal{O}_{\mathsf{SRL}}$ – when a length two encrypted key cycle $\mathsf{Enc}^1_{\mathsf{pk}^1}(\mathsf{sk}^2)$, $\mathsf{Enc}^2_{\mathsf{pk}^2}(\mathsf{sk}^1)$ is published.

The Gay-Pass $i\mathcal{O}$ scheme relies on the above assumption w.r.t. the GSW FHE scheme as Enc^1 and the packed Regev encryption as Enc^2. Notably, they prove that the GSW scheme is SRL-secure based on LWE.

Let's now understand what shielded randomness leakage is. In the plain SRL security game (without encrypted key cycles), the adversary is given a collection of challenge ciphertexts $\{\mathsf{ct}_i = \mathsf{Enc}^1_{\mathsf{pk}^1}(m_i^b; \mathbf{R}_i)\}_i$ encrypting one of the two sets of chosen messages, $\{m_i^0\}_i$ or $\{m_i^1\}_i$, for a random b, using randomness $\{\mathbf{R}_i\}_i$. In addition, the adversary $\mathcal{A}$ can interact with the SRL oracle $\mathcal{O}_{\mathsf{SRL}}$ as follows to help it distinguish.

- *The* $\mathcal{O}_{\mathsf{SRL}}$ *Oracle (Simplified)* gives leakage on the message and randomness $\{m_i^b; \mathbf{R}_i\}_i$ underlying the challenge ciphertexts as follows:
  1. Upon invocation, $\mathcal{O}_{\mathsf{SRL}}$ samples a fresh encryption $\mathsf{ct}^\star = \mathsf{Enc}^1_{\mathsf{pk}^1}(0; \mathbf{R}^\star)$ of zero using randomness $\mathbf{R}^\star$ and sends $\mathsf{ct}^\star$ to the adversary[3].
  2. $\mathcal{A}$ chooses a circuit C and an output y.
  3. $\mathcal{O}_{\mathsf{SRL}}$ homomorphically evaluates C on $\mathsf{ct}^\star$ and the challenge ciphertexts $\{\mathsf{ct}_i\}_i$ to obtain an output ciphertext $\mathsf{ct}_C = \mathsf{HEval}(C, \mathsf{ct}^\star, \{\mathsf{ct}_i\})$ that encrypts y' with randomness $\mathbf{R}_C$ (computed by the randomness homomorphism property of HE from $\{m_i^b; \mathbf{R}_i\}_i$). It returns $\mathbf{R}^\star - \mathbf{R}_C$ if $y = y'$, or nothing if $y \neq y'$.

In the *2-circular* SRL-security game, the adversary is additionally given an encrypted key cycle $\mathsf{Enc}^1_{\mathsf{pk}^1}(\mathsf{sk}^2)$, $\mathsf{Enc}^2_{\mathsf{pk}^2}(\mathsf{sk}^1)$ along with the challenge ciphertexts $\{\mathsf{ct}_i\}$ at the beginning. We remark that for security of the ensuing Gay-Pass $i\mathcal{O}$ construction it is crucial that the adversary is allowed to choose C adaptively. This means in the plain SRL security game, C may depend on $\mathsf{ct}^\star, \{\mathsf{ct}_i\}$, and, in the 2-circular SRL security game, additionally on the encrypted cycle $\mathsf{Enc}^1_{\mathsf{pk}^1}(\mathsf{sk}^2)$, $\mathsf{Enc}^2_{\mathsf{pk}^2}(\mathsf{sk}^1)$. Indeed, the security reduction from $i\mathcal{O}$ to the 2-circular

[3] More concretely for the GSW scheme, this encryption of zero is extra noisy, meaning the magnitude of entries of $\mathbf{R}^\star$ is large enough to smudge entries of $\mathbf{R}_C$ below.

SRL security chooses such a "dependent" C. Looking ahead, our counterexample also crucially exploits this adaptivity.

Our counterexample: We show that the 2-circular SRL security assumption is false w.r.t. the GSW FHE scheme in [24]. Let us now give more details.

Our Ideas In a Nut shell: Given that (modified) GSW is both SRL-secure and plausibly circular secure, the attack must simultaneously leverage the shield-randomness leakage $\mathbf{R}^\star - \mathbf{R}_C$ and the encrypted key cycle $\mathsf{Enc}^1_{\mathsf{pk}^1}(\mathsf{sk}^2), \mathsf{Enc}^2_{\mathsf{pk}^2}(\mathsf{sk}^1)$. Recall that the attack can adaptively choose the circuit C depending on the key cycle, $\mathsf{ct}^\star$, and $\{\mathsf{ct}_i\}$, meaning they can be hardcoded in C. Observe also that the input to C is $(\{m_i^b\}, \mathsf{sk}^2)$, and hence C can compute as an intermediate value sk^1 and can also "access" $\mathbf{R}^\star$ (by decrypting $\mathsf{Enc}^2_{\mathsf{pk}^2}(\mathsf{sk}^1)$ and $\mathsf{ct}^\star$). Since C can "access" both $\mathbf{R}^\star$ and $\{m_i^b\}$, our attack carefully engineers C so that homomorphic evaluation of C produces an output ciphertext ct_C with randomness $\mathbf{R}_C$ correlated with $(\mathbf{R}^\star, \{m_i^b\})$, and then the shield randomness leakage $\mathbf{R}^\star - \mathbf{R}_C$ reveals information of b. More specifically, the attack creates correlation between the *parity bit of noises and values* by carefully engineering C using the following correlation-inducing *gadget circuits*.

- *Correlation Gadget:* The gadget circuit $G(x, 0)$ multiplies x with 0 and produces a fixed output of 0. Homomorphically evaluating G on GSW ciphertexts ct of x and ct_0 of 0 produces a new ciphertext $\mathsf{ct}' = \mathbf{A}\mathbf{R}'$ of zero of the following form:

$$\mathsf{ct} = \mathbf{A}\mathbf{R} + x\mathbf{G},\ \mathsf{ct}_0 = \mathbf{A}\mathbf{R}_0 \quad \xrightarrow{\mathsf{HEval},\times} \quad \mathsf{ct}' = \mathbf{A}\mathbf{R}',\ \mathbf{R}' = \mathbf{R} \cdot G^{-1}(\mathsf{ct}_0) + x\mathbf{R}_0$$

Consider an attack that chooses a circuit C which first computes $x = f(m_i^b, \mathsf{sk}^2)$ and then the above $G(x, 0)$ (f is specified shortly below). The attack receives from the SRL oracle leakage

$$\mathbf{R}^* + \mathbf{R} \cdot G^{-1}(\mathsf{ct}_0) + x\mathbf{R}_0 \ .$$

To learn the bit b, we want to *1)* correlate x with $\mathbf{R}^*$ and b, and *2)* eliminate the middle term $\mathbf{R} \cdot G^{-1}(\mathsf{ct}_0)$.

- We achieve the second by finding a vector $\boldsymbol{v} \in \{0,1\}^m$ such that $G^{-1}(\mathsf{ct}_0) \cdot \boldsymbol{v} = 0 \bmod 2$. This is possible with probability close to $1/2$ as $G^{-1}(\mathsf{ct}_0)$ is a pseudorandom binary matrix and hence is non-singular mod 2 with probability close to $1/2$.
- We achieve the first by letting the function f compute $b \cdot \boldsymbol{e}\mathbf{R}^*\boldsymbol{v} \bmod 2$. Observe that this is computable since homomorphically decrypting ct^* gives exactly $\boldsymbol{e}\mathbf{R}^*$. One can then further multiply b and $\boldsymbol{v}$, followed by modulo 2.

This means the attack can learn

$$\boldsymbol{z} = \mathbf{R}^*\boldsymbol{v} + b \cdot (\boldsymbol{e}\mathbf{R}^*\boldsymbol{v})\mathbf{R}_0\boldsymbol{v} \bmod 2 \ .$$

Let us observe the difference between the cases when $b = 0$ or 1. If $b = 0$, $\boldsymbol{z} = \mathbf{R}^*\boldsymbol{v} \bmod 2$ which is random since $\mathbf{R}^*$ is random and independent of v.

On the other hand, if $b = 0$, $\boldsymbol{z} = \mathbf{R}^*\boldsymbol{v} + (\boldsymbol{e}\mathbf{R}^*\boldsymbol{v})\mathbf{R}_0\boldsymbol{v} \bmod 2$, which satisfies $\boldsymbol{e}\cdot\boldsymbol{z} = 0 \bmod 2$ if $\boldsymbol{e}\mathbf{R}_0\boldsymbol{v} = 1$. The latter condition holds with probability $1/2$ over the random choice of $\mathbf{R}_0$. This difference is sufficient for creating a distinguishing attack: Repeat the above many times to collect different $\boldsymbol{z}_i$ w.r.t. to *different* $\mathsf{ct}_i^* = \mathbf{B}\mathbf{R}_i^*$, and the *same* $\mathsf{ct}_0 = \mathbf{B}\mathbf{R}_0$. If $b = 0$, all $\boldsymbol{z}_i$'s are random, whereas if $b = 1$, all $\boldsymbol{z}_i$'s satisfy $\boldsymbol{e}\cdot\boldsymbol{z}_i = 0 \bmod 2$ conditioned on the event $\boldsymbol{e}\mathbf{R}_0\boldsymbol{v} = 1$ of probability $1/2$.

Please see section 5.1 for how we construct the challenge circuit C and other details in the attack. We note that though our attack is described w.r.t. GSW FHE for arithmetic circuits, it can be easily translated into an attack w.r.t. GSW FHE for Boolean circuits. In particular, the correlation gadget circuit will compute homomorphic AND which translates to computing homomorphic multiplication in GSW and the rest of the attack is the same.

Counterexample to the Wee-Wichs assumption. Wee and Wichs [44] take a different approach, constructing $i\mathcal{O}$ assuming LWE and the ability to obliviously generate LWE samples without knowing the corresponding secrets. They then proposed a heuristic mechanism for *oblivious LWE sampling*, using the dual-GSW homomorphic commitment and any Pseudo-Random Function (PRF). They formulated a concrete conjecture, called the Homomorphic Pseudorandom LWE Samples conjecture, to capture the security of their mechanism. Let us now recall their conjecture.

The Dual GSW Homomorphic Commitment Scheme The scheme is a variant of the homomorphic encryption/commitment schemes of [24,26] with the feature that one can homomorphically evaluate a function with a vector output $f : \{0,1\}^\ell \to \mathbb{Z}_p^m$, and the decommitment to the output commitment to $f(x)$ is shorter than m. Given a public random matrix $\mathbf{A} \in \mathbb{Z}_p^{m\times n}$ where $m \gg n$, a commitment $\mathbf{C}$ to an input $\boldsymbol{x} \in \{0,1\}^\ell$ is

$$\mathbf{C} = (\mathbf{A}\mathbf{R}_1 + x_1\mathbf{G} + \mathbf{E}_1, \cdots, \mathbf{A}\mathbf{R}_\ell + x_\ell\mathbf{G} + \mathbf{E}_\ell)$$

where $\mathbf{R}_i \leftarrow \mathbb{Z}_p^{n\times m\log q}$, $\mathbf{E}_i \leftarrow \chi^{m\times m\log q}$, and $\mathbf{G}$ is the gadget matrix.

The key difference from [24,26] are: *1)* the matrix $\mathbf{A}$ is a thin/tall matrix, whereas in GSW $\mathbf{A}$ is fat/short, *2)* $\mathbf{R}_i$ is fat/short and uniformly sampled, whereas in GSW, they are square matrices consisting of small entries, and *3)* because of the shapes of matrices $\mathbf{A}\mathbf{R}_i$ is far from (pseudo)random and hence additional noises $\mathbf{E}_i$ are added. On the other hand, the hiding property of the commitments still follows directly from LWE, and the same homomorphic evaluation procedure applies. For any Boolean function $f : \{0,1\}^\ell \to \{0,1\}$, one can homomorphically derive a commitment $\mathbf{C}_f = \mathbf{A}\mathbf{R}_f + f(x)\mathbf{G} + \mathbf{E}_f$. Additionally, using the same "packing" procedure, one can homomorphically evaluate $g : \{0,1\}^\ell \to \mathbb{Z}_p^m$ with a vector output to derive a commitment $\mathbf{C}_g = \mathbf{A}\boldsymbol{r}_g + g(x) + \boldsymbol{e}_g$. Observe that the opening to this output commitment is $\boldsymbol{r}_g$ of length $n\log p \ll m$.

The Homomorphic Pseudorandom LWE Samples (HPLS) conjecture considers the following two distributions parameterized by a PRF PRF.

$$\forall\beta \in \{0,1\}, \quad \mathsf{DIST}(\beta) \to \left(\{\boldsymbol{d}_i = \mathbf{A}\widehat{\boldsymbol{s}}_i + \widehat{\boldsymbol{e}}_i\}_{i\in[Q]}, \mathbf{A}, \mathbf{C}, \{\boldsymbol{s}_i\}_{i\in[Q]}\right)$$

where the random variables are sampled as follows: *1)* $\{\boldsymbol{d}_i\}$ are fresh LWE samples with secret $\widehat{\boldsymbol{s}}_i \leftarrow \mathbb{Z}_p^n$ and noise $\widehat{\boldsymbol{e}}_i \leftarrow \chi^m$, *2)* $\mathbf{C}$ is a dual-GSW commitment to a randomly sampled PRF key k and the bit β, and *3)* each $\boldsymbol{s}_i$ is derived from homomorphically evaluating the following computation $g_i(k, \beta)$: the function g_i first evaluates PRF to obtain random bits τ_i, then uses them to sample random LWE secret $\boldsymbol{s}_i^{\mathsf{PRF}}$ and noise $\boldsymbol{e}_i^{\mathsf{PRF}} \leftarrow \chi_{\mathsf{PRF}}^m$ according to a sampling algorithm D, and finally outputs a vector $\mathbf{A}\boldsymbol{s}_i^{\mathsf{PRF}} + \boldsymbol{e}_i^{\mathsf{PRF}} + \beta \boldsymbol{d}_i$.

$$
\begin{aligned}
&g_i(k,\beta): \textit{i)}\ \text{compute } \tau_i \leftarrow \mathsf{PRF}(k,i) \qquad \textit{ii)}\ \text{sample } (\boldsymbol{s}_i^{\mathsf{PRF}}, \boldsymbol{e}_i^{\mathsf{PRF}}) \leftarrow \mathsf{D}(\tau_i)\\
&\qquad \textit{iii)}\ \text{compute and output } \mathbf{A}\boldsymbol{s}_i^{\mathsf{PRF}} + \boldsymbol{e}_i^{\mathsf{PRF}} + \beta\boldsymbol{d}_i = \mathbf{A}(\boldsymbol{s}_i^{\mathsf{PRF}} + \beta\widehat{\boldsymbol{s}}_i) + (\boldsymbol{e}_i^{\mathsf{PRF}} + \beta\widehat{\boldsymbol{e}}_i)\\
&\mathbf{C}_{g_i} = \mathsf{HEval}(g_i, \mathbf{C}) = \mathbf{A}\boldsymbol{r}_i^{\mathsf{Eval}} + g_i(k,\beta) + \boldsymbol{e}_i^{\mathsf{Eval}}\\
&\qquad = \mathbf{A}\underbrace{(\boldsymbol{r}_i^{\mathsf{Eval}} + \boldsymbol{s}_i^{\mathsf{PRF}} + \beta\widehat{\boldsymbol{s}}_i)}_{\boldsymbol{s}_i} + \underbrace{(\boldsymbol{e}_i^{\mathsf{PRF}} + \beta\widehat{\boldsymbol{e}}_i + \boldsymbol{e}_i^{\mathsf{Eval}})}_{\boldsymbol{e}_i}
\end{aligned}
$$

The HPLS conjecture states that for appropriate settings of parameters, in particular when the magnitude of the noises satisfy $\boldsymbol{e}_i^{\mathsf{PRF}} \gg \widehat{\boldsymbol{e}}_i \gg \boldsymbol{e}_i^{\mathsf{Eval}}$, there is a choice of PRF such that $\mathsf{DIST}(0)$ and $\mathsf{DIST}(1)$ are indistinguishable.

Observe that given a sample from the distribution, one can easily compute the noise $\boldsymbol{e}_i$ in $\mathbf{C}_{g_i}$ by using the opened secret vectors $\boldsymbol{s}_i$. Then, the circular security nature of the HPLS conjecture lies in that on one hand we rely on the PRF security to argue that $\boldsymbol{e}_i^{\mathsf{PRF}}$ smudges $\beta\widehat{\boldsymbol{e}}_i + \boldsymbol{e}_i^{\mathsf{Eval}}$, otherwise dual-GSW security is broken, on the other hand, we rely on the dual-GSW security to argue that the PRF key k remains hidden.

Our Counterexample. Our counterexample states that when using dual-GSW for arithmetic computation, for every sampling algorithm D used in the second step of g_i's (that converts random bits τ to a random LWE secret vector $\boldsymbol{s}$ and an error vector $\boldsymbol{e}$ of some distribution χ_{PRF}) there is an arithemtic circuit C_{D} that implements D, such that, for every PRF PRF (and every circuit implementation of PRF), the distributions $\mathsf{DIST}(0)$ and $\mathsf{DIST}(1)$ are distinguishable. In short, the HPLS conjecture is false for every PRF and every sampling algorithm D, if the circuit implementation of D is allowed to be arbitrarily chosen.

Our Ideas In a Nutshell: Our counterexample attacks the noise $\{\boldsymbol{e}_i = (\boldsymbol{e}_i^{\mathsf{PRF}} + \beta\widehat{\boldsymbol{e}}_i + \boldsymbol{e}_i^{\mathsf{Eval}})\}$ that can be derived from a sample of the distribution. To distinguish between $\beta = 0$ or 1, our idea is to create correlation between the parity of $\boldsymbol{e}_i^{\mathsf{PRF}}[1]$ and $\boldsymbol{e}_i^{\mathsf{Eval}}[1]$, so that $\boldsymbol{e}_i[1] \bmod 2$ reveals information about β. We do so by carefully crafting the circuit C_{D} using two gadget circuits described below.

- *Even Gadget:* $G_1(x)$ implements the identity function on a single element x. It first multiplies x by $1/2$, and then adds $x/2$ with itself to get back x (computation over $\mathbb{Z}_p$). Homomorphically evaluating G_1 on a dual-GSW commitment $\mathsf{ct} = \mathbf{AR} + x\mathbf{G} + \mathbf{E}$ to x produces a commitment $\mathbf{C}' = \mathbf{AR}' + x\mathbf{G} + \mathbf{E}'$ with even errors $\mathbf{E}'$.

$$
\begin{aligned}
\mathbf{C} = \mathbf{AR} + x\mathbf{G} + \mathbf{E} \quad &\xrightarrow{\mathsf{HEval}\ \times\frac{1}{2}} \quad \mathbf{C}'' = \mathbf{AR}'' + \frac{x}{2}\mathbf{G} + \mathbf{E}''\\
&\xrightarrow{\mathsf{HEval}\ +} \quad \mathbf{C}' = \mathbf{AR}' + x\mathbf{G} + \mathbf{E}', \quad \text{where } \mathbf{E}' = 2\mathbf{E}''
\end{aligned}
$$

– *Correlation Gadget:* The second gadget circuit $G_2(x, 1)$ first computes $G_1(x)$ to get x, and then multiplies it with 1. Homomorphically evaluating G_2 on dual-GSW commitment $\mathbf{C}$ to x and $\mathbf{C}_1$ to 1 produces a new commitment $\mathbf{C}' = \mathbf{A}\mathbf{R}' + x\mathbf{G} + \mathbf{E}'$ of x where the parity of $\mathbf{E}'[1,1]$ is correlated with x if $\mathbf{E}_1[1,1]$ is odd, where $\mathbf{E}_1$ is the noise in $\mathbf{C}_1$.

$$\mathsf{ct} = \mathbf{A}\mathbf{R} + x\mathbf{G} + \mathbf{E} \xrightarrow{\mathsf{HEval}\ G_1} \mathsf{ct}'' = \mathbf{A}\mathbf{R}' + x\mathbf{G} + (2\mathbf{E}'')$$
$$\xrightarrow{\mathsf{HEval}\ \times(\mathsf{ct}_1 = \mathbf{A}\mathbf{R}_1 + \mathbf{G} + \mathbf{E}_1)} \mathsf{ct}' = \mathbf{A}\mathbf{R}' + x\mathbf{G} + \mathbf{E}',\ \mathbf{E}' = 2\mathbf{E}''G^{-1}(\mathsf{ct}_1) + x\mathbf{E}_1$$

Using them, we create correlation between $\boldsymbol{e}_i^{\mathsf{PRF}}[1] \bmod 2$ and $\boldsymbol{e}_i^{\mathsf{Eval}}[1] \bmod 2$.

Before we can declare success, we must resolve two other issues. First, the correlation created by the second gadget is probabilistic, depending on the parity of noise $\mathbf{E}_1[1,1]$ embedded in commitment $\mathbf{C}_1$. This is not too much of a problem since $\mathbf{C}_1$ is reused for all index i and hence with probability $1/2$, we see an observable pattern in all $\boldsymbol{e}_i$. Second, the homomorphic evaluation of $\beta\boldsymbol{d}_i$ is outside the control of C_{D} and its noise will be added to the final output of g_i. We overcome the issue by observing that noises resulting from this homomorphic evaluation induces an over-determined linear system over the noises $\mathbf{E}_\beta$ in the commitment to β. Thus, we can use linearity testing to help the attack distinguish.

Possible Extension. One natural follow-up question is whether our techniques can be extended to directly attack these recent $i\mathcal{O}$ constructions [10,21,11,44], beyond the circular security assumptions they rely on. On this front, we think that our attack ideas can be extended to break the security of the $i\mathcal{O}$ scheme of [10] (and possibly its followups [21,11]), if one is allowed to manipulate the implementation of the underlying FHE scheme (e.g., using odd noises to generate the public key of the GSW FHE scheme) and the implementation of circuits computed (e.g., the circuit for computing mod). However, in this work, we focus only on the assumptions, and leave direct attacks to the schemes as future work.

A Perspective. First, our attacks highlight the importance of building schemes from well-founded assumptions. However, in cases where existing techniques are far from reaching this goal, one way of making progress is through cycles of proposals and attacks, and a measure of progress is the simplicity of the proposed assumptions, and whether they are natural and connected to well-studied areas in computer science. For instance, the recent line of $i\mathcal{O}$ constructions [5,35,4,30,31,20] started with assuming new assumptions, and eventually led to the first $i\mathcal{O}$ construction [32] based on four well-founded assumptions – LWE, the decision linear assumption over symmetric key pairing, LPN over large fields, and PRG in NC^0.

At this moment, we still lack good understanding on the front of constructing $i\mathcal{O}$ solely from lattices (or constructing post-quantum secure $i\mathcal{O}$). The works of [10,21,11,44] proposed refreshing approaches and ideas. The purpose of our work, through counterexamples, is finding weak points in these new approaches, so that, they can be addressed and the assumptions can be refined in future

works. In particular, a main lesson from our counterexamples is that when working with leakage of noises in LWE, it is important to examine the specific leakage carefully.

Other $i\mathcal{O}$ Constructions. Our work focuses on the new types of circular security assumptions / hard problems underlying recent $i\mathcal{O}$ constructions of [10,21,11,44]. Prior to their work, Agrawal [2] gave an $i\mathcal{O}$ construction based on noisy linear functional encryption and proposed a candidate noisy linear functional encryption based on new types of NTRU assumptions. The work of [3] cryptanalyzed of the new NTRU assumptions and further refined them. There are many $i\mathcal{O}$ constructions based on multilinear maps, which can be instantiated from lattices (see references in [32]). Though all known multilinear map instantiation have been attacked, there are still $i\mathcal{O}$ candidates based on them that are unbroken, for instance [19]. Furthermore, Gentry, Jutla and Kane [23] proposed an $i\mathcal{O}$ candidate using tensor products. Finally, using bilinear pairing, a line of constructions [5,35,4,30,31,20] recently led to the first $i\mathcal{O}$ construction by [32] based on four well-founded assumptions – LWE, the decision linear assumption over symmetric key pairing, LPN over large fields, and PRG in NC^0.

2 Preliminaries

We start by recalling the security definitions that will be useful for the rest of the paper.

2.1 Security Definitions Introduced by Gay-Pass

We now recall the notion of $\mathcal{O}$-leakage resilience property of a public key encryption scheme, PKE. A PKE scheme satisfies $\mathcal{O}$-leakage resilience property if it is hard for a computationally efficient adversary to guess the challenge bit even in presence of valid oracle queries from the oracle $\mathcal{O}$, which may potentially leak information about the challenge message as well as the randomness.

Definition 1 ($\mathcal{O}$-leakage resilient security). *We say that a* $\mathsf{PKE} = (\mathsf{Setup}, \mathsf{Enc}, \mathsf{Dec})$ *scheme satisfies $\mathcal{O}$-leakage resilience security if for every stateful non-uniform ppt adversaries $\mathcal{A}$, there exists some negligible function* $\mathsf{negl}(\cdot)$ *such that for $\lambda \in \mathbb{N}$,* $\Pr[\mathsf{Expt}^{\mathsf{PKE}}_{\lambda,\mathcal{A}} = 1] \leq \frac{1}{2} + \mathsf{negl}$*, where the experiment* $\mathsf{Expt}^{\mathsf{PKE}}_{\lambda,\mathcal{A}}$ *is defined as follows*[4]*:*

$$\mathsf{Expt}^{\mathsf{PKE}}_{\lambda,\mathcal{A}} = \left\{ \begin{array}{c} (\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Setup}(1^\lambda) \\ (\boldsymbol{m}^0, \boldsymbol{m}^1) \leftarrow \mathcal{A}(\mathsf{pk}),\ b \leftarrow \{0,1\} \\ \boldsymbol{m}^* = \boldsymbol{m}^b,\ \boldsymbol{r} \leftarrow \{0,1\}^* \\ \mathsf{ct} = \mathsf{Enc}(\mathsf{pk}, \boldsymbol{m}^*; \boldsymbol{r}); b' \leftarrow \mathcal{A}^{\mathcal{O}(\mathsf{pk}, \boldsymbol{m}^*, \boldsymbol{r})}(\mathsf{ct}) \\ \textit{Return 1 if } |\boldsymbol{m}^0| = |\boldsymbol{m}^1|,\ b' = b \textit{ and } \mathcal{O} \textit{ did not return } \perp;\ 0 \textit{ otherwise} \end{array} \right\}$$

[4] In the definition below and otherwise, we denote by shorthand $\boldsymbol{r} \leftarrow \{0,1\}^*$ to mean that the randomness is sampled from the appropriate distribution.

We say that a PKE *scheme is secure if it is not given access to any oracle in the same experiment.*

We now define the notion $\mathcal{O}$-leakage resilient security in presence of encrypted key cycles. The notion is called $\mathcal{O}$-leakage resilient 2-circular security.

Definition 2 ($\mathcal{O}$-leakage resilient 2-circular security). *We say that the scheme* $\mathsf{PKE}_1 = (\mathsf{Setup}_1, \mathsf{Enc}_1, \mathsf{Dec}_1)$ *and the scheme* $\mathsf{PKE}_2 = (\mathsf{Setup}_2, \mathsf{Enc}_2, \mathsf{Dec}_2)$ *satisfies $\mathcal{O}$-leakage resilient 2- circular security if for every stateful non-uniform ppt adversaries $\mathcal{A}$, there exists some negligible function* $\mathsf{negl}(\cdot)$ *such that for* $\lambda \in \mathbb{N}$, $\Pr[\mathsf{Expt}^{\mathsf{PKE}_1,\mathsf{PKE}_2}_{\lambda,(A)} = 1] \leq \frac{1}{2} + \mathsf{negl}$, *where the experiment* $\mathsf{Expt}^{\mathsf{PKE}_1,\mathsf{PKE}_2}_{\lambda,\mathcal{A}}$ *is defined as follows:*

$$\mathsf{Expt}^{\mathsf{PKE}_1,\mathsf{PKE}_2}_{\lambda,\mathcal{A}} = \left\{ \begin{array}{c} (\mathsf{pk}_1, \mathsf{sk}_1) \leftarrow \mathsf{Setup}_1(1^\lambda),\ (\mathsf{pk}_2, \mathsf{sk}_2) \leftarrow \mathsf{Setup}_2(1^\lambda) \\ (\boldsymbol{m}^0, \boldsymbol{m}^1) \leftarrow \mathcal{A}(\mathsf{pk}_1, \mathsf{pk}_2),\ b \leftarrow \{0,1\} \\ \boldsymbol{m}^* = \mathsf{sk}_2 \| \boldsymbol{m}^b,\ \boldsymbol{r} \leftarrow \{0,1\}^* \\ \mathsf{ct}_1 = \mathsf{Enc}_1(\mathsf{pk}_1, \boldsymbol{m}^*; \boldsymbol{r});\ \mathsf{ct}_2 = \mathsf{Enc}_2(\mathsf{pk}_2, \mathsf{sk}_1);\ b' \leftarrow \mathcal{A}^{\mathcal{O}(\mathsf{pk}_1, \boldsymbol{m}^*, \boldsymbol{r})}(\mathsf{ct}_1, \mathsf{ct}_2) \\ \textit{Return 1 if } |\boldsymbol{m}^0| = |\boldsymbol{m}^1|,\ b' = b \textit{ and } \mathcal{O} \textit{ did not return } \perp;\ 0 \textit{ otherwise} \end{array} \right\}$$

We say that a PKE *scheme is 2-circular secure if it is not given access to any oracle in the same experiment.*

2.2 Fully-Homomorphic Encryption Scheme

We present the definition of a fully-homomorphic encryption scheme below with additional properties as defined by [21].

Definition 3 (FHE). *A fully homomorphic encryption scheme for the circuit class* $\mathcal{C} = \{\mathcal{C}_{\lambda,d}\}_{\lambda,d\in\mathbb{N}}$ *and randomness space* $\mathcal{R} = \{\mathcal{R}_{\lambda,d}\}_{\lambda.d\in\mathbb{N}}$ *is a tuple of PPT algorithms*

$$\mathsf{FHE} = (\mathsf{Setup}, \mathsf{Enc}, \mathsf{Eval}, \mathsf{Dec})$$

satisfying the following specifications:

$(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Setup}(1^\lambda, 1^d)$**:** *The setup algorithm takes as input a security parameter $\lambda \in \mathbb{N}$, a circuit depth bound $d \in \mathbb{N}$ (which is a polynomial in the security parameter). It outputs a key pair* $(\mathsf{pk}, \mathsf{sk})$.

$\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{pk}, m; \boldsymbol{r})$**:** *It takes as input a public key pk and a plaintext $m \in \{0,1\}$ and a randomness $\boldsymbol{r} \in \mathcal{R}_{\lambda,d}$ and outputs a ciphertext* ct. *Here $\mathcal{R}_{\lambda,d} \subseteq \{0,1\}^*$ is some finite set. Encryption of multiple bits is done by encrypting each of them separately.*

$\widehat{\mathsf{ct}} \leftarrow \mathsf{Eval}(C, \mathsf{ct}_1, \ldots, \mathsf{ct}_\ell)$**:** *It takes as input a boolean circuit $C\colon \{0,1\}^\ell \to \{0,1\} \in \mathcal{C}_{\lambda,d}$ of depth $\leq d$ and ciphertexts $ct_1, \ldots, ct_\ell$. It outputs an evaluated ciphertext $\widehat{ct}$.*

$\widehat{m} \leftarrow \mathsf{Dec}(\mathsf{sk}, \widehat{\mathsf{ct}})$**:** *The decryption algorithm takes in the secret key* sk *and a possibly evaluated ciphertext* $\widehat{\mathsf{ct}}$. *It outputs $\widehat{m} \in \{0, 1, \perp\}$.*

A fully-homomorphic encryption scheme satisfies correctness:

(Perfect) Correctness: *For every λ, any polynomial $d(\lambda) \in \mathbb{N}$ and $\ell \in \mathbb{N}$, every key-pair $(\mathsf{pk}, \mathsf{sk})$ in the support of $\mathsf{Setup}(1^\lambda, 1^d)$, every set of messages $m_1, \ldots, m_\ell \in \{0,1\}^\ell$, every ciphertext $\{\mathsf{ct}_i\}_{i\in[\ell]}$ in the support of $\{\mathsf{Enc}(\mathsf{pk}, m_i)\}_{i\in[\ell]}$ and every circuit $C : \{0,1\}^\ell \to \{0,1\}$ in $\mathcal{C}_{\lambda,d}$, $\mathsf{Dec}(\mathsf{sk}, \widehat{\mathsf{ct}}) = C(m_1, \ldots, m_\ell)$ where $\widehat{\mathsf{ct}} = \mathsf{Eval}(C, \mathsf{ct}_1, \ldots, \mathsf{ct}_\ell)$.*

Remark 1. For any polynomial $d(\cdot)$, We denote by FHE_d, an FHE scheme where the depth is hardwired to be $d(\lambda)$.

Above we omit the security definition which is identical to the definition of security for a public-key encryption scheme and the notion of (levelled) compactness which says that the size of a fresh as well as an evaludated ciphertext encrypting a single bit is bounded by $\mathsf{poly}(\lambda, d)$ for some fixed polynomial poly. We refer the reader to [24,13] for detailed definitions.

Now we define additional algorithms that were introduced by [21]. Any FHE scheme is not required to exhibit these, although, most of the known schemes do.

Definition 4 (Extra-Noisy Encryption). *We denote by Enc^* an* extra-noisy *encryption algorithm, which has the same syntax as the encryption algorithm Enc, except that the randomness it uses is sampled uniformly from another set $\mathcal{R}^* = \{\mathcal{R}^*_{\lambda,d}\}_{\lambda,d\in\mathbb{N}}$.*

We call $\mathcal{R}^*$ as the extra-noisy randomness space and any ciphertext encrypted using Enc^* as an "extra-noisy" encryption.

Randomness Homomorphism. Given $\ell \in \mathbb{N}$ ciphertexts, $\{\mathsf{ct}_i\}_{i\in[\ell]}$, underlying message $\{m_i\}_{i\in[\ell]}$ and randomness $\{\boldsymbol{r}_i\}_{i\in[\ell]}$ where each $\boldsymbol{r}_i \in \mathcal{R}_{\lambda,d}$, and any circuit $C \in \mathcal{C}_{\lambda,d}$, in most FHE schemes it is possible to efficiently recover randomness $\boldsymbol{r}_C \in \mathcal{R}^*_{\lambda,d}$ such that $\mathsf{Eval}(\mathsf{pk}, C, \mathsf{ct}_1, \ldots, \mathsf{ct}_\ell) = \mathsf{Enc}^*(\mathsf{pk}, C(m_1, \ldots, m_\ell); \boldsymbol{r}_C)$. This algorithm is denoted by $\mathsf{RandEval}$.

Definition 5 (Randomness Homomorphism). *An FHE scheme with extra noisy randomness space $\mathcal{R}^*$ satisfies randomness homomorphism property if there exists a probabilistic polynomial time algorithm $\mathsf{RandEval}$ with the following property.*
For any $\lambda \in \mathbb{N}$ and any polynomial $d(\lambda) \in \mathbb{N}$, $\mathsf{RandEval}(\mathsf{pk}, C, \boldsymbol{r}, \boldsymbol{m})$ takes as input a public key pk in the support of $\mathsf{Setup}(1^\lambda, 1^d)$, a circuit $C : \{0,1\}^\ell \to \{0,1\} \in \mathcal{C}_{\lambda,d}$, randomness $\boldsymbol{r} = (\boldsymbol{r}_1, \ldots, \boldsymbol{r}_\ell) \in \mathcal{R}^\ell_{\lambda,d}$ and messages $\boldsymbol{m} = (m_1, \ldots, m_\ell) \in \{0,1\}^\ell$, and it outputs $\boldsymbol{r}_C \in \mathcal{R}^*_{\lambda,d}$ such that:

$$\mathsf{Eval}(\mathsf{pk}, C, \mathsf{Enc}(\mathsf{pk}, m_1; \boldsymbol{r}_1), \ldots, \mathsf{Enc}(\mathsf{pk}, m_\ell; \boldsymbol{r}_\ell)) = \mathsf{Enc}^*(\mathsf{pk}, C(m_1, \ldots, m_\ell); \boldsymbol{r}_C)$$

We now define the notion of SRL security for a fully-homomorphic encryption scheme with an extra-noisy encryption algorithm and randomness homomorphism property.

Definition 6. *A fully homomorphic encryption scheme with extra-noisy encryption and randomness homomorphism property for depth d, denoted as* FHE_d *is said to be* SRL-secure *if it is* $\mathcal{O}_{\mathsf{SRL}}^{\mathsf{FHE}_d}$*-leakage resilient secure for the following oracle.*

Oracle $\mathcal{O}_{\mathsf{SRL}}^{\mathsf{FHE}_d}(\mathsf{pk}, \boldsymbol{m}^*, \boldsymbol{r})$

$\boldsymbol{r}^* \leftarrow \mathcal{R}^*,\ \mathsf{ct}^* = \mathsf{Enc}^*(\mathsf{pk}, 0; \boldsymbol{r}^*)$
$(f, \alpha) \leftarrow \mathcal{A}(\mathsf{ct}^*)$
$\boldsymbol{r}_f = \mathsf{RandEval}(\mathsf{pk}, f, \boldsymbol{r}, \boldsymbol{m}^*)$
If $f \in \mathcal{C}_{\lambda,d}$ *and* $\alpha = f(\boldsymbol{m}^*)$*, then set* $\mathsf{leak} = \boldsymbol{r}^* - \boldsymbol{r}_f \in \mathcal{R}^*$
Otherwise set $\mathsf{leak} = \perp$*. Output* leak*.*

3 Homomorphic Encryption Schemes

Below we recall both GSW Encryption [24] and its Dual formulation [14]. In the sections, we also specify the exact modifications we need for our counterexample. We assume familiarity with some notations relevant in lattice based cryptography. For completeness, they are outlined in Appendix A.

3.1 Gentry-Sahai-Waters FHE Scheme

We now describe our scheme, and set parameters later when needed in the counterexample. Below is a list of symbols to be used in the scheme.

- λ is the security parameter,
- $d(\lambda)$ is the polynomial depth bound,
- p is the prime modulus used in the scheme,
- n, m, w are polynomials in λ, and are used as dimensions of the matrices involved,
- χ is an error distribution used for generating LWE samples,
- B_1, B_2 are $\mathsf{poly}(\lambda, d)$-bit positive integers that are used as bounds. B_1 is the bound on the infinity norm of the randomness in the evaluated ciphertext after evaluating a depth d circuit,
- Assuming $w = n \cdot \lceil \log_2 p \rceil$, let $\mathbf{G}$ be the gadget matrix of dimension $n \times w$,
- $\mathcal{C}_{\lambda,d}$ consists of all polynomial sized arithmetic circuits of depth d with Boolean inputs and outputs composed of multiplication, addition and multiplication by a constant in $\mathbb{Z}_p$, with the following special property: For any Boolean input, during the evaluation, all the multiplication gates are evaluated on Boolean inputs.

Remark 2. In the circuit class $\mathcal{C}_{\lambda,d}$, in particular, we allow multiplication by a potentially large field element, as long as all inputs to all multiplication gates are Boolean. Our counterexample for 2-Circ SRL security will only exploit boolean computations, whereas the counterexample for [44] will exploit such multiplication by constant gates.

Now we describe the scheme:

$\mathsf{Setup}(1^\lambda, 1^d) \to (\mathsf{pk}, \mathsf{sk})$: Perform the following steps.
- Sample $\mathbf{A} \leftarrow \mathbb{Z}_p^{(n-1)\times m}$.
- Sample $\boldsymbol{s} \leftarrow \mathbb{Z}_p^{1\times(n-1)}$.
- Sample $\boldsymbol{e} \leftarrow \chi^{1\times m}$. Set $\boldsymbol{b} = \boldsymbol{s}\cdot\mathbf{A} + \boldsymbol{e} \mod p$
- Set $\mathbf{U} = [\mathbf{A}^\top | \mathbf{b}^\top]^\top \in \mathbb{Z}_p^{n\times m}$.
- Output $\mathsf{pk} = \mathbf{U}$ and $\mathsf{sk} = \boldsymbol{s}$.

$\mathsf{Enc}(\mathsf{pk}, \mu) \to \mathsf{ct}$: To encrypt a bit $\mu \in \{0,1\}$ perform the following steps.
- Sample $\mathbf{R} \leftarrow [-1,1]^{m\times w}$.
- Compute and output $\mathsf{ct} = \mathbf{U}\cdot\mathbf{R} + \mu\cdot\mathbf{G}$ where $\mathbf{G}$ is the gadget matrix of dimension $n \times w$.

$\mathsf{Eval}(\mathsf{pk}, C, \mathsf{ct}_1, \ldots, \mathsf{ct}_\ell) \to \widehat{\mathsf{ct}}$: To evaluate an arithmetic circuit $C : \{0,1\}^\ell \to \{0,1\}$, perform the following operations gate by gate, as per the gate evaluation rules below and according to the topological ordering provided by the circuit.
- **Addition:** $\mathsf{Add}(\mathsf{ct}'_1, \mathsf{ct}'_2)$, Output $\mathsf{ct}'_1 + \mathsf{ct}'_2$.
- **Multiplication:** $\mathsf{Mult}(\mathsf{ct}'_1, \mathsf{ct}'_2)$, Output $\mathsf{ct}'_1 \cdot \mathbf{G}^{-1}(\mathsf{ct}'_2)$.
- **Multiplication by a constant** $c \in \mathbb{Z}_q$**:** $\mathsf{ConstMult}(c, \mathsf{ct}'_1)$, Output $\mathsf{ct}'_1 \cdot \mathbf{G}^{-1}(c\cdot\mathbf{G})$.

$\mathsf{Dec}(\mathsf{sk}, \widehat{\mathsf{ct}})$: To decrypt, compute: $z = (-\boldsymbol{s}\|1)\cdot\widehat{\mathsf{ct}}\cdot\boldsymbol{v}$ where $\boldsymbol{v} \in \{0,1\}^{w\times 1}$ with $\boldsymbol{v}_i = 1$ iff $i = w-2$. Output 0 if $|z| \le \frac{p}{16}$ and 1 otherwise.

We now observe that the GSW scheme above satisfies Randomness Homomorphism property.

Randomness Homomorphism: Below we define the algorithms that make up the the randomness homomorphism property.

$\mathsf{Enc}^*(\mathsf{pk}, \mu)$: For an extra noisy encryption of a bit $\mu \in \{0,1\}$ perform the following steps.
- Sample $\mathbf{R}^* \leftarrow [-B_2, B_2]^{m\times w}$.
- Compute and output $\mathsf{ct}^* = \mathbf{U}\cdot\mathbf{R}^* + \mu\cdot\mathbf{G}$ where $\mathbf{G}$ is the gadget matrix of dimension $n \times w$.

$\mathsf{RandEval}(\mathsf{pk}, C, \{\mathbf{R}_i\}_{i\in[\ell]}, \{m_i\}_{i\in[\ell]})$: Just as in evaluation of ciphertext, compute the randomness gate by gate. For gates with fan-in 2, let $\mathbf{R}'_1$ and $\mathbf{R}'_2$ be the input randomness, m'_1, m'_2 be the input messages and ct'_1, ct'_2 be the corresponding ciphertext. For the multiplication gate, let the input be the values with subscript "1". Below we describe the process to compute the randomness that is propogated. Messages can be computed by evaluating the circuit.
- For addition gate, output $\mathbf{R}'_1 + \mathbf{R}'_2$.
- For multiplication gate, output $\mathbf{R}'_1 \cdot \mathbf{G}^{-1}(\mathsf{ct}'_2) + m'_1 \cdot \mathbf{R}'_2$.
- For multiplication by the constant c, output $\mathbf{R}'_1 \cdot \mathbf{G}^{-1}(c\cdot\mathbf{G})$.

[21] observed that GSW scheme satisfies plain SRL security.

3.2 Dual-GSW Homomomorphic Commitment Scheme

We now provide the Dual-GSW homomorphic commitment scheme [14] as described by [44]. We set parameters later when needed in the counterexample. Below is a list of symbols to be used in the scheme.

- λ is the security parameter,
- $d(\lambda)$ is the polynomial depth bound,
- p is the prime modulus used in the scheme,
- n, m, w are polynomials in λ, and are used as dimensions of the matrices involved,
- χ is an error distribution used for generating LWE samples,
- Assuming $w = m \cdot \lceil \log_2 p \rceil$, let $\mathbf{G}$ be the gadget matrix of dimension $m \times w$,
- $\mathcal{C}_{\lambda,d}$ consists of all polynomial sized arithmetic circuits of depth d with boolean inputs and outputs composed of multiplication, addition and multiplication by a constant in $\mathbb{Z}_p$, with the following special property: For any boolean input, during the evaluation, all the multiplication gates are evaluated on binary inputs.

Now we describe the scheme,

$\mathsf{Setup}(1^\lambda, 1^d) \to \mathsf{pk}$: Perform the following steps.
- Sample $\mathbf{A} \leftarrow \mathbb{Z}_p^{m \times n}$.
- Output $\mathsf{pk} = \mathbf{A}$.

$\mathsf{Enc}(\mathsf{pk}, \mu) \to \mathsf{ct}$: To compute a commitment ct to a bit $\mu \in \{0,1\}$ perform the following steps.
- Sample $\mathbf{R} \leftarrow \mathbb{Z}_p^{n \times w}$.
- Sample $\mathbf{E} \leftarrow \chi^{m \times w}$.
- Compute and output $\mathsf{ct} = \mathbf{A} \cdot \mathbf{R} + \mu \cdot \mathbf{G} + \mathbf{E}$ where $\mathbf{G}$ is the gadget matrix of dimension $m \times w$.

Evaluation We now define two evaluation algorithms, Eval_1 and Eval_2. Eval_1 takes as inputs $\mathsf{ct}_1, \ldots, \mathsf{ct}_\ell$ committing bits $\mu_1, \ldots, \mu_\ell$ and a function $C : \{0,1\}^\ell \to \mathbb{Z}_p$ in $\mathcal{C}_{\lambda,d}$ and computes a commitment of $C(\mu_1, \ldots, \mu_\ell)$. Eval_2 takes as input commitments $\widehat{\mathsf{ct}}_1, \ldots, \widehat{\mathsf{ct}}_m$ commiting elements $\widehat{\mu_1}, \ldots, \widehat{\mu_m}$ and outputs a packed commitment $\widehat{\mathsf{ct}}_{packed} \in \mathbb{Z}_p^{m \times 1}$ of the form $\mathbf{A}\widehat{\mathbf{r}} + \widehat{\mu} + \mathbf{e}$ where $\widehat{\mu} = (\widehat{\mu_1}, \ldots, \widehat{\mu_m})^\top$. The evaluation algorithm for circuits of the form $g : \{0,1\}^\ell \to \mathbb{Z}_p^m \in \mathcal{C}_{\lambda,d}$ in [44] is a composition of these two evaluation algorithms (Eval_1 followed by Eval_2).

$\mathsf{Eval}_1(\mathsf{pk}, C, \mathsf{ct}_1, \ldots, \mathsf{ct}_\ell) \to \widehat{\mathsf{ct}}$: To evaluate an arithmetic circuit $C : \{0,1\}^\ell \to \mathbb{Z}_p$ in $\mathcal{C}_{\lambda,d}$, perform the following operations gate by gate, as per the gate evaluation rules below and according to the topological ordering provided by the circuit.
- **Addition:** $\mathsf{Add}(\mathsf{ct}'_1, \mathsf{ct}'_2)$, Output $\mathsf{ct}'_1 + \mathsf{ct}'_2$.
- **Multiplication:** $\mathsf{Mult}(\mathsf{ct}'_1, \mathsf{ct}'_2)$, Output $\mathsf{ct}'_1 \cdot \mathbf{G}^{-1}(\mathsf{ct}'_2)$.
- **Multiplication by a constant** $c \in \mathbb{Z}_p$**:** $\mathsf{ConstMult}(c, \mathsf{ct}'_1)$, Output $\mathsf{ct}'_1 \cdot \mathbf{G}^{-1}(c \cdot \mathbf{G})$.

$\mathsf{Eval}_2(\mathsf{ct}_1, \ldots, \mathsf{ct}_m) \to \widehat{\mathsf{ct}}_{packed}$

– Output $\widehat{\mathsf{ct}}_{packed} = \sum_{i\in[m]} \mathsf{ct}_i \cdot \mathbf{G}^{-1}(\mathbf{1}_i)$ where $\mathbf{1}_i \in \{0,1\}^{1\times m}$ is the indicator vector with 1 at the i^{th} position. We refer to this output as a packed commitment.

$\mathsf{Eval}_{open,packed}(g, \mathbf{A}, \{\mathbf{R}_i\}_{i\in[\ell]}, \{x_i\}_{i\in[\ell]}, \{\mathbf{E}_i\}_{i\in[\ell]})$, the $\mathsf{Eval}_{open,packed}$ takes as input a circuit $g : \{0,1\}^\ell \to \mathbb{Z}_p^m \in \mathcal{C}_{\lambda,d}$, public key $\mathbf{A}$, and ℓ randomness-message tuples $(\mathbf{R}_i, \mathbf{E}_i, x_i)$, and it outputs the opening for $\widehat{\mathsf{ct}}_{packed} = \mathsf{Eval}(g, \mathsf{ct}_1, \ldots, \mathsf{ct}_\ell) = \mathbf{A}\widehat{\mathbf{r}} + \widehat{\mathbf{e}} + g(\boldsymbol{x})$ where $\mathsf{ct}_i = \mathbf{A}\mathbf{R}_i + \mathbf{E}_i + x_i\mathbf{G}$. This is done in two steps. First, it propagates openings for unpacked ciphertexts. Let g_i for $i \in [m]$, denote the circuit computing the i^{th} component. It runs $\mathsf{Eval}_{open}(g_i, \mathbf{A}, \{\mathbf{R}_i\}_{i\in[\ell]}, \{x_i\}_{i\in[\ell]}, \{\mathbf{E}_i\}_{i\in[\ell]})$ for $i \in [m]$ below:

– $\mathsf{Eval}_{open}(g_i, \mathbf{A}, \{\mathbf{R}_i\}_{i\in[\ell]}, \{x_i\}_{i\in[\ell]}, \{\mathbf{E}_i\}_{i\in[\ell]})$, the Eval_{open} algorithm takes as input a circuit $g_i : \{0,1\}^\ell \to \mathbb{Z}_p \in \mathcal{C}_{\lambda,d}$, matrix $\mathbf{A}$, randomness and messages for commitments $\mathsf{ct}_i = \mathbf{A}\mathbf{R}_i + x_i\mathbf{G} + \mathbf{E}_i$ and it outputs randomness and messages of the evaluated commitment $\widehat{\mathsf{ct}}_i = \mathbf{A}\widehat{\mathbf{R}}_i + \widehat{\mathbf{E}}_i + g_i(x_1, \ldots, x_\ell)\mathbf{G}$. This is done by propagating gate by gate. Let $\mathbf{R}'_1$, $\mathbf{R}'_2$, $\mathbf{E}'_1$ and $\mathbf{E}'_2$ be the input randomness, and x'_1 and x'_2 be the inputs. For gates with a single input, let the subscript of the input be 1. Let $\mathsf{ct}'_b = \mathbf{A}\mathbf{R}'_b + x_b\mathbf{G} + \mathbf{E}'_b$ for $b \in \{1,2\}$.
  - For addition gate, output $\mathbf{R}'_1 + \mathbf{R}'_2$, $\mathbf{E}'_1 + \mathbf{E}'_2$ and $x'_1 + x'_2$.
  - For multiplication gate, output $\mathbf{R}'_1 \cdot \mathbf{G}^{-1}(\mathsf{ct}'_2) + x'_1 \cdot \mathbf{R}'_2$, $\mathbf{E}'_1 \cdot \mathbf{G}^{-1}(\mathsf{ct}'_2) + x'_1 \cdot \mathbf{E}'_2$ and $x'_1 \cdot x'_2$.
  - For multiplication by constant output $\mathbf{R}'_1 \cdot \mathbf{G}^{-1}(c \cdot \mathbf{G})$, $\mathbf{E}'_1 \cdot \mathbf{G}^{-1}(c \cdot \mathbf{G})$ and $c \cdot x'_1$.

Then, for the opening of the packed commitment it outputs $\widehat{\mathbf{r}} = \sum_{i\in[m]} \widehat{\mathbf{R}}_i \mathbf{G}^{-1}(\mathbf{1}_i)$, $\widehat{\mathbf{e}} = \sum_{i\in[m]} \widehat{\mathbf{E}}_i \mathbf{G}^{-1}(\mathbf{1}_i)$ and $\boldsymbol{y} = g(\boldsymbol{x})$.

4 Correlation-Inducing Gates

We turn to the conceptual heart of our attacks: two simple transformations on FHE ciphertexts which, put together, have the following effect. Given the ciphertext ct_x for a bit $x \in \{0,1\}$, we produce a new ciphertext ct'_x which still decrypts to x, such that the "noise part" of ct'_x is correlated with x. The exact meaning of "noise part" depends on the underlying FHE scheme – we show this for the dual version of [24] as described by [14].

Crucially, these transformations can be realized by standard homomorphic evaluation of multiplication and addition gates, as well as homomorphic evaluation of gates which multiply by constants $c \in \mathbb{Z}_p$. Therefore, we can package them into a special identity gate which can be appended to any circuit to produce a new circuit which computes the same function as the old one, but such that standard homomorphic evaluation of that circuit produces a ciphertext where the noise part and message part are correlated.

4.1 Correlation-Inducing Gate for Dual-GSW

In this subsection we adopt notation as in Section 3.2. In particular, we assume the presence of a public key $\mathbf{A} \in \mathbb{Z}_p^{m\times n}$.

The first half of our correlation-inducing gate for dual-GSW is captured in the following lemma.

Lemma 1 (Even-noise gate for dual-GSW). *Let* $\mathsf{ct}_x = \mathbf{AS} + x\mathbf{G} + \mathbf{E}$ *be a dual-GSW encryption of some* $x \in \mathbb{Z}_p$*, where each entry* $|\mathbf{E}[i,j]| \leq B$ *for some* $B \leq p/(100w)$ *and let* ct'_x *be the result of homomorphically evaluating the following two gates:*

1. $g_1(x) = \frac{1}{2}x$
2. $g_2(x) = x + x$.

That is, $\mathsf{ct}'_x = \mathsf{Eval}(g_2, \mathsf{Eval}(g_1, \mathsf{ct}_x))$ *(where we use the public key* $\mathbf{A}$*). Then* $\mathsf{ct}'_x = \mathbf{AS}' + x\mathbf{G} + \mathbf{E}'$ *for some* $\mathbf{S}' \in \mathbb{Z}_p^{n\times w}$ *and some matrix* $\mathbf{E}' \in \mathbb{Z}_p^{m\times w}$ *for which every entry satisfies* $\mathbf{E}'[i,j] = 2\cdot e_{ij}$ *for some* $e_{ij} \in \mathbb{Z}_p$ *with* $|e_{i,j}| \leq O(Bw)$.

The proof is a simple calculation

Proof. Expanding,

$$\begin{aligned}\mathsf{Eval}(g_1, \mathsf{ct}_x) &= (\mathbf{AS} + x\mathbf{G} + \mathbf{E}) \cdot \mathbf{G}^{-1}(\tfrac{1}{2} \cdot \mathbf{G}) \\ &= \mathbf{A}(\mathbf{S} \cdot \mathbf{G}^{-1}(\tfrac{1}{2} \cdot \mathbf{G})) + x \cdot \tfrac{1}{2} \cdot \mathbf{G} + \mathbf{E} \cdot \mathbf{G}^{-\mathbf{1}}(\tfrac{1}{2} \cdot \mathbf{G})\end{aligned}$$

So,

$$\begin{aligned}&\mathsf{Eval}(g_2, \mathbf{A}(\mathbf{S} \cdot \mathbf{G}^{-1}(\tfrac{1}{2} \cdot \mathbf{G})) + x \cdot \tfrac{1}{2} \cdot \mathbf{G} + \mathbf{E} \cdot \mathbf{G}^{-1}(\tfrac{1}{2} \cdot \mathbf{G})) \\ &\quad = \mathbf{A}(2 \cdot \mathbf{S} \cdot \mathbf{G}^{-\mathbf{1}}(\tfrac{1}{2} \cdot \mathbf{G})) + x \cdot \mathbf{G} + 2 \cdot \mathbf{E} \cdot \mathbf{G}^{-1}(\tfrac{1}{2} \cdot \mathbf{G})\end{aligned}$$

Since $\mathbf{G}^{-1}(\frac{1}{2} \cdot \mathbf{G}) \in \{0,1\}^{w\times w}$, we have $|(\mathbf{E} \cdot \mathbf{G}^{-1}(\frac{1}{2} \cdot \mathbf{G}))[i,j]| \leq O(Bw)$.

We turn to the second half of the correlation-inducing gate.

Lemma 2 (Multiply-by-one gate for dual-GSW). *Let* $\mathsf{ct}^* = \mathbf{AS}^* + \mathbf{G} + \mathbf{E}^*$ *be a dual-GSW encryption of the constant* 1*, where* $|\mathbf{E}^*[i,j]| \leq p/10$*. Let* $x \in \{0,1\}$ *and let* $\mathsf{ct}_x = \mathbf{AS} + x \cdot \mathbf{G} + \mathbf{E}$ *be a dual-GSW encryption of* x *such each entry* $\mathbf{E}[i,j] = 2\mathbf{E}'[i,j]$ *where* $|\mathbf{E}'[i,j]| \leq p/(100w)$*. Let* $g(x,y) = x \cdot y$*. Then* $\mathsf{Eval}(g, \mathsf{ct}_x, \mathsf{ct}^*) = \mathbf{AS}' + x \cdot \mathbf{G} + \mathbf{E}'$*, where* $\mathbf{S}' \in \mathbb{Z}_p^{n\times w}$ *and* $\mathbf{E}'[1,1] = x \cdot \mathbf{E}^*[1,1] \mod 2$.[5]

Proof. We observe that

$$\begin{aligned}\mathsf{Eval}(g, \mathsf{ct}_x, \mathsf{ct}^*) &= (\mathbf{AS} + x \cdot \mathbf{G} + \mathbf{E}) \cdot \mathbf{G}^{-1}(\mathbf{AS}^* + \mathbf{G} + \mathbf{E}^*) \\ &= \mathbf{A}(\mathbf{SG}^{-1}(\mathbf{AS}^* + \mathbf{G} + \mathbf{E}^*) + x\mathbf{S}^*) \\ &\quad + x \cdot \mathbf{G} + (x \cdot \mathbf{E}^* + \mathbf{EG}^{-1}(\mathbf{AS}^* + \mathbf{G} + \mathbf{E}^*)) .\end{aligned}$$

The entries of $\mathbf{EG}^{-1}(M)$ for any matrix M are at most $p/100$ in magnitude. The lemma follows.

[5] For two field elements $a, b \in \mathbb{Z}_q$, we write $a = b \mod 2$ if this holds in the embedding of $\mathbb{Z}_q$ into the integers $[-\lceil q/2\rceil, \lfloor q/2\rfloor]$.

From Lemmas 1 and 2 we have the following corollary, capturing the correlation-inducing gate for dual-GSW. The gate takes x, multiplies by the constant $1/2$, adds the result to itself, and multiplies by the constant 1. Homomorphically evaluated, this operation introduces correlation between x and the error part of the output ciphertext.

Corollary 1. *Let* $\mathsf{ct}^* = \mathbf{AS}^* + \mathbf{G} + \mathbf{E}^*$ *be a dual-GSW encryption of the constant* 1*, where* $|\mathbf{E}^*[i,j]| \leq p/10$ *for all* i, j*. Let* $x \in \{0,1\}$ *and let* $\mathsf{ct}_x = \mathbf{AS} + x\mathbf{G} + \mathbf{E}$ *be a dual-GSW encryption of* x *such that* $|\mathbf{E}[i,j]| \leq q/\mathsf{poly}(m, \log q)$ *for all* i, j*. Then, for* g_1, g_2 *as in Lemma 1 and* g *as in Lemma 2,*

$$\mathsf{Eval}(g, \mathsf{Eval}(g_2, \mathsf{Eval}(g_1, \mathsf{ct}_x)), \mathsf{ct}^*) = \mathbf{AS}' + x\mathbf{G} + \mathbf{E}'$$

where $\mathbf{E}'[1,1] = x\mathbf{E}^*[1,1] \mod 2$*.*

In both Lemma 2 and Corollary 1, we actually have the stronger conclusion that $\mathbf{E}' = x \cdot \mathbf{E}^* \mod 2$, rather than just the $[1,1]$ entry – however, we will only use the weaker conclusion for the $[1,1]$ entry.

4.2 Correlation-Inducing Gate for GSW

We now state the following fact as a lemma, which follows directly from the properties of homomorphic evaluation of the GSW ciphertexts.

Lemma 3 (Multiply-by-zero). *Let* $\mathsf{ct}^* = \mathbf{UR}^*$ *be a GSW encryption of the constant* 0*. Let* $\mathsf{ct}_x = \mathbf{UR} + x \cdot \mathbf{G}$ *be a GSW encryption of a bit* $x \in \{0,1\}$*, where* $|\mathbf{R}[i,j]|, |\mathbf{R}^*[i,j]| \leq B$*, for all* i, j*. Let* g *be the multiplication gate. Let* $\widehat{\mathsf{ct}} = \mathsf{Eval}(g, \mathsf{ct}_x, \mathsf{ct}^*) = \mathbf{UR}'$*. Then* $\mathbf{R}' = \mathbf{RG}^{-1}(\mathsf{ct}^*) + x\mathbf{R}^*$*. Further, for all* i, j $|\mathbf{R}'[i,j]| = O(B \cdot w)$

We will use this structure of the multiplication by 0 operation to counterexample to 2-circ SRL security.

5 Counter Example to 2-Circular SRL Security

In this section we show that the GSW encryption scheme provided in the Section 3.1 serves as a counterexample to 2-Circ SRL security.

5.1 Counter Example Details

We prove the following theorem:

Theorem 1. *Let* PKE *be any encryption scheme where the depth of the decryption circuit is* $d'(\lambda)$ *for some polynomial* d'*. Let* FHE_d *be the GSW fully-homomorphic encryption scheme described in Section 3.1 for the circuit class* $\mathcal{C}_{\lambda,d}$ *where* $d > d' + \lambda$*, then,* $(\mathsf{FHE}_d, \mathsf{PKE})$ *are not* 2-Circ-SRL *secure.*

We show an explicit polynomial time adversary attacking the 2-Circ-SRL- secure scheme. Below, we write down the interaction between the challenger and adversary $\mathcal{A}$ in the security game and then we prove that the adversary wins with constant (better than 1/2) probability.

1. The challenger runs $\mathsf{PKE.Setup}(1^\lambda) \to (\mathsf{pk}_2, \mathsf{sk}_2)$ and $\mathsf{FHE.Setup}(1^\lambda, 1^d) \to (\mathsf{pk}_1, \mathsf{sk}_1)$. Here pk_1 is a matrix $\mathbf{U}$ and the secret key sk_1 is a vector such that $(-\mathsf{sk}_1, 1) \cdot \mathbf{U} = \boldsymbol{e}$ where $\boldsymbol{e}$ was the errors sampled from $\chi^{1\times m}$. The ciphertexts live in $\mathbb{F}_p^{n\times w}$ and all dimensions n, m and w are polynomial in λ. As a consequence given the secret key, for any ciphertext $\mathbf{US}$ encrypting 0 with randomness $\mathbf{S}$, one can compute $\boldsymbol{e} \cdot \mathbf{S}$ by multiplying with the secret key.

2. The adversary submits two messages $\boldsymbol{m}^0, \boldsymbol{m}^1 \in \{0,1\}^{\lambda+1}$. Here, $\boldsymbol{m}^\beta = (\beta, 0, \ldots, 0)$ for $\beta \in \{0,1\}$. The challenger samples $\beta \leftarrow \{0,1\}$ and lets $\boldsymbol{m}^* = (\boldsymbol{m}^\beta \| \mathsf{sk}_2)$. Denote by ℓ the size of sk_2.

3. The challenger computes $\mathsf{ct}_1 = (\mathsf{ct}_{1,1}, \ldots, \mathsf{ct}_{1,\ell+2})$ and ct_2 as follows. For $j \in [2+\ell]$, compute $\mathsf{ct}_{1,j} = \mathsf{FHE.Enc}(\mathsf{pk}_1, \boldsymbol{m}^*_j; \mathbf{R}_{1,j})$ where $\mathbf{R}_{1,j}$ is chosen as in the scheme. It also computes $\mathsf{ct}_2 = \mathsf{PKE.Enc}(\mathsf{pk}_2, \mathsf{sk}_1)$. Both $\mathsf{ct}_1, \mathsf{ct}_2$ are given to the adversary $\mathcal{A}$. Each $\mathsf{ct}_{1,j} \in \mathbb{F}_p^{n\times w}$ and $\mathbf{G}^{-1}(\mathsf{ct}_{1,j}) \in \{0,1\}^{w\times w}$.

4. The adversary finds at random an index $j_v \in [2, \lambda+1]$ (which is an index for which ct_{1,j_v} encrypts 0) such that there exists a vector $\boldsymbol{v} \in \{0,1\}^{w\times 1}$ such that $\mathbf{G}^{-1}(\mathsf{ct}_{1,j_v})\boldsymbol{v} = \mathbf{0}^{w\times 1} \mod 2$. This can be done with overwhelming probability because each $\mathbf{G}^{-1}(\mathsf{ct}_{1,j})$ for $j \in [2, \lambda+1]$ is rank deficient with probability at least $\frac{1}{2} - \mathsf{negl}(\lambda)$.

5. Use ct_1 and ct_2 to compute $\mathsf{ct}_{\mathsf{sk}_1}$ which is an FHE encryption of sk_1.

6. The adversary now submits $q = \lambda \cdot m$ functions, value tuples $(f_i, 0)$ for $i \in [q]$. For query $i \in [q]$, the function f_i is described below. The function f_i is described in terms of the FHE_d evaluation directly. The underlying boolean function can be inferred from the FHE_d evaluation. The function description depends on $\mathsf{ct}_1, \mathsf{ct}_{\mathsf{sk}_1}$, $\mathsf{ct}^*_i = \mathsf{Enc}^*(\mathsf{pk}_1, 0; \mathbf{R}^*_i)$ which is the i^{th} sampled extra noisy ciphertext, the vector $\boldsymbol{v}$ and the index j_v. For every $i \in [q]$, the adversary receives $\mathsf{leak}_i = \mathbf{R}^*_i - \widehat{\mathbf{R}}_i$ where $\widehat{\mathbf{R}}_i$ is the randomness in the evaluated ciphertext computed for computing f_i.

7. The adversary simply finds the dimension of the space $W = \{\boldsymbol{y} \in \{0,1\}^{1\times w} | \boldsymbol{y} \cdot (\mathsf{leak}_i \cdot v) = 0 \mod 2 \ \forall\ i \in [q]\}$ over $\mathbb{F}_2^m$. If the dimension is 0, output the guess $\beta' = 0$, otherwise output $\beta' = 1$.

Function $\mathsf{FHE}_d.\mathsf{Eval}(f_i, \cdot)$

Input: ct_1
Hardwired: $\mathsf{ct}_{\mathsf{sk}_1}$, $\mathsf{ct}_i^*, \boldsymbol{v}$

1. Compute $\mathsf{ct}'_i = \mathbf{U}\mathbf{R}'_i + (\beta \cdot \langle \boldsymbol{e}, \mathbf{R}_i^* \cdot \boldsymbol{v} \rangle \mod 2)\mathbf{G}$. This is computable because given the secret key sk_1 of the FHE_d, one can compute $\boldsymbol{e}\mathbf{R}_i^*$ as pointed earlier, and we have encryption $\mathsf{ct}_{\mathsf{sk}_1}$ hardwired. Denote $\gamma_v = (\beta \cdot \langle \boldsymbol{e}, \mathbf{R}_i^* \cdot \boldsymbol{v} \rangle \mod 2)$
2. Multiply ct'_i with ct_{1,j_v} to get the following ciphertext (see Lemma 3).

$$\widehat{\mathsf{ct}}_i = \mathbf{U} \underbrace{(\mathbf{R}'_i \mathbf{G}^{-1}(\mathsf{ct}_{1,j_v}) + \gamma_v \mathbf{R}_{1,j_v})}_{\widehat{\mathbf{R}}_i}$$

3. Output $\widehat{\mathsf{ct}}_i$.

We now argue that the success probability of the adversary is almost 3/4. First of all, note that the adversary is admissible because f_i on $\boldsymbol{m}^*$ always outputs 0. This is ensured because in the step 2 of the circuit, ct'_i which computes γ_v is multiplied by ct_{j_v} (which encrypts 0). Hence the output is always 0.

Let's now analyze the depth of the circuit f_i. Encryption of $\mathsf{ct}_{\mathsf{sk}_1}$ can be computed by a circuit that is computable in depth d' (which is the decryption circuit depth of PKE). The second step is in NC^1, because $((-1, \mathsf{sk}_1) \cdot \mathsf{ct}_{i^*}) \mod 2 = \mathbf{e}\mathbf{R}_i^* \mod 2$. Finally, the last step consists of taking the resulting vector's inner product with $\beta \cdot \boldsymbol{v} \mod 2$, which can also be done in NC^1. So, if $d > d' + \lambda$, the function f_i is computable in depth d.

Now we analyze the success probability of this attack. Observe that since $\mathbf{G}^{-1}(\mathsf{ct}_{1,j})$ for all $j \in [2, \lambda + 1]$ behave pseudorandomly, with probability at least $0.5 - \mathsf{negl}(\lambda)$, a given $\mathsf{ct}_{1,j}$ is going to have a vector in the nullspace (the determinant of a random matrix over $\mathbb{F}_2$ is random over $\mathbb{F}_2$). Thus, point 4) succeeds with probability at least $1 - \mathsf{negl}(\lambda)$. Now let us analyze the randomness of the evaluated ciphertext during each step of the evaluation.

1. $\mathsf{leak}_i = \mathbf{R}_i^* - \widehat{\mathbf{R}}_i$. Remember, $\widehat{\mathbf{R}}_i = \mathbf{R}'_i \cdot \mathbf{G}^{-1}(\mathsf{ct}_{1,j_v}) + \gamma_v \mathbf{R}_{1,j_v}$.
2. The last step computes $\mathsf{leak}_i \cdot \boldsymbol{v} \mod 2$ which produces:

$$\mathsf{leak}_i \cdot \boldsymbol{v} = \mathbf{R}_i^* \cdot \boldsymbol{v} - \gamma_v \mathbf{R}_{1,j_v} \boldsymbol{v} \mod 2.$$

   This is because $\mathbf{G}^{-1}(\mathsf{ct}_{1,j_v})\boldsymbol{v} = \mathbf{0} \mod 2$.
3. If $\beta = 0$, $\gamma_v = 0$ and thus $\mathsf{leak}_i \cdot \boldsymbol{v} = \mathbf{R}_i^* \cdot \boldsymbol{v}$. Since $\boldsymbol{v}$ is independent of $\mathbf{R}_i^*$, $\mathbf{R}_i^* \cdot \boldsymbol{v} \mod 2$ is distributed identically like a random vector over $\mathbb{F}_2$. Since $q = m \cdot \lambda$, with probability $1 - \mathsf{negl}(\lambda)$, no non-zero vector $\boldsymbol{y} \mod 2 \in \mathbb{F}_2^{1 \times m}$, can satisfy $\boldsymbol{y} \cdot \mathbf{R}_i^* \boldsymbol{v} \mod 2 = 0$ for all i. This can be shown by computing the probability of a fixed $\boldsymbol{y}$ to satisfy all q independent equations, which is 2^{-q}, and then doing a union bound over all 2^m choices of $\boldsymbol{y}$.

4. If $\beta = 1$, $\gamma_v = \langle \boldsymbol{e}, \mathbf{R}_i^* \cdot \boldsymbol{v} \rangle \mod 2$ and thus $\mathsf{leak}_i \cdot \boldsymbol{v} = \mathbf{R}_i^* \cdot \boldsymbol{v} + \gamma_v \mathbf{R}_{1,j_v} \boldsymbol{v}$. Since $\boldsymbol{e}$ is independent of $\mathbf{R}_{1,j_v}$ and $\boldsymbol{v}$ and further $\mathbf{R}_{1,j_v} \cdot \boldsymbol{v} \neq \mathbf{0}$ (with probability $1 - \mathsf{negl}(\lambda)$ as ct_{1,j_v} is lossy for $\mathbf{R}_{1,j_v}$), it holds that with probability $0.5 - \mathsf{negl}(\lambda)$, $\langle e, \mathbf{R}_{1,j_v} \boldsymbol{v} \rangle \mod 2 = 1$. In this case, we have that at least the vector $\boldsymbol{e} \mod 2$ is a solution of:

$$0 = \boldsymbol{y} \cdot \mathsf{leak}_i \cdot \boldsymbol{v} = \boldsymbol{y} \mathbf{R}_i^* \cdot \boldsymbol{v} - (\langle \boldsymbol{e}, \mathbf{R}_i^* \cdot \boldsymbol{v} \rangle \mod 2)\ \boldsymbol{y} \cdot \mathbf{R}_{1,j_v} \boldsymbol{v} \mod 2,$$

for every $i \in [q]$. This can be seen by substituting $\boldsymbol{e}$ for $\boldsymbol{y}$. Hence dimension of W is at least 1, with probability $0.5 - \mathsf{negl}(\lambda)$.

Thus the probability of guessing β correctly is:

$$\begin{aligned}
&\frac{1}{2}(\Pr[\mathsf{Dim}(W) = 0 | \beta = 0] + \Pr[\mathsf{Dim}(W) > 0 | \beta = 1]) \\
&\geq \frac{1}{2}(1 - \mathsf{negl}(\lambda) + 0.5 - \mathsf{negl}(\lambda)) (\text{from the observations above}) \\
&\geq \frac{3}{4} - \mathsf{negl}(\lambda)
\end{aligned}$$

This concludes the proof.

6 Counter Example for the Conjecture by Wee-Wichs

Now we describe our counterexample to the conjecture of Wee and Wichs [44].

6.1 Homomorphic Pseudorandom LWE Samples Conjecture

The following presentation closely follows Section 6 of [44] – for additional context on the use of these definitions and conjecture to construct an oblivious LWE sampler and then iO, we refer the reader to [44].

For some parameters λ, n, m, p, Q, we will define two distributions over tuples of the form $(\{\boldsymbol{b_i}\}_{i \in [Q]}, \mathbf{A}, \mathbf{C}, \{\boldsymbol{s_i}\}_{i \in [Q]})$, where $\boldsymbol{b_i} \in \mathbb{Z}_p^m$, $\mathbf{A} \in \mathbb{Z}_p^{m \times n}$, $\mathbf{C} = \mathbf{C}_1, \ldots, \mathbf{C}_{\lambda+1}$ with $\mathbf{C_i} \in \mathbb{Z}_p^{m \times m \log q}$, and $\boldsymbol{s_i} \in \mathbb{Z}_p^n$. The conjecture will be that these distributions are computationally indistinguishable.

We first need some additional setup.

Setup for pseudorandom error distribution:

- Let χ_{prf} be a distribution on $\mathbb{Z}_p$.
- Let D be an algorithm which takes v random coins in $\{0,1\}$ and outputs samples $\boldsymbol{s} \leftarrow \mathbb{Z}_p^n$ and $\boldsymbol{e} \leftarrow \chi_{\mathsf{prf}}^m$.
- Let $\mathsf{PRF} : \{0,1\}^\lambda \times \{0,1\}^* \to \{0,1\}^v$ be a pseudo-random function.

Setup for pseudorandomly generating LWE samples:

- For $i \in [Q]$, $\boldsymbol{b} \in \mathbb{Z}_p^m$, let $g_{i,\boldsymbol{b},\mathbf{A}}$ be a circuit with values $(i, \boldsymbol{b}, \mathbf{A})$ hard-coded and which performs the following computation on input $(k, \beta) \in \{0,1\}^{\lambda+1}$:

$$\text{Let } (\boldsymbol{s}_i^{\mathsf{prf}}, \boldsymbol{e}_i^{\mathsf{prf}}) = \mathsf{D}(\mathsf{PRF}(k, i)). \text{ Output } \mathbf{A}\boldsymbol{s}_i^{\mathsf{prf}} + \boldsymbol{e}_i^{\mathsf{prf}} + \beta \cdot \boldsymbol{b}$$

Following [44], we now define two distributions $\mathsf{DIST}(\beta)$ for $\beta \in \{0,1\}$ as follows. Let χ be a B-bounded distribution.

- For $i \in [Q]$, generate LWE samples $\boldsymbol{b_i}$. Concretely, $\mathbf{A} \leftarrow \mathbb{Z}_p^{m \times n}$, $\widehat{\boldsymbol{s_i}} \leftarrow \mathbb{Z}_p^n$, $\widehat{\boldsymbol{e_i}} \leftarrow \widehat{\chi}^m$.
- Let $k \leftarrow \{0,1\}^\lambda$ and sample dual-GSW commitments $\mathbf{C}_1, \ldots, \mathbf{C}_\lambda$ to $k_1, \ldots, k_\lambda$. That is, sample $\mathbf{R}_i \leftarrow \mathbb{Z}_p^{n \times w}$ and $\mathbf{E}_i \leftarrow \chi^{m \times w}$ and set $\mathbf{C}_i = \mathbf{A}\mathbf{R}_i + k_i\mathbf{G} + \mathbf{E}_i$.
- Let $\mathbf{C}_\beta = \mathbf{A}\mathbf{R}_\beta + \beta\mathbf{G} + \mathbf{E}_\beta$ be a dual-GSW commitment to β.
- For $i \in [Q]$, let $(\boldsymbol{s}_i^{\mathsf{prf}}, \boldsymbol{e}_i^{\mathsf{prf}}) = \mathsf{D}(\mathsf{PRF}(k, i))$.
- Let $(\boldsymbol{r}_i^{\mathsf{Eval}}, \boldsymbol{e}_i^{\mathsf{Eval}}) = \mathsf{Eval}_{open,packed}(g_{i,\mathbf{A}\widehat{\boldsymbol{s}}_i + \widehat{\boldsymbol{e}}_i, \mathbf{A}}, \mathbf{A}, (k, \beta), \mathbf{R}, \mathbf{E})$.
- Let $\boldsymbol{s}_i = \boldsymbol{r}_i^{\mathsf{Eval}} + \boldsymbol{s}_i^{\mathsf{prf}} + \beta\widehat{\boldsymbol{s}}_i$.
- Output $(\{\mathbf{A}\widehat{\boldsymbol{s}}_i + \widehat{\boldsymbol{e}}_i\}_{i \in [Q]}, \mathbf{A}, \mathbf{C}_1, \ldots, \mathbf{C}_\lambda, \mathbf{C}_\beta, \{\boldsymbol{s}_i\}_{i \in [Q]})$.

Conjecture 1 (HPLS Conjecture, [44] Conjecture 6.4). Let λ be a security parameter and $n, m, q, \chi, \widehat{\chi}, \chi^{\mathsf{prf}}$ be such that the LWE assumption holds with parameters (n, q, χ) and with $(n, q, \widehat{\chi})$. Furthermore, suppose that χ^{prf} smudges out error of size $\widehat{B} + B \cdot m^{O(t)}$, where t is the depth of the circuit $g_{i,\boldsymbol{b},\mathbf{A}}$ (which is dominated by the depth of PRF). Then there is a choice of PRF such that $\mathsf{DIST}(0)$ and $\mathsf{DIST}(1)$ are computationally indistinguishable.

6.2 Counter Example Details

In our main theorem, we make the following assumption about implementation details of the circuit $g_{i,\boldsymbol{b},\mathbf{A}}$ which are left unspecified in [44]. We assume there is a Boolean circuit C_{PRF} computing the pseudorandom function and another Boolean circuit C_{D} implementing the sampling algorithm D, whose outputs are the binary expansion of $(\boldsymbol{s}_i^{\mathsf{prf}}, \boldsymbol{e}_i^{\mathsf{prf}})$. Then g is given by

1. composing the circuits $C_{\mathsf{PRF}}, C_{\mathsf{D}}$,
2. multiplying by field elements in $\mathbb{Z}_p$ and adding to compute each entry of the vector $\mathbf{A}\boldsymbol{s}_i^{\mathsf{prf}} + \boldsymbol{e}_i^{\mathsf{prf}}$,
3. multiplying the input β by the field element $\boldsymbol{b}[j]$ for $j \leq m$, and
4. adding (2) and (3) to obtain the final outputs. $(\mathbf{A}\boldsymbol{s}_i^{\mathsf{prf}})[j] + \boldsymbol{e}_i[j] + \beta \cdot \boldsymbol{b}[j]$, for $j \in [m]$. These outputs are packed into a vector by Eval.

We prove the following theorem:

Theorem 2. *With $\lambda, n, m, q, \chi, \widehat{\chi}, \chi^{\mathsf{prf}}$ as in Conjecture 1, for any sampling algorithm D as above there is an arithmetic circuit C_{D} over $\mathbb{Z}_p$ implementing D such that for any function $F : \{0,1\}^\lambda \times \{0,1\}^* \to \{0,1\}^v$, if $Q \gg m^2 \log q$ then the resulting distributions $\mathsf{DIST}(0)$ and $\mathsf{DIST}(1)$ are distinguishable with nontrivial probability in polynomial time.*

Remark 3. A somewhat easier argument than the below shows the same result if we allow ourselves nonstandard implementations of parts (2) and (3) of $g_{i,\boldsymbol{b},\mathbf{A}}$. A merit of our attack is that it allows any sampling algorithm D and any choice of PRF, and requires only a careful choice of circuit C_{D} to implement D.

The circuit C_{D} We start by describing the circuit C_{D}. Start with any circuit C for D, with output in binary, and modify it in the following way.

- Let C_0 be a circuit which takes the first bit x_1 of its input and performs the computation $x_1 + (1 + (q-1) \cdot x_1))$. (The output wire of C_0 therefore always carries the value 1.)
- To the output wire of C_0 corresponding to the lowest-order bit of $\boldsymbol{e}_i^{\mathsf{prf}}[1]$, attach a new gate which performs the correlation-inducing transformation described in Corollary 1, using the output wire of C_0 as the special "1" input.
- To all of the other output wires, attach a gate performing the "even-noise" transformation of Lemma 1. (As noted before, these gates can be implemented using only multiplication and addition of boolean values and multiplication by a field element.) The result is the new circuit C_{D}.

A Linear System Part (3) above of the circuit computing $g_{i,\boldsymbol{b},\mathbf{A}}$ induces a linear system in $m \times m \log q$ variables E_{ij}, in the following way. On input $C_\beta(E) = \mathbf{A}\mathbf{S}_\beta + \beta\mathbf{G} + E$, part (3) of that circuit, evaluated homomorphically, produces outputs of the form $\mathbf{A}\mathbf{S}\mathbf{G}^{-1}(\boldsymbol{b}_i[j]) + \beta \boldsymbol{b}_i[j]\mathbf{G} + E\mathbf{G}^{-1}(\boldsymbol{b}_i[j])$. Let $L_{\boldsymbol{b}_i[j]}(E)$ be the matrix of linear functions given in E given by $E\mathbf{G}^{-1}(\boldsymbol{b}_i[j])$. Let $L_i(E)$ be the linear function in E given by the $L_{\boldsymbol{b}_i[j]}(E)[1,1]$.

Distinguishing algorithm Now we describe an algorithm to distinguish $\mathsf{DIST}(0)$ and $\mathsf{DIST}(1)$ with the above choice of F.

Input: $(\{\boldsymbol{b}_i\}_{i\in[Q]}, \mathbf{A}, \mathbf{C}_1, \ldots, \mathbf{C}_\lambda, \mathbf{C}_\beta, \{\boldsymbol{s}_i\}_{i\in[Q]})$,

1. Using the commitments $\mathbf{C}_1, \ldots, \mathbf{C}_\lambda, \mathbf{C}_\beta$, homomorphically evaluate the circuits $g_{i,\boldsymbol{b}_i,\mathbf{A}}$ to obtain (packed) ciphertexts $\mathsf{ct}_1, \ldots, \mathsf{ct}_Q \in \mathbb{Z}_p^m$.
2. Compute vectors $\boldsymbol{e}'_i = \mathsf{ct}_i - \mathbf{A}\boldsymbol{s}_i$. Let e_i be the first entry of $\boldsymbol{e}'_i$.
3. Check if the linear system in $m \times m \log q$ variables E given by the equations $L_i(E) = e_i$ has a solution over $\mathbb{F}_2$. If it does, output "$\beta = 0$". Otherwise, output a random $\beta \in \{0, 1\}$.

Proof (Proof of Theorem 2). To prove the theorem it will be enough to show that the linear system $L_i(E) = e_i$ has a solution with probability $\Omega(1)$ when the underlying distribution is $\mathsf{DIST}(0)$, but has a solution only with probability $o(1)$ when the underlying distribution is $\mathsf{DIST}(1)$.

We start by examining the structure of e_i in both the $\beta = 0$ and $\beta = 1$ cases. We first expand ct_i.

$$\mathsf{ct}_i = \mathbf{A}\boldsymbol{r}_i^{\mathsf{Eval}} + \mathbf{A}\boldsymbol{s}_i^{\mathsf{prf}} + \boldsymbol{e}_i^{\mathsf{prf}} + \beta \cdot (\mathbf{A}\widehat{\boldsymbol{s}}_i + \widehat{\boldsymbol{e}}_i) + \boldsymbol{e}_i^{\mathsf{Eval}}\,.$$

Then

$$\boldsymbol{e}'_i = \mathsf{ct}_i - \mathbf{A}\boldsymbol{s}_i = \boldsymbol{e}_i^{\mathsf{prf}} + \boldsymbol{e}_i^{\mathsf{Eval}} + \beta \cdot \widehat{\boldsymbol{e}}_i\,.$$

Furthermore, $\boldsymbol{e}_i^{\mathsf{Eval}}$ has two parts, coming from parts (2) and (3) of the circuit for g – let us call $\boldsymbol{e}_i^{\mathsf{Eval}^2}$ the part coming from step (2) and $\boldsymbol{e}_i^{\mathsf{Eval}^3}$ the part from step (3), so that $\boldsymbol{e}_i^{\mathsf{Eval}} = \boldsymbol{e}_i^{\mathsf{Eval}^2} + \boldsymbol{e}_i^{\mathsf{Eval}^3}$ and

$$\boldsymbol{e}'_i = \boldsymbol{e}_i^{\mathsf{prf}} + \boldsymbol{e}_i^{\mathsf{Eval}^2} + \boldsymbol{e}_i^{\mathsf{Eval}^3} + \beta \cdot \widehat{\boldsymbol{e}}_i$$

We claim that with probability $\Omega(1)$, all $i \in [Q]$ satisfy $\boldsymbol{e}_i^{\mathsf{prf}}[1] = \boldsymbol{e}_i^{\mathsf{Eval}^2}[1] \mod 2$, as a result of our design of the circuit C_{D}. In fact, we claim that this occurs whenever the (random) commitment $\mathbf{C}_1$ to k_1 is such that homomorphically evaluating the circuit C_0 yields a ciphertext $\mathbf{AS}^* + k_1 \cdot \mathbf{G} + \mathbf{E}^*$ such that $\mathbf{E}^*[1,1]$ is odd, which occurs with probability $1/2$. This follows directly from Lemmas 1 and 2, Corollary 1, and our assumption about the structure of the circuit g. Together, these ensure that:

- the noise coming from homomorphically evaluating $\mathbf{A}\boldsymbol{s}^{\mathsf{prf}}$ is all $0 \mod 2$, and
- the noise coming from homomorphically evaluating $\mathbf{e}_i^{\mathsf{prf}}[1]$ has upper-left entry equal to $0 \mod 2$ (this entry is preserved by packing).

We conclude that, on the event above, $e_i = \boldsymbol{e}_i^{\mathsf{Eval}^3} + \beta \cdot \widehat{\boldsymbol{e}}_i \mod 2$. Now, if $\beta = 0$, observe that $\boldsymbol{e}_i^{\mathsf{Eval}_3} = L_i(E)$, the linear function described above in $m \times m \log q$ variables E. So the linear system has a solution. Finally, if $\beta = 1$, since $Q \gg m^2 \log q$ the linear system is whp overdetermined, and $e_i \mod 2$ is independent of the coefficients of L_i and independent of other e_i', because of the presence of the random vector $\widehat{e}_i$. So whp the linear system is unsatisfiable.

7 Acknowledgements

Sam Hopkins was supported by the Miller Institue, UC Berkeley.

Aayush Jain was supported by a Google PhD fellowship in the area of security and privacy (2018) and in part from DARPA SAFEWARE and SIEVE awards, NTT Research, NSF Frontier Award 1413955, and NSF grant 1619348, BSF grant 2012378, a Xerox Faculty Research Award, a Google Faculty Research Award, an equipment grant from Intel, and an Okawa Foundation Research Grant. This material is based upon work supported by the Defense Advanced Research Projects Agency through Award HR00112020024 and the ARL under Contract W911NF-15-C- 0205.

Huijia Lin was supported by NSF grants CNS-1528178, CNS-1929901, CNS-1936825 (CAREER), CNS-2026774, a Hellman Fellowship, a JP Morgan AI Research Award, a Simons Collaboration grant on the Theory of Algorithmic Fairness, the Defense Advanced Research Projects Agency (DARPA) and Army Research Office (ARO) under Contract No. W911NF-15-C-0236, and a subcontract No. 2017-002 through Galois.

References

1. Acar, T., Belenkiy, M., Bellare, M., Cash, D.: Cryptographic agility and its relation to circular encryption. In: Gilbert, H. (ed.) EUROCRYPT 2010. LNCS, vol. 6110, pp. 403–422. Springer, Heidelberg (May / Jun 2010). https://doi.org/10.1007/978-3-642-13190-5$_2$1

2. Agrawal, S.: Indistinguishability obfuscation without multilinear maps: New methods for bootstrapping and instantiation. In: Ishai, Y., Rijmen, V. (eds.) EUROCRYPT 2019, Part I. LNCS, vol. 11476, pp. 191–225. Springer, Heidelberg (May 2019). https://doi.org/10.1007/978-3-030-17653-2$_7$
3. Agrawal, S., Pellet-Mary, A.: Indistinguishability obfuscation without maps: Attacks and fixes for noisy linear FE. In: Canteaut, A., Ishai, Y. (eds.) Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12105, pp. 110–140. Springer (2020)
4. Ananth, P., Jain, A., Lin, H., Matt, C., Sahai, A.: Indistinguishability obfuscation without multilinear maps: New paradigms via low degree weak pseudorandomness and security amplification. In: Boldyreva, A., Micciancio, D. (eds.) CRYPTO 2019, Part III. LNCS, vol. 11694, pp. 284–332. Springer, Heidelberg (Aug 2019). https://doi.org/10.1007/978-3-030-26954-8$_1$0
5. Ananth, P., Jain, A., Sahai, A.: Indistinguishability obfuscation without multilinear maps: io from lwe, bilinear maps, and weak pseudorandomness. IACR Cryptology ePrint Archive **2018**, 615 (2018)
6. Ballard, L., Green, M., de Medeiros, B., Monrose, F.: Correlation-resistant storage via keyword-searchable encryption. Cryptology ePrint Archive, Report 2005/417 (2005), `http://eprint.iacr.org/2005/417`
7. Barak, B., Goldreich, O., Impagliazzo, R., Rudich, S., Sahai, A., Vadhan, S.P., Yang, K.: On the (im)possibility of obfuscating programs. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 1–18. Springer, Heidelberg (Aug 2001). https://doi.org/10.1007/3-540-44647-8$_1$
8. Bishop, A., Hohenberger, S., Waters, B.: New circular security counterexamples from decision linear and learning with errors. In: Iwata, T., Cheon, J.H. (eds.) ASIACRYPT 2015, Part II. LNCS, vol. 9453, pp. 776–800. Springer, Heidelberg (Nov / Dec 2015). https://doi.org/10.1007/978-3-662-48800-3$_3$2
9. Black, J., Rogaway, P., Shrimpton, T.: Encryption-scheme security in the presence of key-dependent messages. IACR Cryptol. ePrint Arch. **2002**, 100 (2002), `http://eprint.iacr.org/2002/100`
10. Brakerski, Z., Döttling, N., Garg, S., Malavolta, G.: Candidate iO from homomorphic encryption schemes. In: Rijmen, V., Ishai, Y. (eds.) EUROCRYPT 2020, Part I. pp. 79–109. LNCS, Springer, Heidelberg (May 2020). https://doi.org/10.1007/978-3-030-45721-1$_4$
11. Brakerski, Z., Döttling, N., Garg, S., Malavolta, G.: Factoring and pairings are not necessary for io: Circular-secure LWE suffices. IACR Cryptol. ePrint Arch. **2020**, 1024 (2020), `https://eprint.iacr.org/2020/1024`
12. Brakerski, Z., Gentry, C., Vaikuntanathan, V.: (Leveled) fully homomorphic encryption without bootstrapping. In: Goldwasser, S. (ed.) ITCS 2012. pp. 309–325. ACM (Jan 2012). https://doi.org/10.1145/2090236.2090262
13. Brakerski, Z., Gentry, C., Vaikuntanathan, V.: (leveled) fully homomorphic encryption without bootstrapping. In: Innovations in Theoretical Computer Science 2012, Cambridge, MA, USA, January 8-10, 2012. pp. 309–325 (2012)
14. Brakerski, Z., Halevi, S., Polychroniadou, A.: Four round secure computation without setup. In: Kalai, Y., Reyzin, L. (eds.) TCC 2017, Part I. LNCS, vol. 10677, pp. 645–677. Springer, Heidelberg (Nov 2017). https://doi.org/10.1007/978-3-319-70500-2$_2$2

15. Brakerski, Z., Vaikuntanathan, V.: Efficient fully homomorphic encryption from (standard) LWE. In: Ostrovsky, R. (ed.) 52nd FOCS. pp. 97–106. IEEE Computer Society Press (Oct 2011). https://doi.org/10.1109/FOCS.2011.12
16. Camenisch, J., Lysyanskaya, A.: An efficient system for non-transferable anonymous credentials with optional anonymity revocation. In: Pfitzmann, B. (ed.) EUROCRYPT 2001. LNCS, vol. 2045, pp. 93–118. Springer, Heidelberg (May 2001). https://doi.org/10.1007/3-540-44987-6$_7$
17. Cash, D., Green, M., Hohenberger, S.: New definitions and separations for circular security. In: Fischlin, M., Buchmann, J., Manulis, M. (eds.) PKC 2012. LNCS, vol. 7293, pp. 540–557. Springer, Heidelberg (May 2012). https://doi.org/10.1007/978-3-642-30057-8$_3$2
18. Damgård, I., Jurik, M.: A generalisation, a simplification and some applications of Paillier's probabilistic public-key system. In: Kim, K. (ed.) PKC 2001. LNCS, vol. 1992, pp. 119–136. Springer, Heidelberg (Feb 2001). https://doi.org/10.1007/3-540-44586-2$_9$
19. Garg, S., Miles, E., Mukherjee, P., Sahai, A., Srinivasan, A., Zhandry, M.: Secure obfuscation in a weak multilinear map model. In: Hirt, M., Smith, A.D. (eds.) TCC 2016-B, Part II. LNCS, vol. 9986, pp. 241–268. Springer, Heidelberg (Oct / Nov 2016). https://doi.org/10.1007/978-3-662-53644-5$_1$0
20. Gay, R., Jain, A., Lin, H., Sahai, A.: Indistinguishability obfuscation from simple-to-state hard problems: New assumptions, new techniques, and simplification. IACR Cryptol. ePrint Arch. **2020**, 764 (2020)
21. Gay, R., Pass, R.: Indistinguishability obfuscation from circular security. In: Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2021. ACM (2021)
22. Gentry, C.: Fully homomorphic encryption using ideal lattices. In: Mitzenmacher, M. (ed.) 41st ACM STOC. pp. 169–178. ACM Press (May / Jun 2009). https://doi.org/10.1145/1536414.1536440
23. Gentry, C., Jutla, C.S., Kane, D.: Obfuscation using tensor products. Electronic Colloquium on Computational Complexity (ECCC) **25**, 149 (2018)
24. Gentry, C., Sahai, A., Waters, B.: Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013, Part I. LNCS, vol. 8042, pp. 75–92. Springer, Heidelberg (Aug 2013). https://doi.org/10.1007/978-3-642-40041-4$_5$
25. Goldreich, O.: Candidate one-way functions based on expander graphs. Electronic Colloquium on Computational Complexity (ECCC) **7**(90) (2000)
26. Gorbunov, S., Vaikuntanathan, V., Wichs, D.: Leveled fully homomorphic signatures from standard lattices. In: Servedio, R.A., Rubinfeld, R. (eds.) 47th ACM STOC. pp. 469–477. ACM Press (Jun 2015). https://doi.org/10.1145/2746539.2746576
27. Goyal, R., Koppula, V., Waters, B.: Separating semantic and circular security for symmetric-key bit encryption from the learning with errors assumption. In: Coron, J., Nielsen, J.B. (eds.) EUROCRYPT 2017, Part II. LNCS, vol. 10211, pp. 528–557. Springer, Heidelberg (Apr / May 2017). https://doi.org/10.1007/978-3-319-56614-6$_1$8
28. Green, M., Hohenberger, S.: Cpa and cca-secure encryption systems that are not 2-circular secure. IACR Cryptol. ePrint Arch. **2010**, 144 (2010)
29. Ishai, Y., Prabhakaran, M., Sahai, A.: Secure arithmetic computation with no honest majority. In: Reingold, O. (ed.) Theory of Cryptography, 6th Theory of Cryptography Conference, TCC 2009, San Francisco, CA, USA, March 15-17, 2009.

Proceedings. Lecture Notes in Computer Science, vol. 5444, pp. 294–314. Springer (2009)
30. Jain, A., Lin, H., Matt, C., Sahai, A.: How to leverage hardness of constant-degree expanding polynomials overa $\mathbb{R}$ to build $i\mathcal{O}$. In: Ishai, Y., Rijmen, V. (eds.) EUROCRYPT 2019, Part I. LNCS, vol. 11476, pp. 251–281. Springer, Heidelberg (May 2019). https://doi.org/10.1007/978-3-030-17653-2$_9$
31. Jain, A., Lin, H., Sahai, A.: Simplifying constructions and assumptions for $i\mathcal{O}$. IACR Cryptol. ePrint Arch. **2019**, 1252 (2019), `https://eprint.iacr.org/2019/1252`
32. Jain, A., Lin, H., Sahai, A.: Indistinguishability obfuscation from well-founded assumptions. In: Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2021. ACM (2021)
33. Koppula, V., Ramchen, K., Waters, B.: Separations in circular security for arbitrary length key cycles. In: Dodis, Y., Nielsen, J.B. (eds.) TCC 2015, Part II. LNCS, vol. 9015, pp. 378–400. Springer, Heidelberg (Mar 2015). https://doi.org/10.1007/978-3-662-46497-7$_1$5
34. Koppula, V., Waters, B.: Circular security separations for arbitrary length cycles from LWE. In: Robshaw, M., Katz, J. (eds.) CRYPTO 2016, Part II. LNCS, vol. 9815, pp. 681–700. Springer, Heidelberg (Aug 2016). https://doi.org/10.1007/978-3-662-53008-5$_2$4
35. Lin, H., Matt, C.: Pseudo flawed-smudging generators and their application to indistinguishability obfuscation. IACR Cryptology ePrint Archive **2018**, 646 (2018)
36. Marcedone, A., Orlandi, C.: Obfuscation $\Rightarrow$ (IND-CPA security $\not\Rightarrow$ circular security). In: Abdalla, M., Prisco, R.D. (eds.) SCN 14. LNCS, vol. 8642, pp. 77–90. Springer, Heidelberg (Sep 2014). https://doi.org/10.1007/978-3-319-10879-7$_5$
37. Micciancio, D., Peikert, C.: Trapdoors for lattices: Simpler, tighter, faster, smaller. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 700–718. Springer, Heidelberg (Apr 2012). https://doi.org/10.1007/978-3-642-29011-4$_4$1
38. Micciancio, D., Regev, O.: Worst-case to average-case reductions based on Gaussian measures. In: 45th FOCS. pp. 372–381. IEEE Computer Society Press (Oct 2004). https://doi.org/10.1109/FOCS.2004.72
39. Paillier, P.: Public-key cryptosystems based on composite degree residuosity classes. In: Stern, J. (ed.) EUROCRYPT'99. LNCS, vol. 1592, pp. 223–238. Springer, Heidelberg (May 1999). https://doi.org/10.1007/3-540-48910-X$_1$6
40. Peikert, C., Vaikuntanathan, V., Waters, B.: A framework for efficient and composable oblivious transfer. In: Wagner, D. (ed.) CRYPTO 2008. LNCS, vol. 5157, pp. 554–571. Springer, Heidelberg (Aug 2008). https://doi.org/10.1007/978-3-540-85174-5$_3$1
41. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. In: Gabow, H.N., Fagin, R. (eds.) 37th ACM STOC. pp. 84–93. ACM Press (May 2005). https://doi.org/10.1145/1060590.1060603
42. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. In: STOC. pp. 84–93 (2005)
43. Rothblum, R.: On the circular security of bit-encryption. In: Sahai, A. (ed.) TCC 2013. LNCS, vol. 7785, pp. 579–598. Springer, Heidelberg (Mar 2013). https://doi.org/10.1007/978-3-642-36594-2$_3$2
44. Wee, H., Wichs, D.: Candidate obfuscation via oblivious LWE sampling. In: Canteaut, A., Standaert, F. (eds.) Advances in Cryptology - EUROCRYPT 2021 -

40th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, October 17-21, 2021, Proceedings, Part III. Lecture Notes in Computer Science, vol. 12698, pp. 127–156. Springer (2021)
45. Wichs, D., Zirdelis, G.: Obfuscating compute-and-compare programs under LWE. In: Umans, C. (ed.) 58th FOCS. pp. 600–611. IEEE Computer Society Press (Oct 2017). https://doi.org/10.1109/FOCS.2017.61

A Lattice Preliminaries

Lattices. An m-dimensional lattice $\mathcal{L}$ is a discrete additive subgroup of $\mathbb{R}^m$ (not contained in any subspace of strictly smaller dimension). Given positive integers n, m, q and a matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$, we let $\Lambda_q^\perp(\mathbf{A})$ denote the lattice $\{\boldsymbol{x} \in \mathbb{Z}^m \mid \mathbf{A}\boldsymbol{x} = \mathbf{0} \bmod q\}$.

Discrete Gaussians. Let σ be any positive real number. The Gaussian distribution $\mathcal{D}_\sigma$ with parameter σ is defined by the probability distribution function $\rho_\sigma(\boldsymbol{x}) = \exp(-\pi\|x\|^2/\sigma^2)$. For any discrete set $\mathcal{L} \subseteq \mathbb{R}^m$, define $\rho_\sigma(\mathcal{L}) = \sum_{\boldsymbol{x} \in \mathcal{L}} \rho_\sigma(\boldsymbol{x})$. The discrete Gaussian distribution $\mathcal{D}_{\mathcal{L},\sigma}$ over $\mathcal{L}$ with parameter σ is defined by the probability distribution function $\rho_{\mathcal{L},\sigma}(\boldsymbol{x}) = \rho_\sigma(\boldsymbol{x})/\rho_\sigma(\mathcal{L})$.

The following lemma (e.g., [38, Lemma 4.4]) shows that if the parameter σ of a discrete Gaussian distribution is small, then any vector drawn from this distribution will be short (with high probability).

Lemma 4. *Let m, n, q be positive integers with $m > n$, $q > 2$. Let $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ be a matrix of dimensions $n \times m$, $\sigma \in \widetilde{\Omega}(n)$, and $\mathcal{L} = \Lambda_q^\perp(\mathbf{A})$. Then, there is a negligible function $\mathsf{negl}(\cdot)$ such that*

$$\Pr_{\boldsymbol{x} \leftarrow \mathcal{D}_{\mathcal{L},\sigma}} \left[\|\boldsymbol{x}\| > \sqrt{m}\sigma\right] \leq \mathsf{negl}(n),$$

where $\|\boldsymbol{x}\|$ denotes the ℓ_2 norm of $\boldsymbol{x}$.

Truncated Discrete Gaussians. The truncated discrete Gaussian distribution over $\mathbb{Z}^m$ with parameter σ, denoted by $\widetilde{\mathcal{D}}_{\mathbb{Z}^m,\sigma}$, is the same as the discrete Gaussian distribution $\mathcal{D}_{\mathbb{Z}^m,\sigma}$ except that it outputs 0 whenever the ℓ_∞ norm exceeds $\sqrt{m}\sigma$. By definition, we can say that $\widetilde{\mathcal{D}}_{\mathbb{Z}^m,\sigma}$ is $\sqrt{m}\sigma$-bounded, where a

family of distributions $\mathcal{D} = \{\mathcal{D}_\lambda\}_{\lambda \in \mathbb{N}}$ over the integers is *B-bounded* (for $B = B(\lambda) > 0$) if for every $\lambda \in \mathbb{N}$ it holds that $\Pr_{x \leftarrow \mathcal{D}_\lambda}[|x| \leq B(\lambda)] = 1$.

Also by 4, $\widetilde{\mathcal{D}}_{\mathbb{Z}^m,\sigma}$ and $\mathcal{D}_{\mathbb{Z}^m,\sigma}$ are statistically indistinguishable. Therefore, in the preliminaries below, unless specified, the lemmata will apply in the setting where by sampling from discrete Gaussian we mean sampling from truncated discrete Gaussian distribution.

A.1 Learning With Errors

The learning with errors (LWE) problem was defined by Regev [42]. The $\mathsf{LWE}_{n,m,q,\chi}$ problem for parameters $n, m, q \in \mathbb{N}$ and for a distribution χ supported over $\mathbb{Z}$ is to distinguish between the following pair of distributions

$$(\mathbf{A}, \boldsymbol{s}\mathbf{A} + \boldsymbol{e} \bmod q) \quad \text{and} \quad (\mathbf{A}, \boldsymbol{u}),$$

where $\mathbf{A} \leftarrow \mathbb{Z}_q^{n \times m}$, $\boldsymbol{s} \leftarrow \mathbb{Z}_q^{1 \times n}$, $\boldsymbol{e} \leftarrow \chi^{1 \times n}$ and $\boldsymbol{u} \leftarrow \mathbb{Z}_q^{1 \times m}$. Similarly, we can define the matrix version of the problem, which is known to be hard, if the version above is hard. Specifically, let $k \in \mathsf{poly}(n, m)$, then in the matrix the task is to distinguish between the following two distributions

$$(\mathbf{A}, \mathbf{S}\mathbf{A} + \mathbf{E} \bmod q) \quad \text{and} \quad (\mathbf{A}, \mathbf{U}),$$

where $\mathbf{A} \leftarrow \mathbb{Z}_q^{n \times m}$, $\mathbf{S} \leftarrow \mathbb{Z}_q^{k \times n}$, $\mathbf{E} \leftarrow \chi^{k \times n}$ and $\mathbf{U} \leftarrow \mathbb{Z}_q^{k \times m}$.

The gadget matrix [37]. Fix a dimension n and a modulus q. Define the gadget vector $\boldsymbol{g} = (1, 2, 4, \ldots, 2^{\lceil \log q \rceil - 1})$ and the gadget function $g^{-1} \colon \mathbb{Z}_q \to \{0,1\}^{\lceil \log q \rceil}$ to be the function that computes the $(\log q)$th bit decomposition of an integer. For some integer z the function is defined as $g^{-1}(z) = \boldsymbol{v} = (v_1, \ldots, v_{\log q})$ where $v_i \in \{0,1\}$ such that $z = \langle \boldsymbol{g}, \boldsymbol{v} \rangle$. By extension we define the augmented gadget function $G^{-1} \colon \mathbb{Z}_q^{n \times m} \to \{0,1\}^{(n \cdot \lceil \log q \rceil) \times m}$ to be the function that computes the $(\log q)$th bit decomposition of every integer in a matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$, and arranges them as a binary matrix of dimension $(n \cdot \lceil \log q \rceil) \times m$ which we denote $\boldsymbol{G}^{-1}(\mathbf{A})$. Hence, $\mathbf{G}_n \cdot G^{-1}(\boldsymbol{z}) = \mathbf{Z}$, where the gadget matrix $\mathbf{G}_n$ is $\mathbf{G}_n = \boldsymbol{g} \otimes \mathbf{I}_n \in \mathbb{Z}_q^{n \times (n \cdot \lceil \log q \rceil)}$. When n is clear from context, we denote $\mathbf{G}_n$ simply by $\mathbf{G}$.