Designs for practical SHE schemes based on RLWR

Madalina Bolboceanu, Anamaria Costache, Erin Hales, Rachel Player, Miruna Rosca, and Radu Titiu

Talk Outline

- What is Ring Learning with Rounding (RLWR)?
- Motivation
- Overview
- Ring Learning with Errors (RLWE) LPR scheme recap [LPR13]
- RLW**R** version of LPR scheme
- Security proof sketch
- Evaluation
- Summary

[LPR13] Lyubashevsky, V., Peikert, C., & Regev, O. (2013). On ideal lattices and learning with errors over rings. *Journal of the ACM (JACM)*, *60*(6), 1-35.

What is Ring Learning with Rounding (RLWR)?

To start: what is RLWE?

Let N be a power of 2, q a modulus, and take the ring $R_q = \mathbb{Z}_q[X]/\left(X^N + 1\right)$

a is chosen randomly from R_q

$$(a, b) = (\boxed{a}, \boxed{a}s + e) \in R_q \times R_q$$

Decision problem: are the pairs sampled from this distribution, or the uniform random distribution?
Search problem: what is the secret s?

To start: what is RLWE?

Let N be a power of 2, q a modulus, and take the ring $R_q = \mathbb{Z}_q[X]/\left(X^N + 1\right)$

a is chosen randomly from R_q

$$(a, b) = (\boxed{a}, \boxed{a}\boxed{s} + e) \in R_q \times R_q$$

secret (usually ternary)

Decision problem: are the pairs sampled from this distribution, or the uniform random distribution?

Search problem: what is the secret s?

To start: what is RLWE?

Let N be a power of 2, q a modulus, and take the ring $R_q = \mathbb{Z}_q[X]/\left(X^N + 1\right)$

a is chosen randomly from R_q

error chosen from Gaussian distribution

$$(a, b) = (a, as + e) \in R_q \times R_q$$

secret (usually ternary)

Decision problem: are the pairs sampled from this distribution, or the uniform random distribution?

Search problem: what is the secret s?

How is RLWR different?

- No Gaussian errors
- Deterministic variant of RLWE
- Achieve this through rounding operation on $a.\, s$

$$(a, b) = (\boxed{a}, \boxed{a}\boxed{s} + \boxed{e}) \in R_q \times R_q$$

$$R_q = \mathbb{Z}_q[X]/(X^N + 1)$$

What is RLWR?

Let N be a power of 2, and q and p moduli

a is chosen randomly from R_q

Rounding operation:

$$\lfloor x \rceil_{q,p} := \left\lfloor \frac{p}{q} x \right\rceil$$

$$(a, b) = (\boxed{a}, \lfloor \boxed{a}\boxed{s} \rceil_{\boxed{q,p}}) \in R_q \times R_p$$

secret (usually ternary)

Decision problem: are the pairs sampled from this distribution, or the uniform random distribution?

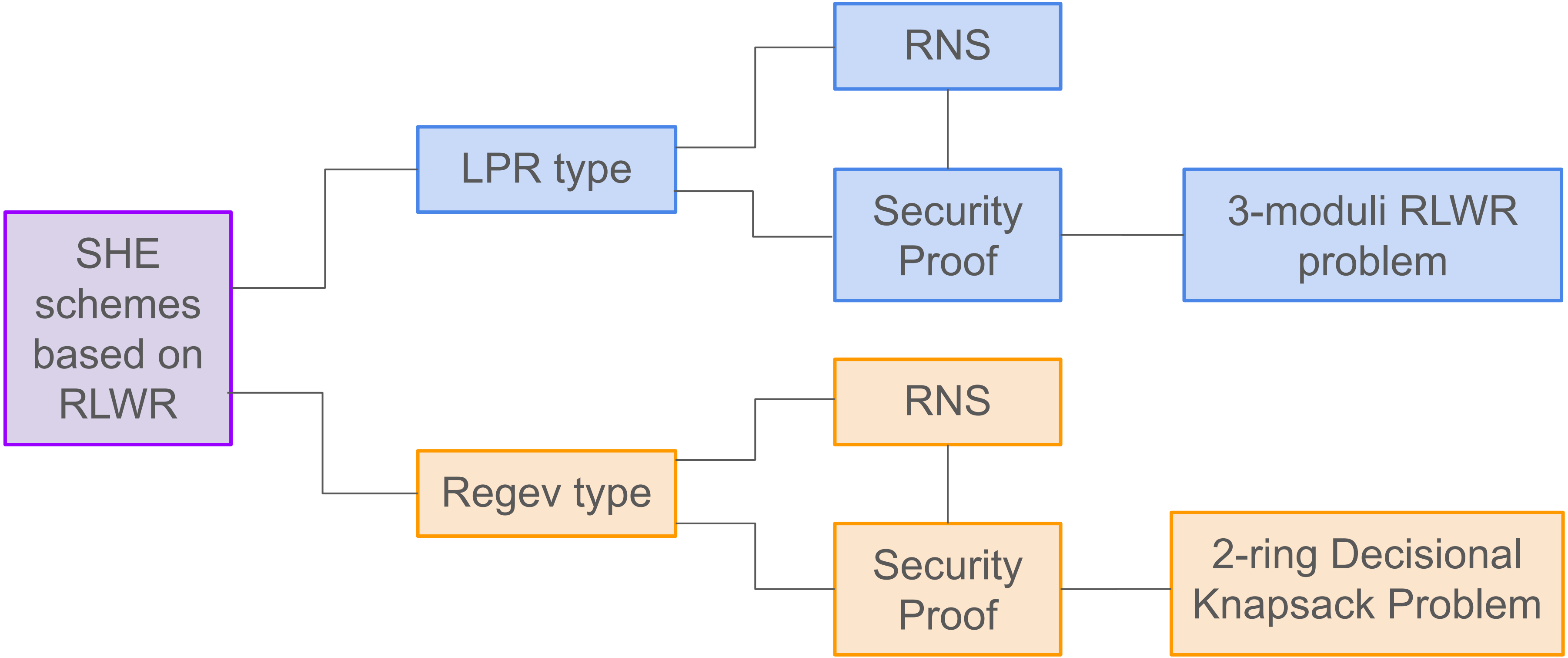
Search problem: what is the secret s?

Motivation for considering RLWR

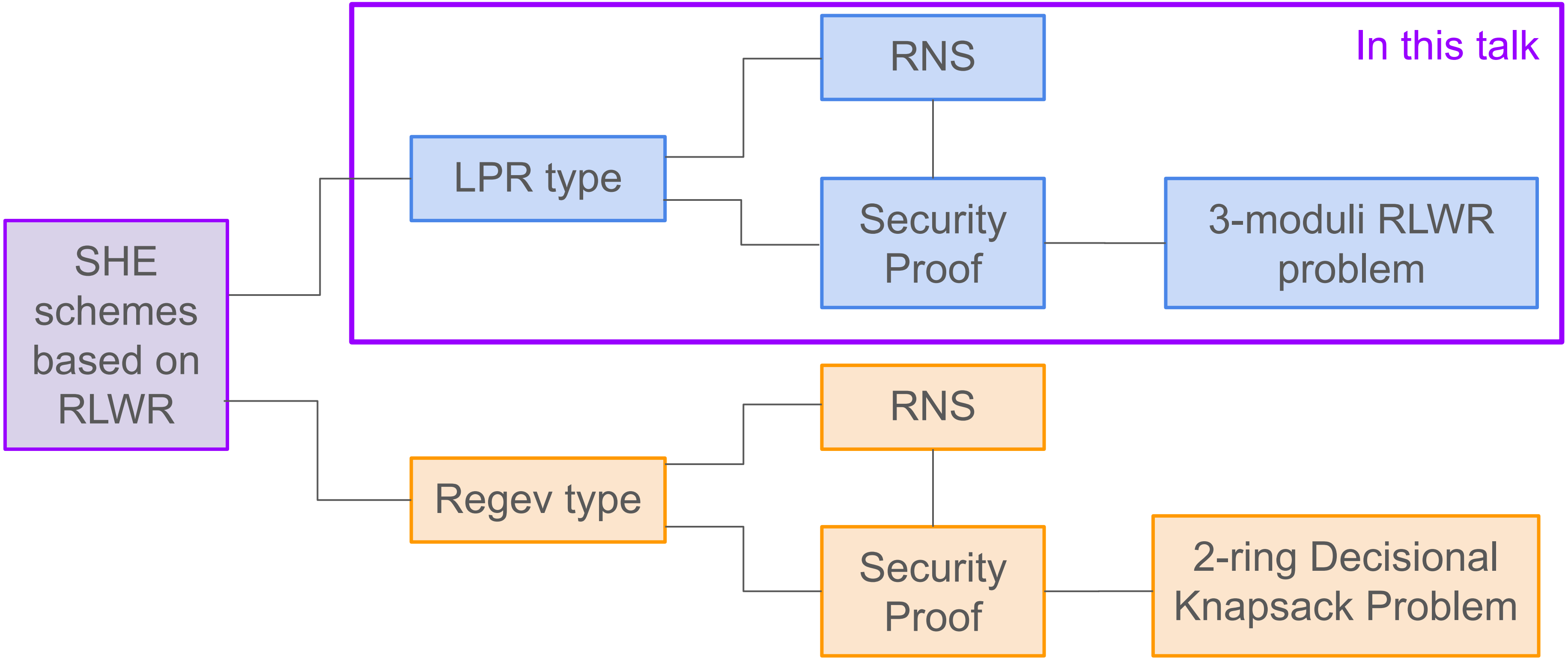
- Avoids need for Gaussian sampling
    - Side channels
    - Cost
- Easy to implement rounding
- Potential for improved bandwidth
- LWR is used by several NIST candidates (Lizard, Round5, Saber)
- Not yet been considered for many advanced primitives

Jose Maria Bermudo Mera, Angshuman Karmakar, Suparna Kundu, and In- grid Verbauwhede. Scabbard: a suite of efficient learning with rounding key-encapsulation mechanisms. IACR Trans. Cryptogr. Hardw. Embed. Syst., 2021(4):474–509, 2021

Overview of our work



Overview of our work



RLWE LPR-style BFV interlude

LPR-style encryption scheme

Key Generation:

$$\mathtt{sk} = s \in R_q$$
$$\mathtt{pk} = (a, as + e) \in R_q \times R_q$$

Encryption: $\mathtt{m} \in R_t$

$$\mathtt{ct} = (c_0, c_1) \in R_q \times R_q, \text{ where}$$

$$c_0 = \mathtt{pk}_0 u + e_0$$
$$c_1 = \mathtt{pk}_1 u + e_1 + \lfloor q/t \rfloor \cdot \mathtt{m}$$

Decryption of ct:

$$\mathtt{m}' = \left[\left\lfloor \frac{t}{q} \left[(-c_0 s) + c_1 \right]_q \right\rceil \right]_t$$

Lyubashevsky, V., Peikert, C., & Regev, O. (2013). On ideal lattices and learning with errors over rings. *Journal of the ACM (JACM)*, *60*(6), 1-35.

LPR-style encryption: multiplication

Multiplication of $ct = (c_0, c_1)$ and $ct' = (c_0', c_1')$:

$$ct_{\mathrm{mult}} = \left(\left\lfloor \frac{t}{q} c_0 c_0' \right\rceil, \left\lfloor \frac{t}{q} (c_0 c_1' + c_1 c_0') \right\rceil, \left\lfloor \frac{t}{q} c_1 c_1' \right\rceil \right) \in R_q \times R_q \times R_q$$

Lyubashevsky, V., Peikert, C., & Regev, O. (2013). On ideal lattices and learning with errors over rings. *Journal of the ACM (JACM)*, *60*(6), 1-35.

LPR-style RLWR scheme

Our LPR-style SHE scheme from RLWR

Key Generation:

$$\mathtt{sk} = s \in R_r$$
$$\mathtt{pk} = (a, b = \lfloor a \cdot s \rceil_{r,q}) \in R_r \times R_q$$

Choose:
$pr = q^2$

Encryption:

$$\mathtt{ct} = (c_0, c_1) \in R_q \times R_p, \text{ where}$$
$$c_0 = \lfloor \mathtt{pk}_0 u \rceil_{r,q}$$
$$c_1 = \lfloor \mathtt{pk}_1 u \rceil_{q,p} + \lfloor p/t \rfloor \cdot \mathtt{m}$$

Decryption:

$$\mathtt{m}' = \left[\left\lfloor \frac{t}{p} \left(-\frac{p}{q} c_0 s + c_1 \right) \right\rceil \right]_t$$

Our LPR scheme multiplication

LPR-style encryption: multiplication

Multiplication of $ct = (c_0, c_1)$ and $ct' = (c_0', c_1')$:

$$ct_{\text{mult}} = \left(\left\lfloor \frac{t}{q} c_0 c_0' \right\rceil, \left\lfloor \frac{t}{q} (c_0 c_1' + c_1 c_0') \right\rceil, \left\lfloor \frac{t}{q} c_1 c_1' \right\rceil \right) \in R_q \times R_q \times R_q$$

How we multiply together two ciphertexts in the RLWR setting:

$$\mathsf{ct}_{\mathsf{mult}} := \left(\left[\left\lfloor \frac{t}{p} c_0 c_0' \right\rceil \right]_{q^2/p}, \left[\left\lfloor \frac{t}{p} (c_0 c_1' + c_1 c_0') \right\rceil \right]_q, \left[\left\lfloor \frac{t}{p} c_1 c_1' \right\rceil \right]_p \right)$$

Security proof?

Security proof for regular LPR (from RLWE)

Hyb1 Honestly generated $\texttt{pk}$ and $\texttt{ct}$ (Real IND-CPA game)

Lyubashevsky, V., Peikert, C., & Regev, O. (2013). On ideal lattices and learning with errors over rings. *Journal of the ACM (JACM)*, *60*(6), 1-35.

Security proof for regular LPR (from RLWE)

Hyb1 Honestly generated $\mathtt{pk}$ and $\mathtt{ct}$ (Real IND-CPA game)

$\downarrow$ Decision Ring-LWE

Hyb2 Uniform $\mathtt{pk} = (a, b)$, honestly generated $\mathtt{ct} = (au + e_1, bu + e_2)$

Lyubashevsky, V., Peikert, C., & Regev, O. (2013). On ideal lattices and learning with errors over rings. *Journal of the ACM (JACM)*, *60*(6), 1-35.

Security proof for regular LPR (from RLWE)

Hyb1 Honestly generated $\mathtt{pk}$ and $\mathtt{ct}$ (Real IND-CPA game)

$\downarrow$ Decision Ring-LWE

Hyb2 Uniform $\mathtt{pk} = (a, b)$, honestly generated $\mathtt{ct} = (au + e_1, bu + e_2)$

$\downarrow$ Decision Ring-LWE

Hyb3 Uniform $\mathtt{pk}$, uniform $\mathtt{ct}$ ($\mathcal{A}$ has no information)

Lyubashevsky, V., Peikert, C., & Regev, O. (2013). On ideal lattices and learning with errors over rings. *Journal of the ACM (JACM)*, *60*(6), 1-35.

Our RLWR LPR-style security proof

Hyb1 Honestly generated $\mathtt{pk} \in R_r \times R_q$, $\mathtt{ct} \in R_q \times R_p$ (IND-CPA)

Our RLWR LPR-style security proof

Hyb1 Honestly generated $\mathtt{pk} \in R_r \times R_q$, $\mathtt{ct} \in R_q \times R_p$ (IND-CPA)

$\downarrow$ Decision Ring-LWR

Hyb2 Uniform $\mathtt{pk} = (a, b)$, honestly generated $\mathtt{ct} = (\lfloor au \rceil_{r,q}, \lfloor bu \rceil_{q,p})$

Our RLWR LPR-style security proof

Hyb1 Honestly generated $\mathtt{pk} \in R_r \times R_q$, $\mathtt{ct} \in R_q \times R_p$ (IND-CPA)

$\downarrow$ Decision Ring-LWR

Hyb2 Uniform $\mathtt{pk} = (a, b)$, honestly generated $\mathtt{ct} = (\lfloor au \rceil_{r,q}, \lfloor bu \rceil_{q,p})$

$\downarrow$ 3-moduli Ring-LWR

Hyb3 Uniform $\mathtt{pk}$, uniform $\mathtt{ct}$ ($\mathcal{A}$ has no information)

Why do we need 3-moduli RLWR?

(Decisional) 3-moduli RLWR problem: Distinguish between 3-moduli RLWR,

$$(a, b, \lfloor au \rceil_{r,q}, \lfloor bu \rceil_{q,p}) \in R_r \times R_q \times R_q \times R_p$$

and uniform tuples:

$$(a, b, c, d) \in R_r \times R_q \times R_q \times R_p$$

Why do we need 3-moduli RLWR?

(Decisional) 3-moduli RLWR problem: Distinguish between 3-moduli RLWR,

$$(a, b, \lfloor au \rceil_{r,q}, \lfloor bu \rceil_{q,p}) \in R_r \times R_q \times R_q \times R_p$$

and uniform tuples:

$$(a, b, c, d) \in R_r \times R_q \times R_q \times R_p$$

Two (Decisional) RLWR problems: Distinguish between 2 RLWR instances,

$$(a, b, \lfloor au \rceil_{r,q}, \lfloor bu' \rceil_{r,q}) \in R_r \times R_r \times R_q \times R_q$$

and uniform tuples:

$$(a, b, c, d) \in R_r \times R_r \times R_q \times R_q$$

Mapping Lemma

Let α be positive integer. Then the map $\pi : R_{\alpha q} \times R_{\alpha p} \to R_q \times R_p$ given by $(x, y) \mapsto (x \mod q, y \mod p)$ maps Ring-LWR$_{n,\alpha q,\alpha p}$ samples to Ring-LWR$_{n,q,p}$ samples and the uniform distribution to the uniform distribution.

Reduction from 3-moduli RLWR to RLWR

Choose:
q|r and
$pr = q^2$

Efficient algorithm for 3-moduli RLWR problem

$\downarrow$ Theorem 1

Efficient algorithm for RLWR problem

Evaluation

Comparing to BFV asymptotically

Choose:
$pr = q^2$

The LPR modulus r has the same size as the BFV modulus q, and the Regev modulus q

	LPR-style scheme	Regev-style scheme	Prior work [LWC18]	BFV
`pk` size	$n\log(rq)$	$ln\log(pq)$	$(l+1)n\log(p)$	$2n\log(r)$
`ct` size	$n\log(pq)$	$n\log(pq)$	$(l+1)n\log(p)$	$2n\log(r)$
security	RLWR	RLWR	RLWR	RLWE

Luo, F., Wang, K., & Lin, C. (2018) Leveled hierarchical identity-based fully homomorphic encryption from learning with rounding.
Fan, J., & Vercauteren, F. (2012). Somewhat practical fully homomorphic encryption.

Comparing to BFV asymptotically

Choose:
$pr = q^2$

The LPR modulus r has the same size as the BFV modulus q, and the Regev modulus q

	LPR-style scheme	Regev-style scheme	Prior work [LWC18]	BFV
`pk` size	$n \log(rq)$	$ln \log(pq)$	$(l+1)n \log(p)$	$2n \log(r)$
`ct` size	$n \log(pq)$	$n \log(pq)$	$(l+1)n \log(p)$	$2n \log(r)$
security	RLWR	RLWR	RLWR	RLWE

Luo, F., Wang, K., & Lin, C. (2018) Leveled hierarchical identity-based fully homomorphic encryption from learning with rounding.
Fan, J., & Vercauteren, F. (2012). Somewhat practical fully homomorphic encryption.

Methodology for concrete comparison with BFV

Goal: compare ciphertext sizes between our schemes and BFV

Find parameter set with minimal ciphertext size such that we have:

1) 128 bits security
2) Supports tree-shaped arithmetic circuit with depth L

Each level is 8 additions and 1 multiplication

What is the ciphertext size?

Minimal ciphertext size in kilobytes that supports multiplicative depth L with plaintext modulus t = 3

Scheme	Depth, L														
	1	3	5	7	9	11	13	15	17	19	21	23	25	27	29
BFV	18	74	232	309	383	969	1128	1296	1448	1607	3717	4068	4390	4738	5078
LPR-like	15	68	221	298	375	949	1105	1269	1428	1596	3692	4012	4360	4673	5043
Regev-like	17	72	228	304	380	962	1121	1287	1448	1607	3692	4040	4390	4705	5043

What is the ciphertext size?

Minimal ciphertext size in kilobytes that supports multiplicative depth L with plaintext modulus $t = 2^8$

Scheme	Depth, L														
	1	3	5	7	9	11	13	15	17	19	21	23	25	27	29
BFV	24	99	309	410	1067	1278	1489	1698	3956	4421	4837	5293	5713	6165	6608
LPR-like	22	94	298	399	1046	1252	1468	1746	3929	4360	4804	5257	5673	6123	6562
Regev-like	23	97	304	407	1053	1269	1479	1687	3956	4390	4837	5257	5713	6165	6608

Summary

What did we do?

Designed two SHE schemes based on the RLWR problem, Regev-style and LPR-style

- Demonstrate that building BFV-like schemes is possible from RLWR
- Provide security proofs for both schemes
- Show RNS variant can be achieved
- Improve ciphertext sizes compared to BFV
- Give comparable parameters to BFV

Next steps…

- Library integration?
- Building other things from RLWR?
- Applications? E.g. Private Set Intersection
- [your cool idea here!]

Thank you!

erin.hales.2018@live.rhul.ac.uk
@erin__hales
Paper coming soon to an eprint near you…