When Bad News Become Good News
Towards Usable Instances of Learning with Physical Errors

Davide Bellizia\textsuperscript{1}  Clément Hoffmann\textsuperscript{1}  Dina Kamel\textsuperscript{1}  Pierrick Méaux\textsuperscript{2}  François-Xavier Standaert\textsuperscript{1}

\textsuperscript{1} UCLouvain, ICTEAM, Crypto Group, Louvain-la-Neuve, Belgium
firstname.lastname@uclouvain.be
\textsuperscript{2} Luxembourg University, SnT, Luxembourg
pierrick.meaux@uni.lu

Monday 19\textsuperscript{th} September, 2022
Learning problems

Learning problems have proven to be interesting computationally hard problems.

- LPN
- LWE
- LWR
- MLWE

⇒

- One-way function
- Secret-key encryption scheme
- Post-quantum PKE
- Identity-based encryption
- Secure MPC
- Indistinguishability obfuscation

...
Learning problems - Presentation

\( k \overset{\$}{\leftarrow} \mathbb{F}_2^n \)

\( a \overset{\$}{\leftarrow} \mathbb{F}_2^n \rightarrow \langle a, k \rangle + e \)

LPN samples
Learning problems - Implementation

\[ a \leftarrow \mathbb{F}_2^n \]

LPN samples
Learning problems - Implementation

Learning With Physical Error

Monday 19th September, 2022 4 / 19

\[ a \leftarrow F_2^n \]

\[ \langle a, k \rangle + e \]

side-channel weakness

LPN samples
Physical learning problems - Presentation

\[ a \leftarrow \mathbb{F}_2^n \]  \[ \langle a, k \rangle + e \]

LPN samples

side-channel weakness
Physical learning problems - Presentation

\[ \mathcal{F}_2^n \rightarrow \langle a, k \rangle \]

**Example:** clock or voltage manipulation

LPPN samples
Output data dependencies:

- Error probability depending on the correct output value
- Not negligible
- Reduction for LPPN

Input data dependencies:

- Computationally hard to exploit
- Can be made small by design

Error probability depending on the correct output value
Physical learning problems - Data dependencies

OUTPUT DATA DEPENDENCIES:

- Not negligible
- Reduction for LPPN

Error probability depending on the correct output value
Physical learning problems - Data dependencies

**OUTPUT DATA DEPENDENCIES:**

- Error probability depending on the correct output value
  - Not negligible
  - Reduction for LPPN

**INPUT DATA DEPENDENCIES:**

- Computationally hard to exploit
- Can be made small by design
Can this extend to LWE?

**Goal:**

- From $\mathbb{GF}_2$ to larger rings/fields
- Error distribution approximating a CBD$_2$ or $3$ (used in Kyber)
Inner product structure

\[ \langle a, k \rangle \]
Inexact computation occurs at the final adder stage.
- Inexact computation occurs at the final adder stage
Natural solution - Results

LWPE$_A$: Error distribution approximating $CBD_3$. 

Pr[e0 = 1] = 0.50
Pr[e1 = 1] = 0.37
Bad news - Mathematical data dependencies (1/2)

Toy example:

- $\langle a, k \rangle = 0$
- modulo 4
Bad news - Mathematical data dependencies (1/2)

**Toy Example:**
- $\langle a, k \rangle = 0$
- modulo 4

\[
\langle a, k \rangle = 00 \rightarrow e = 0
\quad \begin{array}{c} \overline{00} \rightarrow e = 0 \\
\overline{01} \rightarrow e = 1 \\
\overline{10} \rightarrow e = 2 \\
\overline{11} \rightarrow e = 3 \\
\end{array}
\]

Wrong support
Bad news - Mathematical data dependencies

Regular LWE

LWPE_A

Value of last 2 LSBs of \( \langle a, k \rangle \)

Value of last 2 LSBs of \( \langle a, k \rangle \)
Bad news - Mathematical data dependencies

Regular LWE

LWPE$_A$

Value of last 2 LSBs of $\langle a, k \rangle$

Value of last 2 LSBs of $\langle a, k \rangle$

$\langle a, k \rangle$

$\langle a, k \rangle$
Inexact computation occurs at the intermediate adder stage.

Inexact computation occurs on LSBs.

\[ \langle a, k \rangle \]
Inexact computation occurs at the intermediate adder stage

Inexact computation occurs on LSBs
Results

LWPEB: Error distribution

CBD2

CBD3
Good news - physical data dependencies
Good news - physical data dependencies

- Adding independent uniform diffuses mathematical data dependencies

- Coupling could still cause output dependencies

\[ \langle a, k \rangle \]
LWPE FPGA prototype (dashed lines are only for configuration and testing).

Empirical verification that **data dependencies** cannot be observed.
Conclusion - What did we gain?

Interesting potential against leakage
Conclusion - What did we gain?

Interesting potential against leakage

• **Linear** overhead in the shares number
Conclusion - What did we gain?

Interesting potential against leakage

- **Linear** overhead in the shares number
- Trivial composition (key-homomorphic)

\[
\langle a, k \rangle = \langle a, k_0 \rangle + \langle a, k_1 \rangle + \cdots + \langle a, k_{d-1} \rangle
\]
Conclusion - What did we gain?

Interesting potential against leakage

- **Linear** overhead in the shares number
- Trivial composition (key-homomorphic)
- Inherently good against glitches

\[ \langle a, k \rangle = \langle a, k_0 \rangle + \langle a, k_1 \rangle + \cdots + \langle a, k_{d-1} \rangle \]
Find an application of this **design space** (e.g. CPE encryption, signature)
Conclusion - Next steps (2/2)

THEORETICAL WORK:

- Understanding the impact of physical data dependencies
- **Reduction** towards standard learning problems
Appendix - Physical function

**Figure 6:** LPPN processor calibration: error probability (top) and control signal (bottom).