Towards Private Deep Learning-Based Side-Channel Analysis Using Homomorphic Encryption

Fabian Schmid, Shibam Mukherjee, Stjepan Picek,
Marc Stöttinger, Fabrizio De Santis, and Christian Rechberger

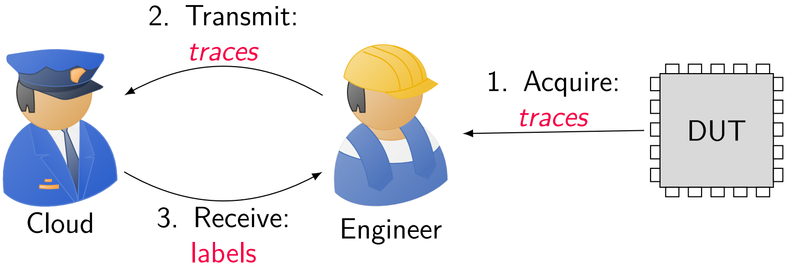
April 10, 2024

Outline

Introduction

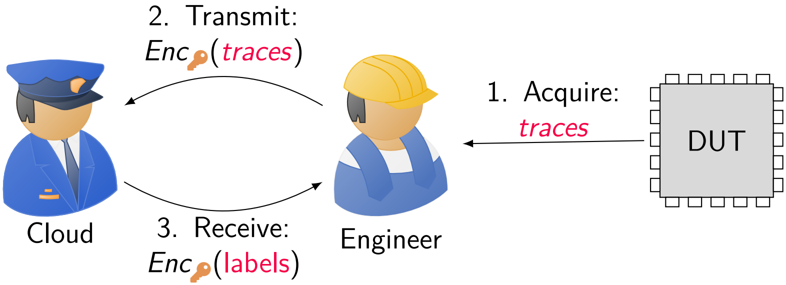
Outsourcing Side-Channel Analysis

Outsourced Side-Channel Analysis



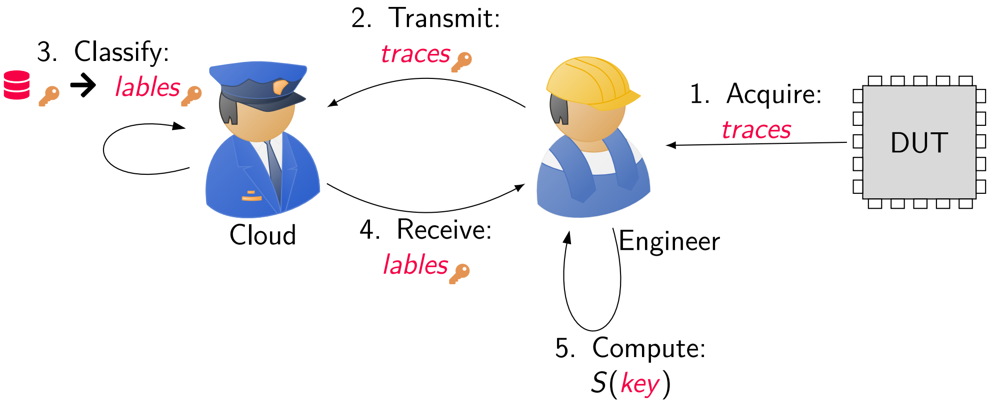
Preliminary evaluation in MLaaS setting

Private Side-Channel Analysis: Vision



Preliminary evaluation in MLaaS setting

Private Side-Channel Analysis: First Step



Outsourced Classification

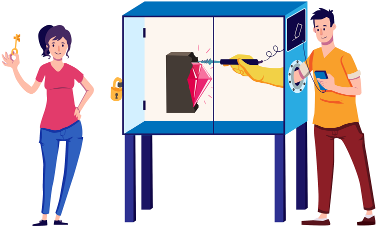
Background

Building Blocks

Homomorphic Encryption

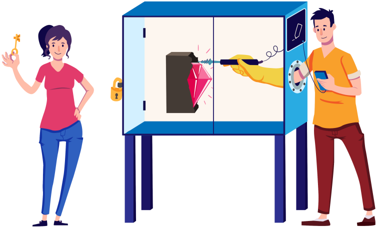
- HE scheme $\mathcal{E}$ is set of functions:
  - $\texttt{Setup}, \texttt{Enc}, \texttt{Dec}, \texttt{KeyGen}, \texttt{Eval}$

- We need addition and multiplication
- $\mathcal{E}.\texttt{Enc}$ introduces noise, increased by $\mathcal{E}.\texttt{Eval}$
- Bootstrapping resets noise
- $\mathcal{E}$ leads to Ciphertext Expansion
- $\texttt{Packing}$ alleviates it



Homomorphic Encryption: Trade-Offs

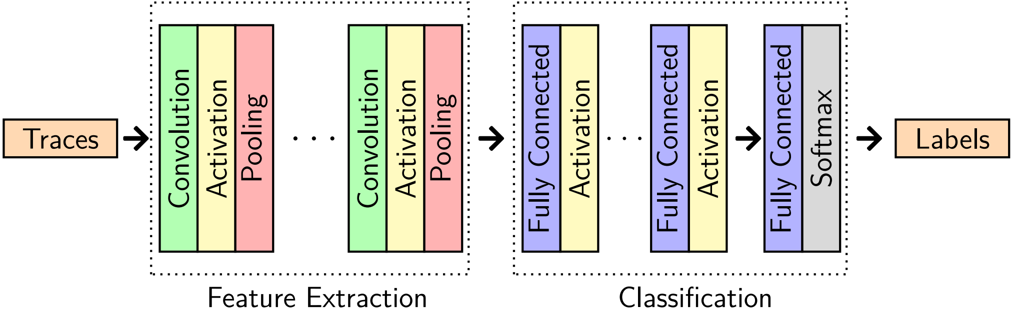
- Traits of HE schemes differ significantly:
- Binary vs. arithmetic domain
- Unlimited operations vs. high throughput
- Arbitrary operations vs. `Packed` encryption



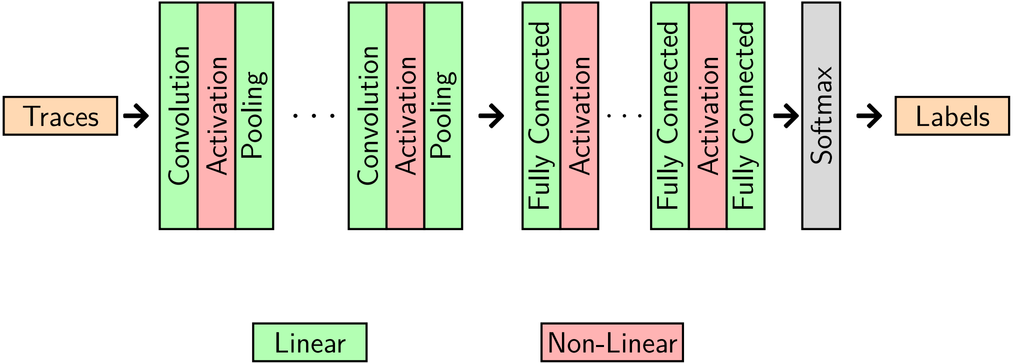
Homomorphic Encryption: CKKS Encryption Scheme [1]

- Plaintext space is polynomial ring $R = \mathbb{Z}[X]/(X^n + 1)$
- Arithmetic scheme that encodes $\mathbb{C}^{n/2} \mapsto R$
- Limited number of SIMD addition, multiplication and vector rotation
- Level parameter L limits multiplications
- Increasing L impacts performance
- High throughput scheme, for limited depth arithmetic circuits over real numbers

Convolutional Neural Network: Anatomy



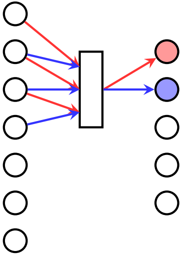
Convolutional Neural Network: Encrypted Evaluation



Method

Encrypted Power Trace Classification

Linear Layers: Convolution and Pooling



$\rightarrow$

$y_i = \sum_{j=i}^{i+f} x_j \cdot k_i$

$y'_i = F_{pool}(x'_i \ldots x'_{i+f})$

$\rightarrow$

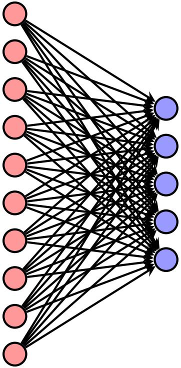
$y = \sum_{j=0}^{f} \mathtt{rot}_j(x) \cdot k$

$k = 0 \mapsto \mathtt{AvgPool}$

Convolution or Pooling

Encrypted Evaluation

Linear Layers: Fully Connected



Dense Layer

$$[x_0, \ldots, x_n] \times \begin{bmatrix} w_{0,0} & \cdots & w_{0,m} \\ \vdots & \ddots & \vdots \\ w_{n,0} & \cdots & w_{n,m} \end{bmatrix}$$

$$= [y_0, \ldots, y_m]$$

Vector Matrix Product (BSGS [2])

Non-Linear Layers: Activation Functions

- Inherently non-linear functions
- Prominent example: Scaled Exponential Linear Unit $\mathtt{SELU}(x)$:
  - If $x \leq 0$:
  $$\mathtt{SELU}(x) = \lambda\alpha \cdot (\exp(x) - 1)$$
  - Else:
  $$\mathtt{SELU} = \lambda \cdot x$$
- Low-degree polynomials via Chebyshev approximation

Evaluation

Charts and Figures

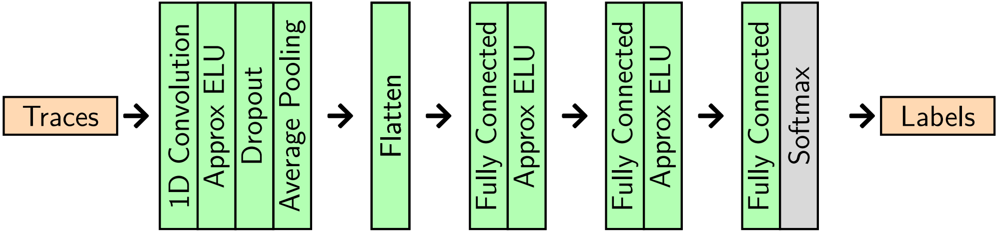
Neural Network Implementation on unprotected AES

- ChipWhisperer Dataset [3]
- Convolutional Neural Network:
  - 1 Convolution Block
  - Average Pooling
  - 2 Fully Connected Layers
  - All settings converge within 10 classifications

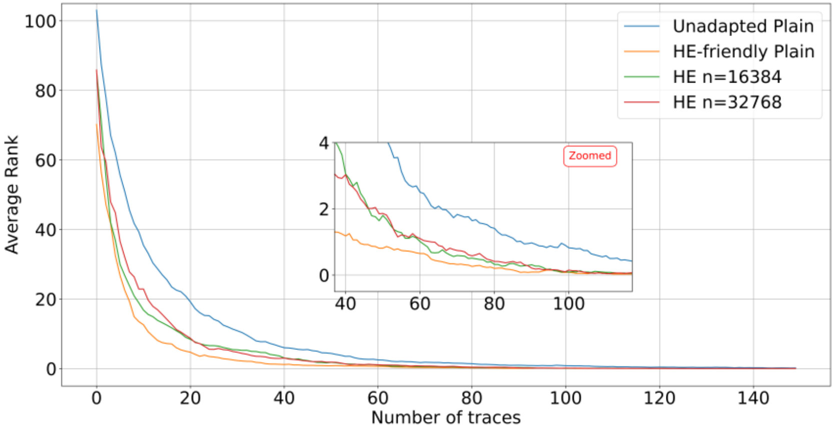
PoIs	Accuracy	Query Time [s]	Overhead
20	0.5	0.141	$14 \cdot 10^4$
500	0.8	1	$53 \cdot 10^4$
2500	1.0	4.7	$4.7 \cdot 10^4$
5000	0.2	25	$12.5 \cdot 10^4$

ASCAD Database

- 8-bit masked AES implementation (ATmega8515) [4]
- We start from optimized network architectures [5]
- Adapt for HE-friendly model



ASCAD Results



Key recovery with different network architectures and security parameters

ASCAD Results

Model	Security Level	Query Time	Overhead
Reference [5]	None	0.03 ms	-
Our Model	None	0.03 ms	-
	128 bit	13.3 s	$4.5 \cdot 10^5$
	256 bit	27.4 s	$9.4 \cdot 10^5$

After 75 traces and $\approx$ 17 minutes, the correct key is in the first two ranks.

Dataset	Security Level	n	$\log_2 q$	L
CW dataset	128 bit	16 384	360	6
	256 bit	32 768	360	6
ASCAD dataset	128 bit	16 384	430	11
	256 bit	32 768	450	11

CKKS parameters for the CW and ASCAD CNN implementations

Conclusion

Conclusion

- CNN architectures for SCA can be evaluated securely
- Our results are competitive in accuracy and trade runtime for privacy
- Secure computation may allow to outsource security evaluation
- Hardware Accelerators promise substantial improvements

However:

- Profiling requires additional techniques (MPC, Bootstrapping)
- HE-friendly circuits: Impact on other SCA techniques?

Questions

?

Bibliography I

[1] Cheon, J.H., Kim, A., Kim, M., Song, Y.S.: Homomorphic encryption for arithmetic of approximate numbers. In: Takagi, T., Peyrin, T. (eds.) Advances in Cryptology - ASIACRYPT 2017 - 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part I. Lecture Notes in Computer Science, vol. 10624, pp. 409–437. Springer (2017). https://doi.org/10.1007/978-3-319-70694-8_15, https://doi.org/10.1007/978-3-319-70694-8_15

[2] Halevi, S., Shoup, V.: Faster homomorphic linear transformations in helib. In: Shacham, H., Boldyreva, A. (eds.) Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part I. Lecture Notes in Computer Science, vol. 10991, pp. 93–120. Springer (2018). https://doi.org/10.1007/978-3-319-96884-1_4, https://doi.org/10.1007/978-3-319-96884-1_4

[3] O'Flynn, C., Chen, Z.D.: Chipwhisperer: An open-source platform for hardware embedded security research. Cryptology ePrint Archive, Paper 2014/204 (2014), https://eprint.iacr.org/2014/204, https://eprint.iacr.org/2014/204

[4] Prouff, E., Strullu, R., Benadjila, R., Cagli, E., Dumas, C.: Study of deep learning techniques for side-channel analysis and introduction to ASCAD database. IACR Cryptol. ePrint Arch. p. 53 (2018), http://eprint.iacr.org/2018/053

[5] Zaid, G., Bossuet, L., Habrard, A., Venelli, A.: Methodology for efficient CNN architectures in profiling attacks. IACR Trans. Cryptogr. Hardw. Embed. Syst. **2020**(1), 1–36 (2020). https://doi.org/10.13154/TCHES.V2020.I1.1-36, https://doi.org/10.13154/tches.v2020.i1.1-36