Low Latency Evaluation of AES via (leveled) TFHE

Benqiang Wei[1,2], Xianhui Lu[1,2], Ruida Wang[1,2], ZhihaoLi[1,2] and Kunpeng Wang[1,2]

1. Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
2. School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China

Advanced Encryption Scheme (AES)

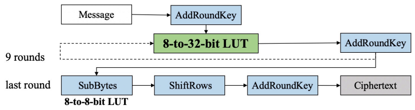
The Advanced Encryption Standard (AES) is a widely embraced block encryption standard by the United States federal government, known for its efficiency and prevalent use in securing sensitive information across diverse applications.

Motivation: AES stands as one of the top choices for application in the transciphering framework. How to achieve efficient AES evaluation has been a major challenge for the transciphering community.

Implementation methods of AES:

(1) Using four basic functions
SubBytes, RowShifts, MixColumns and AddRoundKey

(2) Using LUT-based implementation
Merge SubBytes, RowShifts and MixColumns three functions into 8-to-32-bit LUT, as follows. We present faster evaluation of AES using this implementation based on leveled TFHE.



TFHE

The TFHE scheme is based on the LWE and RLWE problems and it supports efficient gate bootstrapping, functional(programmable) bootstrapping and circuit bootstrapping.

(1) TFHE ciphertexts:
TLWE(m)、TRLWE(m(x))、TRGSW(m(x))

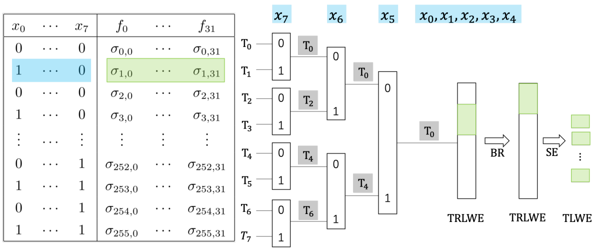
(2) Building Blocks:
- External Multiplication ⊡: TRGSW×TRLWE→TRLWE
- $\mathrm{CMUX}(c, d_1, d_0)$: $c ⊡ (d_1 - d_0) + d_0$
- KeySwitching: TLWE→T(R)LWE、TRLWE→TRLWE
- SampleExtraction(SE): TRLWE→TLWE
- BlindRotation(BR): Rotate the test polynomial blindly

(3) Bootstrapping types in TFHE:
- Identity bootstrapping
- Gate bootstrapping
- Functional (Programmable) bootstrapping
- Full domain functional bootstrapping
- Multi-value bootstrapping (PBSmanyLUT)
- Circuit bootstrapping: TLWE→TRGSW

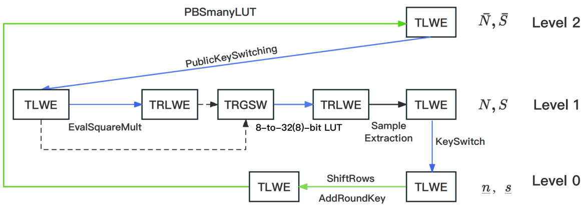
Evaluation Framework

Message Encoding: $\{0,1\} \to \{0, \frac{1}{2}\}$ over the Torus

1、Efficient 8-to-32-bit lookup table using CMUX and mixed packing



2、Efficient AES evaluation framework based on leveled TFHE



ShiftRows and AddRoundKey can be evaluated at Level 0 for free, while SubBytes can be performed in Level 1 efficiently.

Efficient Circuit Bootstrapping

New TLWE-to-TRGSW Conversion: Bridge the evaluation of AES

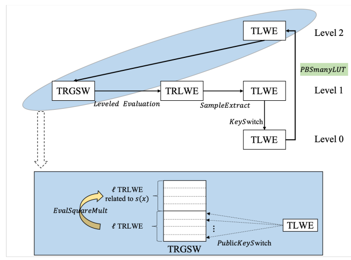
(1) **PBSmanyLUT:** $\mathrm{TLWE}(m) \to \mathrm{TLWE}\left(m \cdot \frac{1}{B^{-j}}\right), j = 1, \cdots, \ell$

(2) The second ℓ rows of TRGSW can be constructed by **PublicKeySwitch:**

$$\mathrm{TLWE}(m) \to \mathrm{TRLWE}\left(m \cdot \frac{1}{B^{-j}}\right), j = 1, \cdots, \ell$$

The first ℓ rows of TRGSW can be constructed by **EvalSquareMult:**

$$\mathrm{TRLWE}\left(m \cdot \frac{1}{B^{-j}}\right) \to \mathrm{TRLWE}\left(-s \cdot m \cdot \frac{1}{B^{-j}}\right), j = 1, \cdots, \ell$$



A test polynomial that satisfies the **negacyclic** property for PBSmanyLUT as follows:

$$P(X) = \sum_{i=0}^{\frac{N}{2^{\rho}\cdot 2}-1} \sum_{j=0}^{2^{\rho}-1} (-1) \cdot \frac{1}{2B_g^j} X^{2^{\rho}\cdot i + j} + \sum_{i=\frac{N}{2^{\rho}\cdot 2}}^{\frac{N}{2^{\rho}}-1} \sum_{j=0}^{2^{\rho}-1} \frac{1}{2B_g^j} X^{2^{\rho}\cdot i + j}, \text{ where } \rho = \lceil \log_2 \ell \rceil$$

Performance

Experimental environment

a single core of Intel(R) Core(TM) i5-11500 CPU @ 2.70GHz and 32 GB RAM, running the Ubuntu 20.04 operating system.

Implementation result

Table: Comparison of AES-128 evaluation latency based on different schemes

Scheme	Evaluation mode	Latency	Amortized
BGV	Leveled[GHS12]	4 mins	2 s
	Bootstrapping[GHS12]	18 mins	6 s
CKKS	Bootstrapping[ADE+23]	31mins	56.7 ms*
TFHE	Functional bootstrapping[SMK22]	4.2 mins	4.2 mins*
	Functional bootstrapping[TCBS23]	270 s	270 s
	Functional bootstrapping[BPR23]	211 s	211 s
	Ours(Leveled)	**46 s**	**46 s**

Our AES evaluation latency based on leveled TFHE is about 5x improvement over the state-of-the-art work in terms of latency.

References

[CGGI20] Ilaria Chillotti, Nicolas Gama, Mariya Georgieva, and Malika Izabachène.TFHE: fast fully homomorphic encryption over the torus. J. Cryptol., 33(1): 34–91, 2020.

[CLOT21] Ilaria Chillotti, Damien Ligier, Jean-Baptiste Orfila, and Samuel Tap. Improved programmable bootstrapping with larger precision and efficient arithmetic circuits for TFHE. In Advances in Cryptology-ASIACRYPT2021.

[GHS12] Craig Gentry, Shai Halevi, and Nigel P. Smart. Homomorphic evaluation of the AES circuit. In Advances in Cryptology-CRYPTO2012, Springer, 2012.

[CLT14] Jean-Sébastien Coron, Tancrède Lepoint, and Mehdi Tibouchi. Scale-invariant fully homomorphic encryption over the integers. In Public-Key Cryptography - PKC 2014, pages 311 – 328. Springer, 2014.

[GBA22] Antonio Guimarães, Edson Borin, and Diego F. Aranha. MOSFHET: optimized software for FHE over the torus. IACR Cryptol. ePrint Arch., page 515, 2022.

[SMK22] Roy Stracovsky, Rasoul Akhavan Mahdavi, and Florian Kerschbaum. Faster evaluation of aes using tfhe. Poster Session, FHE. Org-2022, 2022.

[TCBS23] Daphné Trama, Pierre-Emmanuel Clet, Aymen Boudguiga, and Renaud Sirdey. At last!A homomorphic AES evaluation in less than 30 seconds by means of TFHE.

[KLD+23] Andrey Kim, Yongwoo Lee, Maxim Deryabin, Jieun Eom, and Rakyong Choi. LFHE: fully homomorphic encryption with bootstrapping key size less than a megabyte. IACR Cryptol. ePrint Arch., page 767, 2023.

[BPR23] Nicolas Bon, David Pointcheval, and Matthieu Rivain. Optimized homomorphic evaluation of boolean functions. Cryptology ePrint Archive, Paper 2023/1589, 2023. https://eprint.iacr.org/2023/1589.

[ADE+23] Ehud Aharoni, Nir Drucker, Gilad Ezov, Eyal Kushnir, Hayim Shaul, and Omri Soceanu. E2E near-standard and practical authenticated transciphering. IACR Cryptol. ePrint Arch., page1040, 2023.

[WWL+23] Benqiang Wei, Ruida Wang, Zhihao Li, Qinju Liu, and Xianhui Lu. Fregata: Faster homomorphic evaluation of AES via TFHE. In Information Security - 26th International Conference, ISC 2023.

Contact us: {weibenqiang,luxianhui,wangruida,lizhihao,wangkunpeng}@iie.ac.cn