ICCD 2025

SecNPU: Securing LLM inference on NPU

Xuanyao Peng[12], Yinghao Yang[1], Shangjie Pan[13], Junjie Huang[2], Yujun Liang[2], Hang Lu[13], Fengwei Zhang[2], Xiaowei Li[13]

[1]SKLP, Institute of Computing Technology, CAS

[2]Department of Computer Science and Engineering, SUSTech

[3]Zhongguancun Laboratory

Summary

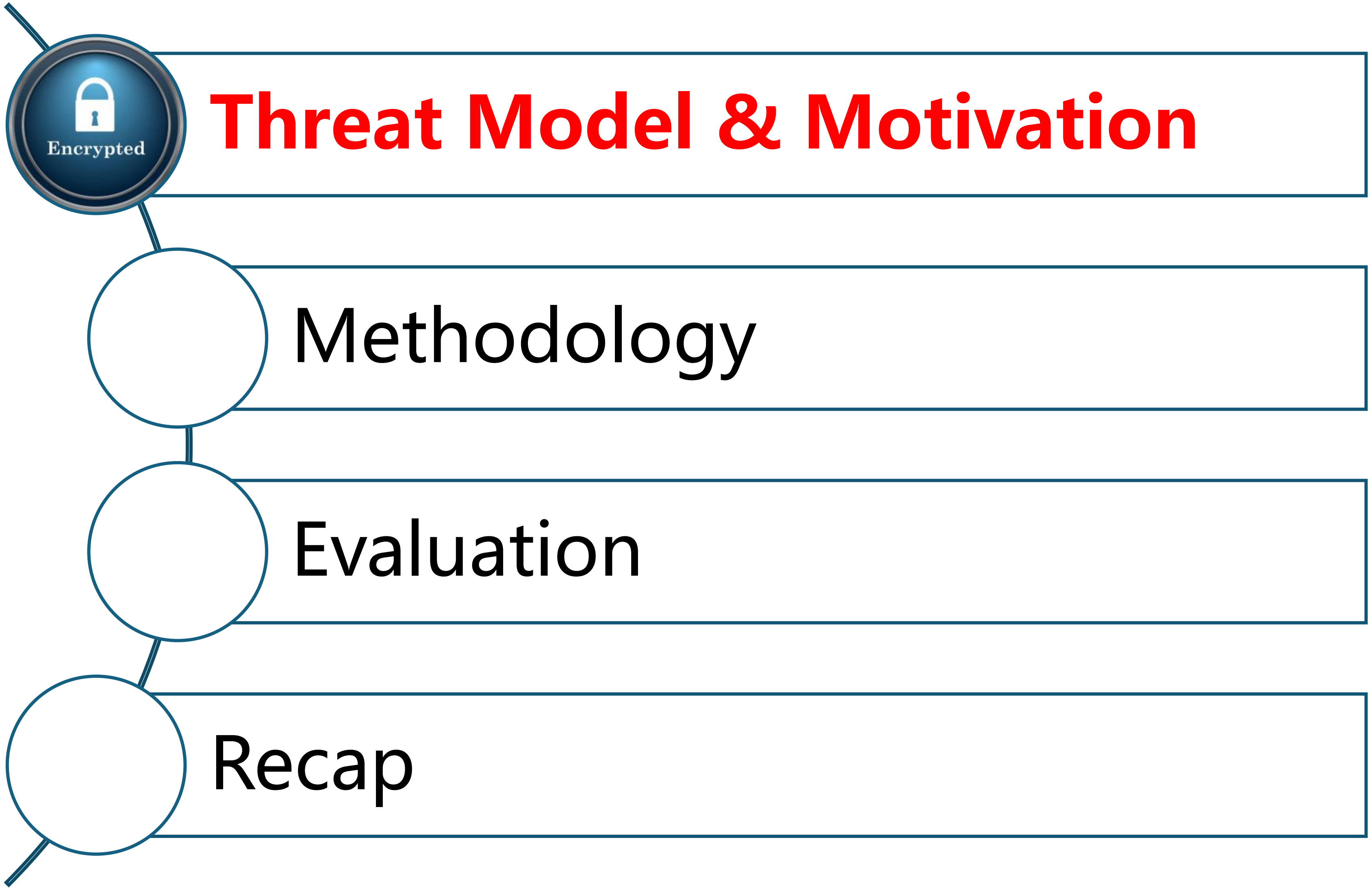
The contributions of this work:

1. Propose a CPU-decoupled TEE architecture for LLM inference – SecNPU.
2. Propose an near-zero-overhead secure startup mechanism for LLMs.
3. Implement the prototype based on RTL design and evaluate its performance using a cycle-accurate NPU simulator.

Benefits:

① Broad compatibility	Use unified security metadata and apply to various kinds of CPU.
② High performance	1.6x speedup for LLM startup and 1.5x speedup for LLM decoding.
③ Strict security guarantee	Protect from malicious OS and hardware attacks.

OUTLINE



- Threat Model & Motivation
- Methodology
- Evaluation
- Recap

Threat Model

Security Threats in CPU-NPU Heterogeneous Systems:

User's Privacy:

- Confidential user prompts
- Private user data

Model's Parameters:

- Data poisoning attacks
- Theft of model weights

Image sources: hiddenlayer.com; forbes.com

Threat Model

Security Threats in CPU-NPU Heterogeneous Systems:

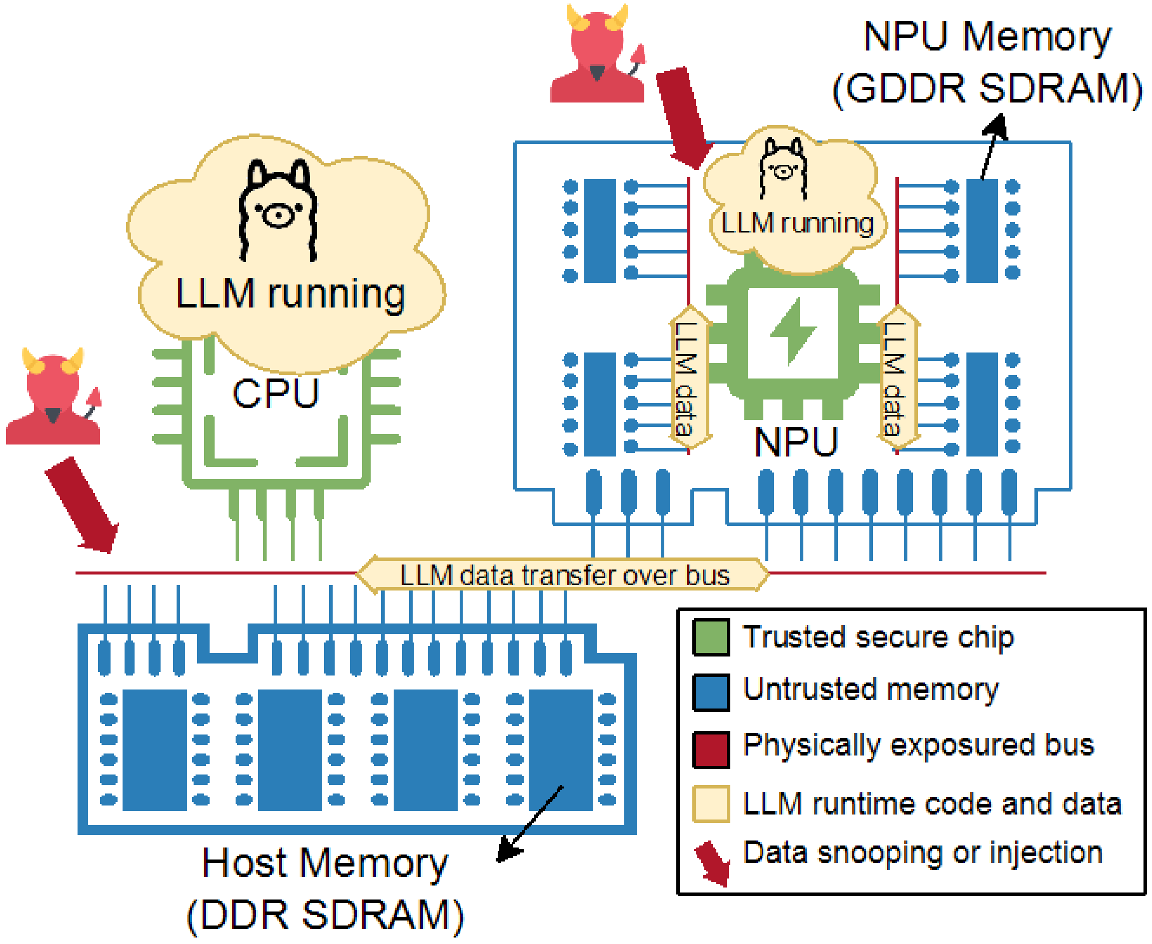
User's Privacy:

- Confidential prompts
- Private inputs

Model's Parameters:

- Data poisoning
- Steal confidential weights

Inputs and model parameters can be transmitted to the NPU by a malicious OS, exposing the data **on the physical bus**.

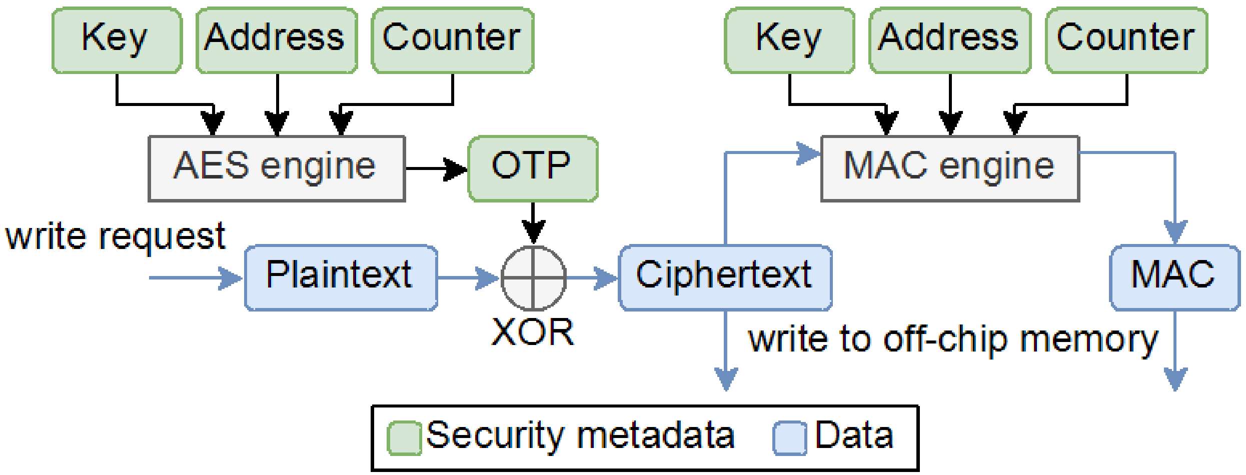


Motivation

The security mechanisms of traditional TEE (Trusted Execution Environment):

- **Encrypt plaintext using AES-GCM**
- **Protect ciphertext's integrity using MAC**

(Message Authentication Code)



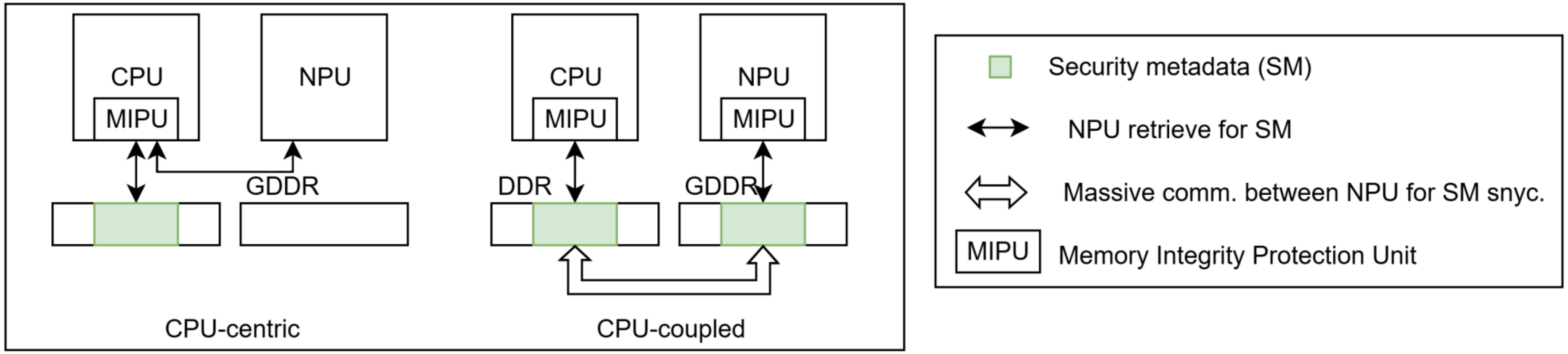
3 types of security metadata:

- **Private key stored within the Root of Trust (RoT) on chip**
- **Physical address of data**
- **Counter to ensure data freshness**

Motivation

The traditional CPU-NPU TEE can be classified in two categories:

- **CPU-centric: All security functions (AES-GCM/MAC) are handled by the CPU (e.g. TNPU HPCA'22)**
- **CPU-coupled：Part of security functions （MAC） are delegated to the CPU （e.g. TensorTEE ASPLOS'24)**

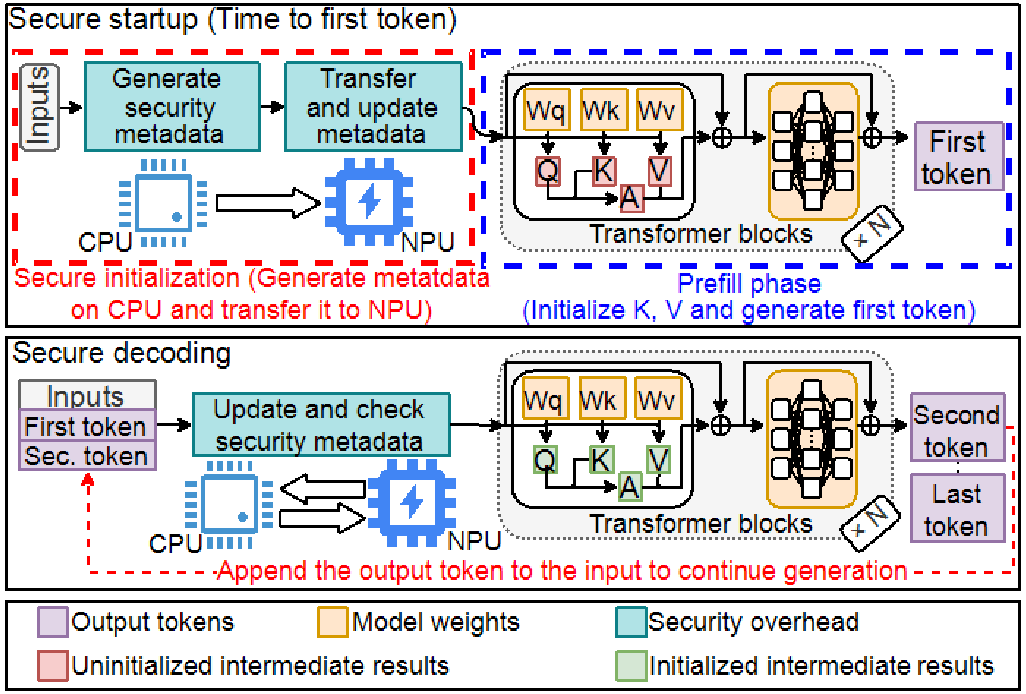


Both face:

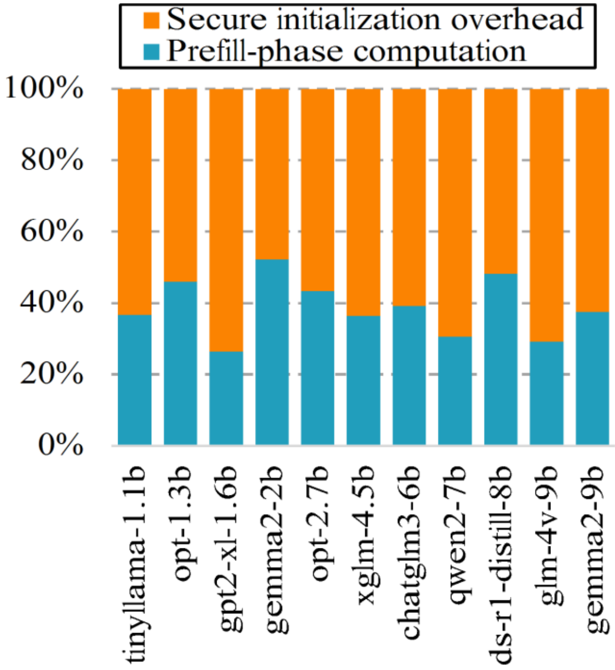
- **Slow startup due to security metadata initialization and transmission**
- **High communication overhead during LLM inference**

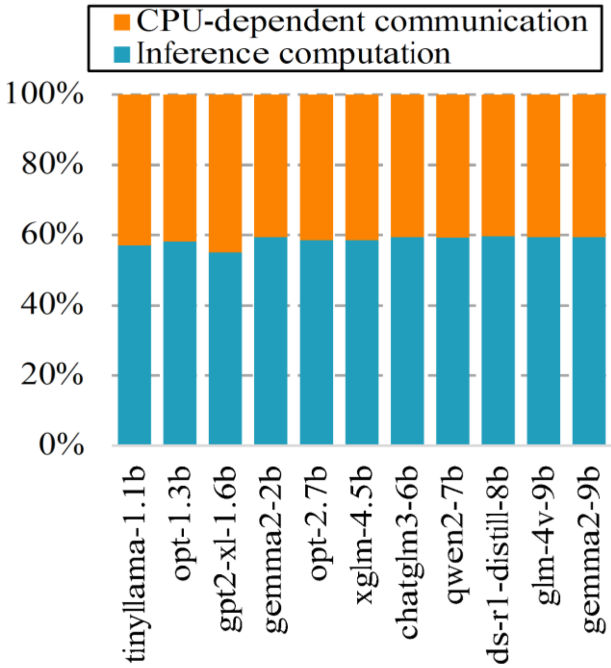
Motivation

The **security overhead** of CPU-centric/coupled TEE is significant



Secure startup (Time to first token)
Inputs
Generate security metadata
Transfer and update metadata
CPU
NPU
Wq
Wk
Wv
Q
K
V
A
Transformer blocks
× N
First token
Secure initialization (Generate metatdata on CPU and transfer it to NPU)
Prefill phase
(Initialize K, V and generate first token)
Secure decoding
Inputs
First token
Sec. token
Update and check security metadata
CPU
NPU
Wq
Wk
Wv
Q
K
V
A
Transformer blocks
× N
Second token
Last token
Append the output token to the input to continue generation
Output tokens
Model weights
Security overhead
Uninitialized intermediate results
Initialized intermediate results





The security overhead introduced:

- **Secure startup**
- **Secure decoding**

Startup overhead: 60% (average)

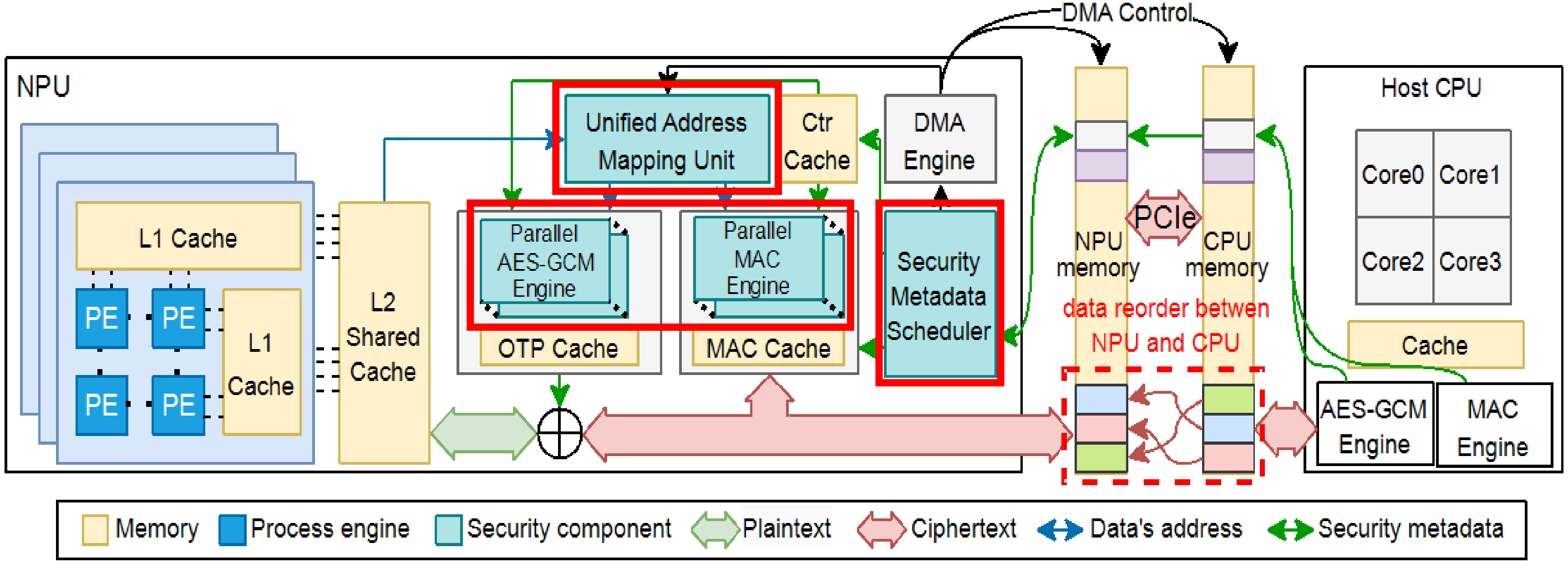
Inference overhead: 40% (average)

OUTLINE

- Threat Model & Motivation
- **Methodology**
- Evaluation
- Recap

Methodology

The overview architecture of SecNPU: CPU-decoupled TEE



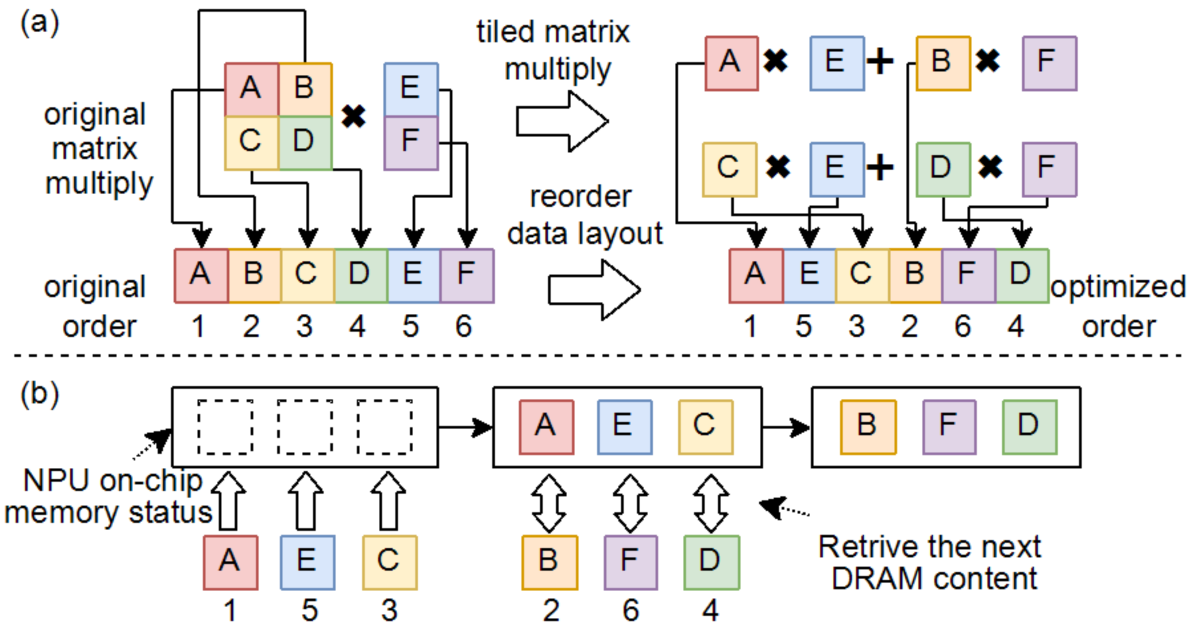
3 key security components introduced:

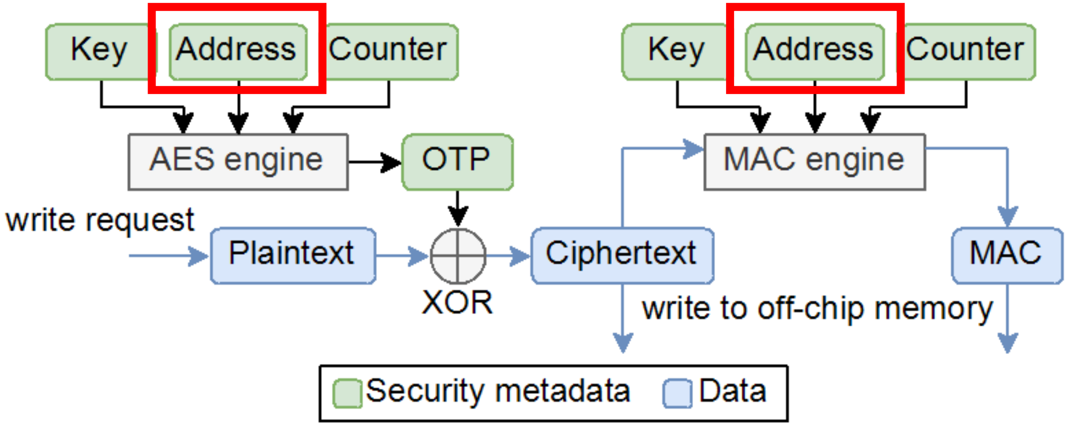
- **Unified Address Mapping Unit**: Handles data remapping after transfer from the CPU
- **Parallel AES-GCM/MAC Engine**: Accelerates NPU encryption & integrity verification
- **Security Metadata Scheduler**: Mitigates security overhead during startup

Towards Unified Security Metadata and Near-Zero-Overhead Secure Startup!

Methodology

Unified Security Metadata: Unified Physical Address





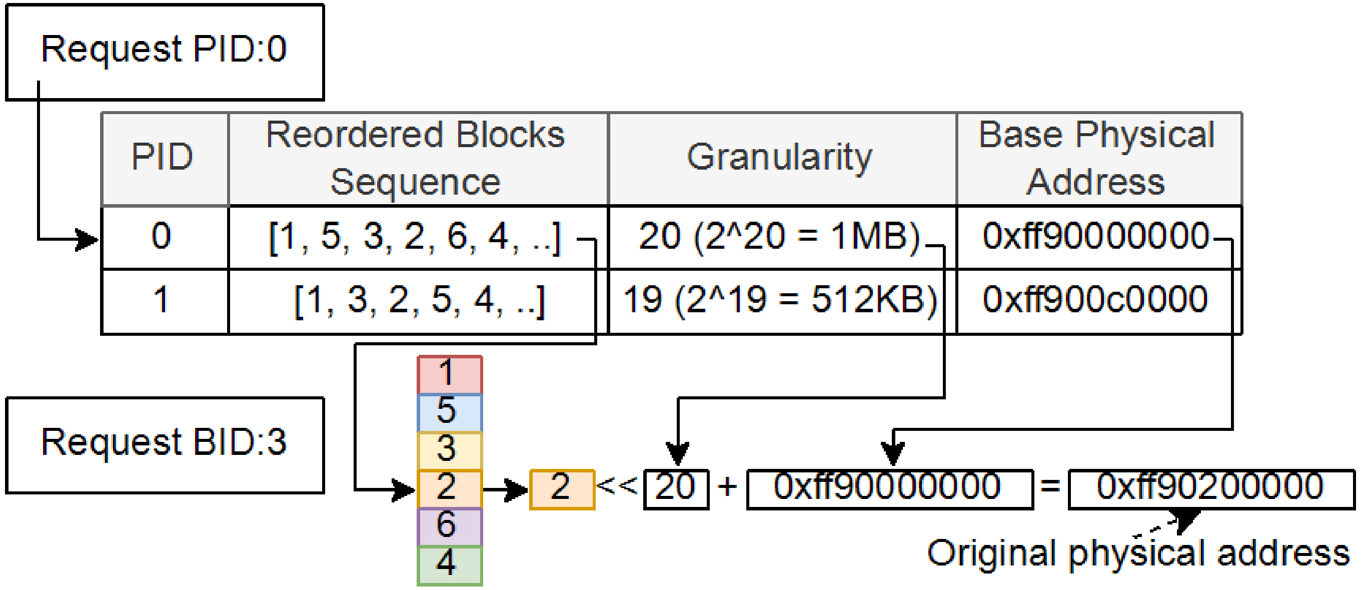
Data is reordered on NPU-side to improve prefetch accuracy.

Security metadata generated for old addresses becomes invalid!

How to maintain security metadata?

Methodology

Unified Security Metadata: Unified Physical Address



PID	Reordered Blocks Sequence	Granularity	Base Physical Address
0	[1, 5, 3, 2, 6, 4, ..]	20 (2^20 = 1MB)	0xff90000000
1	[1, 3, 2, 5, 4, ..]	19 (2^19 = 512KB)	0xff900c0000

Design a dynamic mapping table:

- **Original order**
- **Reorder granularity**
- **Original base physical address**

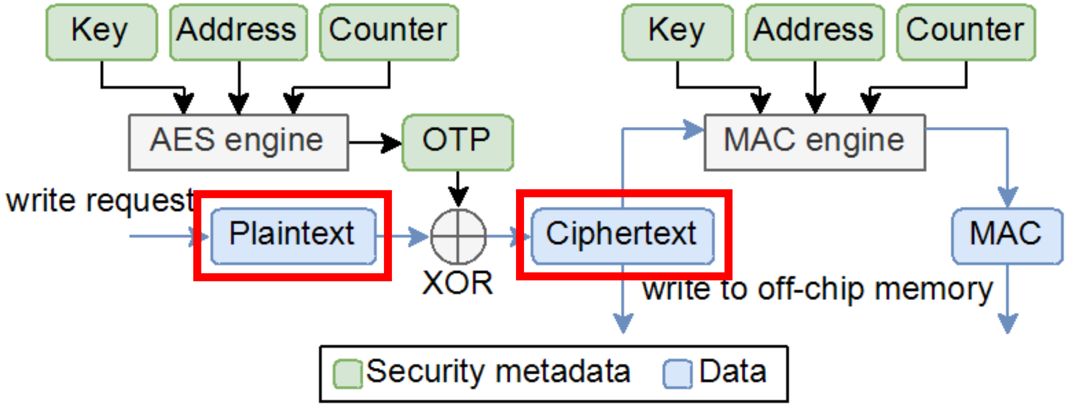
Original physical address

Methodology

Unified Security Metadata: Unifying Memory Protection Granularity

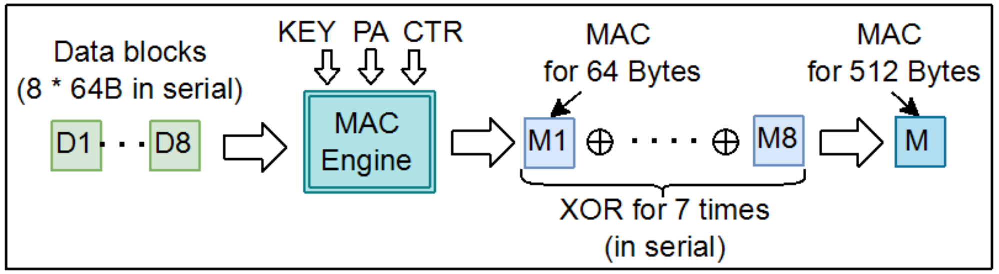
Access Granularity Mismatch:

- **CPU: 64-byte**
- **NPU: Larger blocks with DMA (Vendor-dependent)**

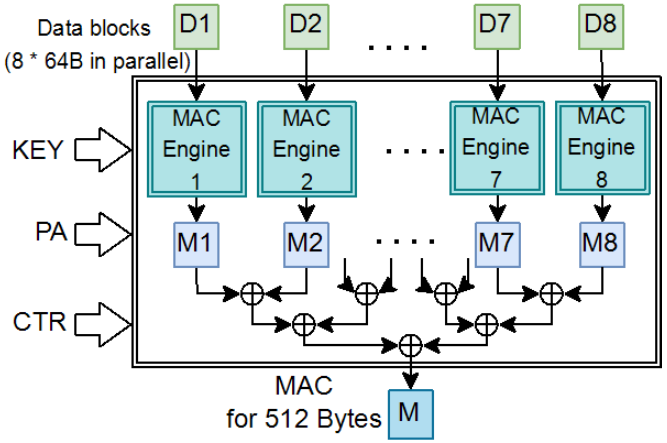


The CPU and NPU use different plaintext/ciphertext block sizes.

CPU:



NPU:



Unified to NPU-side granularity

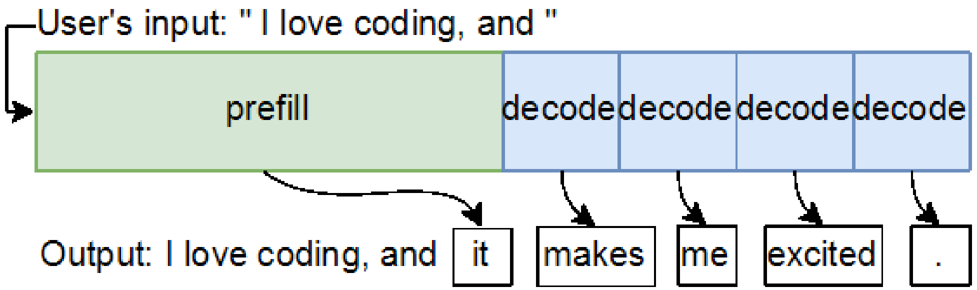
(CPU performs software-based metadata alignment to match NPU)

Methodology

Near-Zero-Overhead Startup: LLM-Oriented Optimization

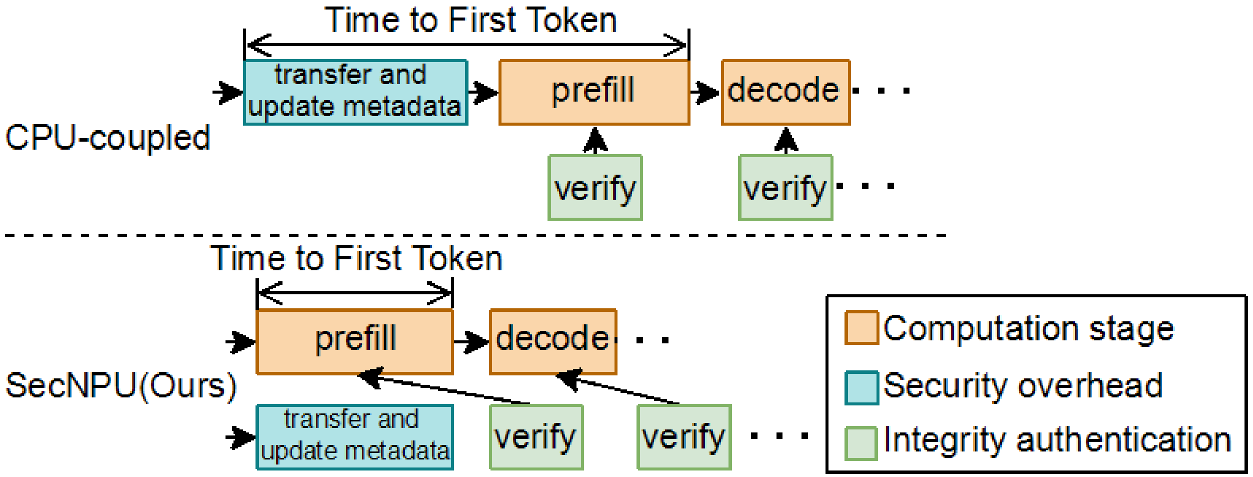
The two stages of LLM inference

- **Prefill: Compute-intensive stage**
- **Decode: Memory-intensive stage**


User's input: " I love coding, and "
prefill
decode
decode
decode
decode
Output: I love coding, and
it
makes
me
excited
.

This results in the prefill phase occupying significantly less memory bandwidth than the decode phase.

Transfer security metadata during the prefill stage to eliminate overhead!



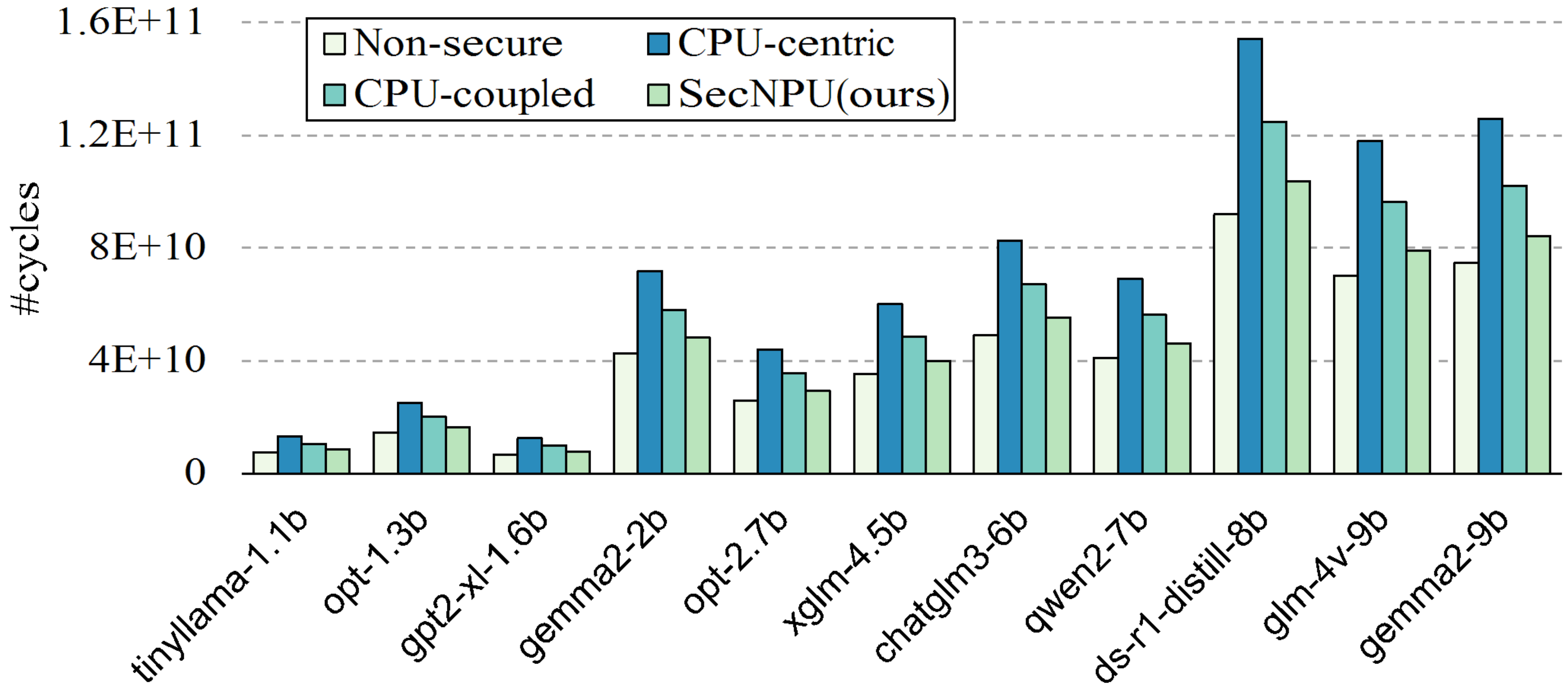
OUTLINE

- Threat Model & Motivation
- Methodology
- **Evaluation**
- Recap

Evaluation

Overall Performance

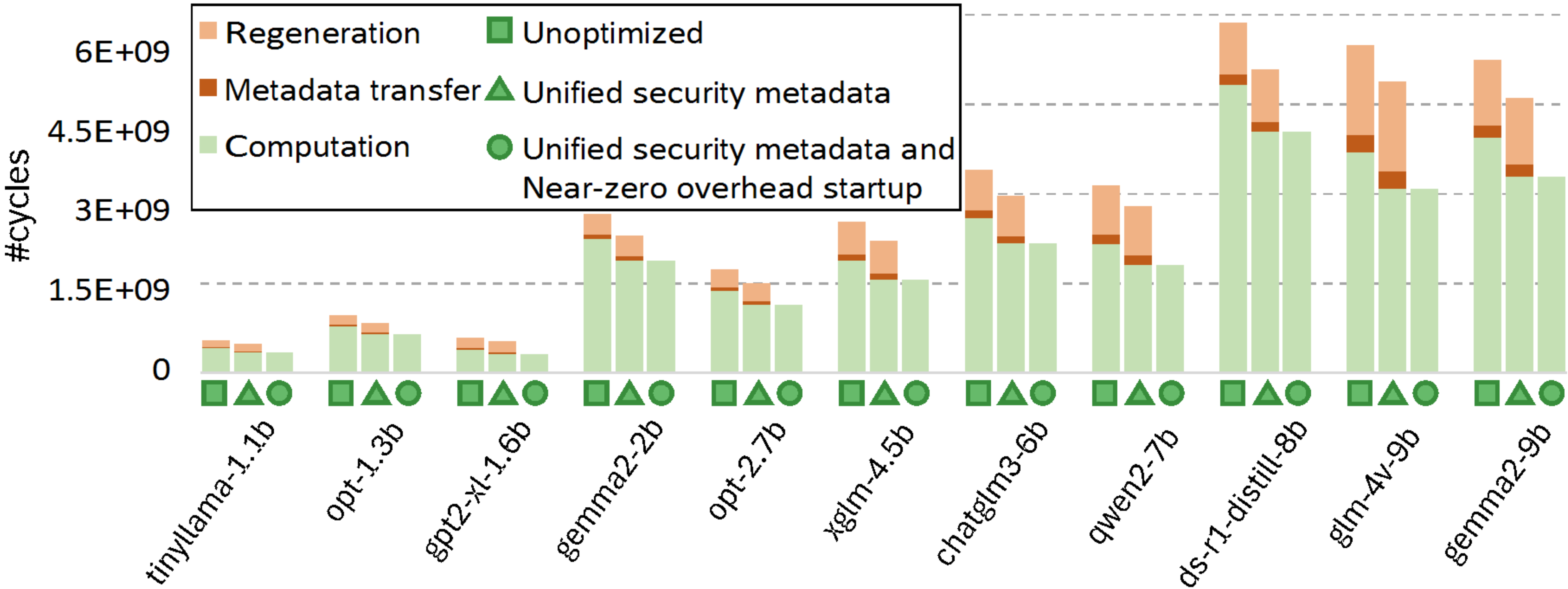
CPU-centric：TNPU
CPU-coupled：TensorTEE



The index measures the total cycles required; a lower value is better.

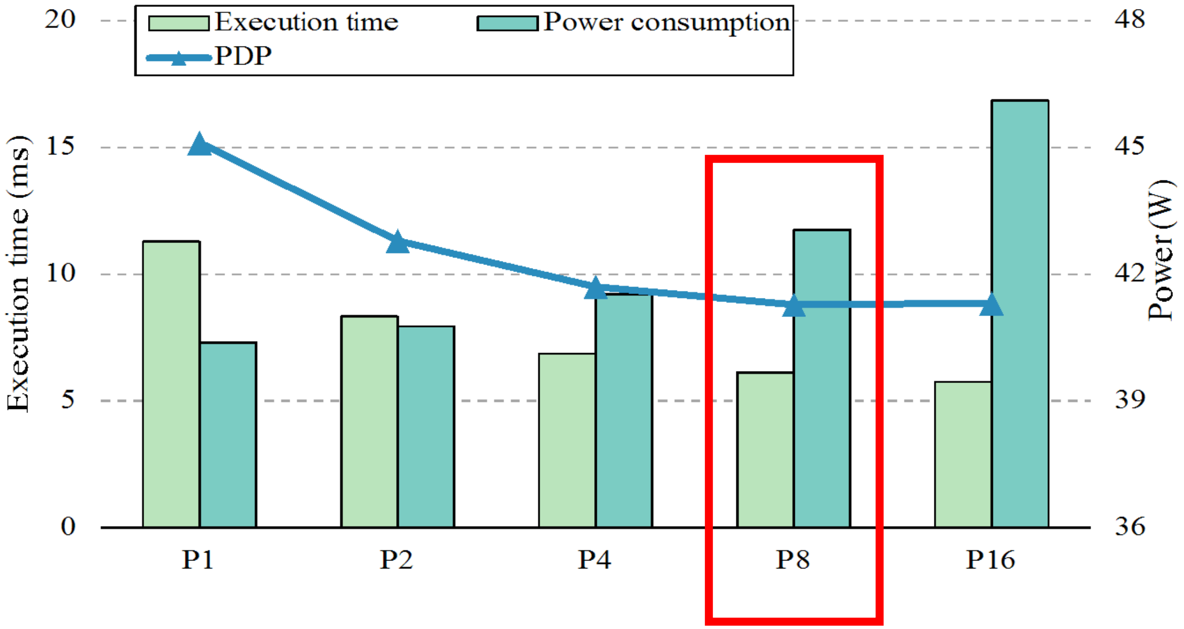
Evaluation

Ablation Study

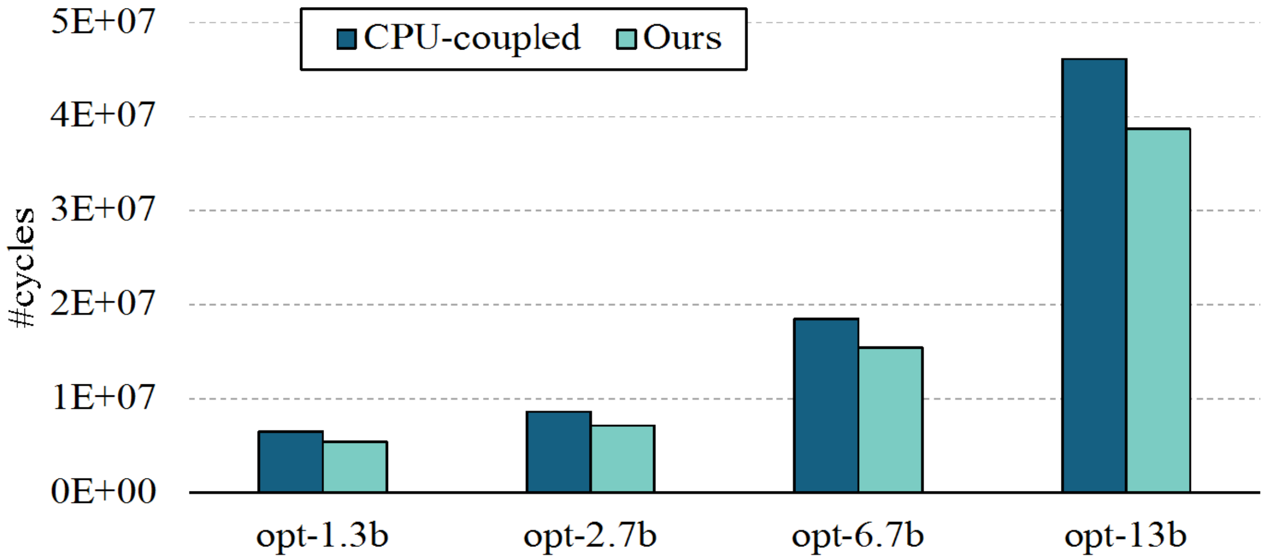


Evaluation

Design Space Exploration



Sensitivity Analysis of Multi-size LLMs



Recap

The contributions of this work:

1. Our work, SecNPU, proposes a CPU-decoupled TEE architecture:
   - unified security metadata
   - near-zero-overhead startup
2. Our prototype demonstrates speedups of up to 1.6x, all while providing robust security guarantees against both OS and hardware attacks

Thanks for listening!

For any further questions, please feel free to contact:
Xuanyao Peng
Email: pengxuanyao23s@ict.ac.cn

SKLP
处理器芯片全国重点实验室
State Key Lab of Processors , ICT , CAS

ICT 中国科学院计算技术研究所
INSTITUTE OF COMPUTING TECHNOLOGY , CHINESE ACADEMY OF SCIENCES

COMPASS Lab
COMPuter And System Security Lab

SUSTech Southern University of Science and Technology

中关村实验室 ZGC LAB