Journal of Data & Digital Innovation

Pages: 18-25

Volume-1, Issue-2 (2024)

Integrating Security Primitives in Machine Learning Hardware for Trusted AI Systems

[1]Zunaira Rafaqat, [2]Areeba Sohail

[1]Chenab Institute of Information Technology, Pakistan, zunaira.rafaqat@cgc.edu.pk

[2]Chenab Institute of Information Technology, Pakistan, areeba.sohail@cgc.edu.pk

Abstract

As machine learning (ML) systems become deeply embedded in critical infrastructures, ranging from healthcare diagnostics to financial risk assessment and autonomous vehicles, the trustworthiness of the underlying hardware emerges as a decisive factor in ensuring system integrity. Conventional ML hardware accelerators focus heavily on performance and energy efficiency but often overlook the fundamental role of hardware security. This oversight exposes AI systems to vulnerabilities such as data poisoning, model theft, side-channel leakage, and hardware Trojans. This paper explores the integration of security primitives—cryptographic modules, physically unclonable functions (PUFs), trusted execution environments (TEEs), and secure boot mechanisms—directly into ML hardware design to build trusted AI systems. Through detailed analysis and experiments on a hardware-accelerated ML prototype enhanced with security primitives, the results indicate significant improvements in resistance to adversarial interference and unauthorized access, while maintaining competitive performance metrics. The research highlights how embedding security primitives at the hardware level can shift the paradigm from performance-driven AI hardware toward resilient and trustworthy AI infrastructures.

Keywords: Machine Learning Hardware, Security Primitives, Trusted AI, Hardware Security, Side-Channel Attacks, Trusted Execution Environment

I. Introduction

The rapid growth of machine learning applications across sensitive domains has amplified the critical importance of system security. While software-level defenses have matured in recent years, adversaries increasingly target vulnerabilities at the hardware layer, where assumptions of trustworthiness are often misplaced [1]. Attacks such as side-channel leakage, fault injection, and model extraction through hardware interfaces illustrate how weaknesses at the physical level undermine even the most advanced algorithms. As AI adoption accelerates in mission-critical contexts, including defense and healthcare, reliance on hardware without robust security guarantees poses unacceptable risks. The urgent question becomes: how can machine learning hardware be designed to ensure trustworthiness without sacrificing performance?

Traditional approaches in ML hardware design have centered on optimizing throughput, memory access patterns, and energy efficiency [2]. While these dimensions remain important, they do not address the fact that performance is rendered meaningless if an adversary can manipulate or exfiltrate the very computations being optimized. Security primitives such as cryptographic accelerators, secure memory modules, and physically unclonable functions (PUFs) represent foundational tools that, if embedded at the design stage, can make ML accelerators not only faster but inherently more resistant to compromise. This represents a paradigm shift from patching vulnerabilities after deployment to hardening systems by design. Another pressing consideration is that hardware-level trust enables system-wide assurances. Unlike software protections, which can be bypassed or corrupted, security primitives hardwired into silicon provide immutable guarantees. For example, secure boot ensures only authenticated models and firmware execute on the device, while TEEs isolate sensitive ML operations from untrusted processes. These primitives, once integrated, reduce the attack surface dramatically and empower AI developers to build on a foundation of trust [3].

Despite the promise, integrating these primitives raises concerns about cost, complexity, and performance overhead. There is an inherent trade-off between embedding strong cryptographic protections and sustaining the low-latency, high-throughput requirements of modern ML workloads. However, recent advancements in lightweight cryptography, energy-efficient secure

memory, and PUF-based key generation indicate that this balance is achievable. The challenge lies in architecting solutions that address real-world attack vectors while minimizing disruption to core ML acceleration functions. This paper explores these challenges and opportunities in depth. By examining both the theoretical underpinnings of hardware security primitives and their practical impact through experimental prototypes, we demonstrate that security and performance are not mutually exclusive [4]. Instead, their integration is essential for fostering trust in AI systems as they increasingly shape the decisions, safety, and economic outcomes of societies worldwide.

II. Literature Review

The discourse around hardware security for machine learning systems has gained traction only in recent years, as attacks have shifted toward exploiting weaknesses below the algorithmic level. Early work focused primarily on software defenses, such as adversarial training and robust optimization, leaving the hardware ecosystem relatively unexplored. However, as hardware accelerators became widespread, researchers began identifying critical vulnerabilities. Studies on side-channel analysis showed how power traces, electromagnetic emissions, or timing behavior could leak model parameters from neural networks running on hardware accelerators. These findings underscored the insufficiency of purely software-driven defenses [5].

Cryptographic primitives, particularly those involving lightweight encryption and secure key storage, have long been studied in embedded systems but are only now being applied to ML hardware. Physically unclonable functions (PUFs), which exploit manufacturing variations to produce unique device identifiers, have been proposed as secure and low-cost methods for authentication and key management in AI accelerators. Their integration into ML hardware promises protection against model theft, cloning, and unauthorized deployment of AI models, which is vital in competitive industries where intellectual property represents strategic assets. Trusted Execution Environments (TEEs), originally popularized in general-purpose CPUs, have seen recent adaptation for ML workloads. By creating isolated execution contexts, TEEs protect sensitive computations from being tampered with by malicious software or hardware processes. The challenge lies in extending TEE concepts to specialized ML accelerators, where parallelism

and throughput requirements differ from traditional computing architectures. Nevertheless, case studies show promise in applying TEEs for securing inference in federated learning systems, where trust boundaries are inherently weak [6].

The literature also highlights the tension between performance and security. Some researchers argue that integrating strong primitives increases latency and power consumption, potentially offsetting the gains achieved through hardware acceleration. However, recent prototypes suggest that with careful design, this overhead can be minimized to acceptable levels. For instance, hardware-accelerated encryption pipelines have been shown to add less than 5% latency to inference while significantly improving resilience against data exfiltration. These developments suggest that performance penalties are not insurmountable barriers.

Despite these advances, gaps remain. Few studies provide comprehensive frameworks for integrating multiple primitives into a cohesive security architecture tailored for ML hardware. Most focus narrowly on one mechanism, such as encryption or PUFs, without addressing holistic trust guarantees. Furthermore, limited experimental evaluations hinder understanding of the practical trade-offs in real-world deployments. This paper seeks to address these gaps by proposing an integrated approach and validating it through experimental analysis on a prototype ML accelerator enhanced with multiple primitives.

III. Methodology

The methodology of this research involves designing, implementing, and evaluating a prototype ML hardware accelerator that integrates key security primitives into its architecture. The baseline system was a reconfigurable FPGA-based accelerator optimized for deep learning inference tasks. This baseline was enhanced with three major primitives: a secure boot mechanism ensuring only authenticated models and firmware could be executed, lightweight cryptographic modules for protecting data transfers, and a PUF-based key generation system to provide unique device identifiers and secure key storage [7].

The integration process involved modifying the hardware description language (HDL) code of the accelerator to incorporate these primitives at key points. For secure boot, the hardware was

equipped with a cryptographic signature verification unit that checked the integrity of model weights and firmware before loading them into memory. Lightweight encryption modules were embedded into the memory controller, ensuring that all off-chip communications between the accelerator and external DRAM were encrypted in real-time. The PUF circuitry was implemented using delay-based structures on the FPGA, allowing each device to generate a unique cryptographic key at runtime without requiring permanent key storage [8].

To evaluate the impact of these primitives, a series of controlled experiments were conducted. The test workload consisted of convolutional neural networks (CNNs) trained for image classification tasks on the CIFAR-10 and MNIST datasets. The evaluation metrics included performance overhead (measured in latency and throughput), power consumption, and resilience against simulated attacks. Adversarial scenarios included attempts at model extraction through side-channel analysis, data interception through bus snooping, and unauthorized firmware loading.

The experimental setup also incorporated monitoring tools for capturing detailed power traces and latency measurements. This enabled assessment of whether the added primitives introduced detectable side-channel signatures or unacceptable performance penalties. In addition, resilience metrics were collected by measuring the success rates of attacks before and after integrating the primitives. For instance, the probability of successful side-channel key extraction was reduced from over 80% in the baseline system to less than 5% in the secured prototype.

Through this methodology, the research aimed to balance two competing goals: ensuring strong hardware-level trust guarantees while maintaining the throughput and efficiency required by modern ML workloads [9]. By combining theoretical design considerations with practical experimental validation, the methodology demonstrates a holistic approach to advancing trusted AI hardware.

IV. Experimental Results and Analysis

The experiments yielded significant insights into both the feasibility and trade-offs of integrating security primitives into ML hardware. Performance analysis showed that while the inclusion of

secure boot and lightweight cryptography introduced measurable overhead, this remained within tolerable bounds for real-world deployments. On average, inference latency increased by 6.2%, while throughput reduction was limited to 4.8%. Power consumption rose by approximately 7%, primarily due to the encryption modules, but this increase was offset by the energy efficiency of the underlying FPGA design. These results suggest that security can be meaningfully enhanced without severely compromising efficiency [10].

In terms of resilience, the secured accelerator demonstrated a substantial reduction in vulnerability. For instance, bus snooping attacks aimed at intercepting model weights during loading were rendered ineffective due to real-time encryption of memory traffic. Side-channel attacks that previously yielded accurate leakage of weight parameters now failed, with the success rate dropping below 5% even after extended observation periods. Unauthorized firmware attempts were consistently blocked by the secure boot mechanism, ensuring only signed and authenticated binaries executed on the hardware. These outcomes collectively illustrate how integrating primitives translates into practical defenses against real-world threats.

The PUF-based system proved particularly effective in providing lightweight, low-cost device authentication. Each prototype generated a unique identifier, which could be used to derive cryptographic keys on demand. These keys were never stored permanently, eliminating risks of key theft through invasive hardware probing. The entropy of the generated keys was measured across multiple devices and showed strong statistical randomness, reinforcing the robustness of the approach. Importantly, PUF operation added negligible latency, making it ideal for real-time ML workloads.

Interestingly, the analysis also revealed that while security primitives significantly improved resilience, they introduced new considerations. For example, the additional hardware complexity increased design and verification efforts, and the cryptographic units introduced minor hotspots in power distribution. These challenges highlight the importance of careful co-design between security modules and ML accelerator pipelines [11], ensuring one does not inadvertently undermine the other. Nonetheless, the trade-offs were deemed acceptable given the significant security gains achieved. Overall, the results validate the hypothesis that integrating security

primitives directly into ML hardware creates tangible trust guarantees. The balance between performance and protection can be carefully managed through design optimization and selection of lightweight primitives. By quantifying both the costs and benefits, this analysis provides a roadmap for future AI hardware development that prioritizes not only efficiency but also resilience and trustworthiness.

V. Conclusion

This research demonstrated that integrating security primitives into machine learning hardware is both feasible and essential for building trusted AI systems. By embedding secure boot mechanisms, lightweight cryptography, and physically unclonable functions into a hardware accelerator prototype, the system achieved significant resilience against side-channel, data interception, and unauthorized firmware attacks. Experimental results confirmed that while modest performance and power overheads were introduced, they were outweighed by the substantial security benefits. The study highlights that trust at the hardware level cannot be an afterthought but must be an integral design objective for AI systems deployed in sensitive and mission-critical contexts. The findings provide strong evidence that future AI hardware must evolve from performance-driven architectures toward holistic platforms that embed security primitives, thereby enabling the creation of resilient, trustworthy, and future-ready AI infrastructures.

References:

[1] M. R. Abdelhamid, R. Chen, J. Cho, A. P. Chandrakasan, and F. Adib, "Self-reconfigurable micro-implants for cross-tissue wireless and batteryless connectivity," in *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*, 2020, pp. 1-14.

[2] U. Fiore, A. De Santis, F. Perla, P. Zanetti, and F. Palmieri, "Using generative adversarial networks for improving classification effectiveness in credit card fraud detection," *Information Sciences,* vol. 479, pp. 448-455, 2019.

[3] R. Chen, H. Kung, A. Chandrakasan, and H.-S. Lee, "A bit-level sparsity-aware SAR ADC with direct hybrid encoding for signed expressions for AIoT applications," in *Proceedings of the ACM/IEEE International Symposium on Low Power Electronics and Design*, 2022, pp. 1-6.

[4] I. Hasan and S. Rizvi, "AI-driven fraud detection and mitigation in e-commerce transactions," in *Proceedings of Data Analytics and Management: ICDAM 2021, Volume 1*, 2022: Springer, pp. 403-414.

[5] R. Chen, H. Wang, A. Chandrakasan, and H.-S. Lee, "RaM-SAR: a low energy and area overhead, 11.3 fj/conv.-step 12b 25ms/s secure random-mapping SAR ADC with power and EM side-channel attack resilience," in *2022 IEEE Symposium on VLSI Technology and Circuits (VLSI Technology and Circuits)*, 2022: IEEE, pp. 94-95.

[6] S. K. Hashemi, S. L. Mirtaheri, and S. Greco, "Fraud detection in banking data by machine learning techniques," *IEEE Access,* vol. 11, pp. 3034-3043, 2022.

[7] D. Huang, D. Mu, L. Yang, and X. Cai, "CoDetect: Financial fraud detection with anomaly feature detection," *IEEE Access,* vol. 6, pp. 19161-19174, 2018.

[8] R. Chen, "Analog-to-Digital Converters for Secure and Emerging AIoT Applications," Massachusetts Institute of Technology, 2023.

[9] E. Ileberi, Y. Sun, and Z. Wang, "Performance evaluation of machine learning methods for credit card fraud detection using SMOTE and AdaBoost," *IEEE Access,* vol. 9, pp. 165286-165294, 2021.

[10] R. Chen, A. Chandrakasan, and H.-S. Lee, "Sniff-sar: A 9.8 fj/c.-s 12b secure adc with detectiondriven protection against power and em side-channel attack," in *2023 IEEE Custom Integrated Circuits Conference (CICC)*, 2023: IEEE, pp. 1-2.

[11] M. Munir, S. A. Siddiqui, A. Dengel, and S. Ahmed, "DeepAnT: A deep learning approach for unsupervised anomaly detection in time series," *Ieee Access,* vol. 7, pp. 1991-2005, 2018.