SoK: Trusted Execution in SoC-FPGAs

Garrett Perkins*, Benjamin Macht*, Lucas Ritzdorf*, Tristan Running Crane*,
Brock LaMeres*, Clemente Izurieta*†‡, Ann Marie Reinhold*‡

*Montana State University, Bozeman, MT, USA
†Idaho National Laboratory, Idaho Falls, ID, USA ‡Pacific Northwest National Laboratory, Richland, WA, USA

***Abstract*—Trusted Execution Environments (TEEs) have emerged at the forefront of edge computing to combat the lack of trust between system components. Field Programmable Gate Arrays (FPGAs) are commonly used as edge computers but were not created with security as a primary consideration. Thus, FPGA-based edge computers are increasingly the target of cyberattacks. We analyze the existing literature to systematize the applications and features of FPGA-based TEEs. We identified 27 primary studies related to different types of System-on-Chip FPGA-based TEEs. Across a wide range of applications and features, the availability of extensible solutions is limited. Most solutions focus on specific features and applications, whereas few solutions focus on feature-rich, comprehensive TEEs that can be utilized across computer systems. Whether TEEs are specific or extensible, the paucity of published studies provides evidence of research gaps. This SoK delineates these gaps revealing opportunities for researchers and developers.**

***Index Terms*—Trusted Execution Environment (TEE), Field Programmable Gate Array (FPGA), RISC-V**

I. Introduction

In the rapidly evolving landscape of the Internet of Things (IoT) and edge computing, the demand for secure environments (SEs) has grown markedly. With the increasing interconnectivity of devices, traditional computer systems are no longer able to rely on mutual trust among components, as a compromise in one area can lead to vulnerabilities in others [1]. This heightened risk has underscored the need for SEs that can adapt to the challenges posed by the evolving domain of secure computing [2]–[4].

Most major CPU vendors have introduced their own chip-specific Trusted Execution Environments (TEE) solutions. For example, *ARM TrustZone*, *Intel SGX*, and *AMD SEV*, each provide secure computing for their respective hardware. However, these chip-specific TEEs constrain developers to a singular platform creating a unique security challenge [5]–[7].

New solutions are emerging to address this challenge by providing more modular and flexible secure environments. Consequently, significant R&D efforts are being applied to TEEs and hardware-based solutions. Among these hardware solutions are FPGAs and ASICs. This paper focuses on FPGAs, which are application-agnostic, as opposed to ASICs, which are "application-specific" by definition. FPGAs also provide expanded I/O over ASICs while allowing real-time hardware configuration to support field upgrades. FPGAs are inherently modular and used across several applications, such as radar, Unmanned Aerial Vehicles (UAVs), Industrial Control Systems (ICS), data centers, neural networks, and space avionics [8]–[10]. These applications require that FPGAs be secure.

We explore how FPGA-based TEEs are currently being used to provide secure computing environments and the specific features that make them suitable for applications in IoT and other computing domains. By highlighting gaps in existing research and solutions that improve FPGA security, our study addresses the following research questions: **RQ1:** What are the applications of FPGA-based TEEs and which features do FPGA-based TEEs employ according to the literature? **RQ2:** What gaps exist in the field of FPGA-based TEEs according to the literature?

II. Methods

We searched two databases, ACM Digital Library and IEEE Xplore, identifying 109 peer-reviewed papers using the search strings and filters shown in Table 1. After applying inclusion criteria (Table II), 27 papers remained for full evaluation.

We applied inclusion criteria focused on the convergence of TEEs, FPGAs, and cybersecurity. First, we included only papers that primarily addressed security concerns, excluding those not focused on security. Second, we considered only studies that demonstrated practical implementations or empirical evaluations, thus excluding theoretical papers and literature reviews. Furthermore, our review was limited to papers discussing System-on-Chip (SoC)-based FPGA environments, excluding those involving non-SoC processors to maintain technological specificity. Last, we prioritized open-source systems, excluding studies reliant on proprietary platforms. This prioritization ensured the studies were universally accessible and modifiable. This meticulous selection process was critical to accurately mapping the landscape of FPGA-based TEEs, identifying their applications, and detailing the specific features they employ, directly addressing our research question.

Of the 109 papers, 75 were from ACM Digital Library, and 34 were from IEEE Xplore. After applying the inclusion criteria listed in Table II, 31 papers remained: 17 from IEEE Xplore and 14 from ACM Digital Library. Despite meeting the inclusion criteria, four papers were removed from the pool of 31 due to lack of relevance, leaving 27 papers in the study. A stacked bar plot was made based on the number of papers published each year (Figure 1).

After selecting 27 papers, each was read to categorize the features and applications of these custom TEEs. Notable

TABLE I: Database search details, including search strings, filters, results.

Database	Search String	Filters	Results
ACM Digital Library	["trusted execution environment"] AND [fpga]	Past 5 years, Research Articles Only	75
IEEE Xplore	("All Metadata":"trusted execution environment") AND ("All Metadata":fpga)	2019-2024, Journals/Conferences	34

TABLE II: Inclusion criteria and number of papers excluded for each criterion. Total count of excluded papers exceeds the 109 papers obtained from the initial search strings because some papers were excluded for not meeting multiple criteria.

Criteria	Count of papers excluded
Security Focused	4
Applied Research	16
Open-source Platform	28
System on Chip Based	63

features and applications were separately categorized by paper in Table III. This table does not include the papers [6], [11], and [12], as [6] and [11] are categorized as extensible TEEs and [12] is an implementation of [6].

We built a heatmap to identify which features are most commonly associated with each application and highlight areas of researcher attention (Figure 2). This aids in visualizing the distribution of features across various applications of FPGA-based TEEs and facilitates clear and immediate understanding of the landscape of FPGA-based TEEs.

III. Results & Discussion

The increasing rate of publications around TEEs and FPGAs indicates that these are both growing areas of research in the cybersecurity community (Figure 1) [13]. Despite recent growth in publications, research on FPGA-based TEEs remains limited, with only 27 relevant studies identified.

The majority of the 27 papers focus on applications and features. More specifically, 24 papers address application-specific (15 papers) (Subsection III-A) and feature-specific (9 papers)(Subsection III-B) TEEs. Only two papers present a holistic approach to TEEs (Subsection III-C). One paper presents a use case of a holistic approach. The application-based papers focus on the topics of accelerators, cloud computing, and attack mitigation (Table III); the rest of the 24 papers are feature-driven. Root of Trust (RoT) and various memory security features are common, while features such as password recovery and upgraded page table walks are less common. Each paper presents a unique combination of applications and features (Figure 2).

A. Application Specific

Across the 27 selected papers, 15 constructed TEEs that served niche purposes. However, note that some of these "niche papers" developed TEEs that are multi-applicational (i.e., acceleration in cloud computing) but not fully extensible.

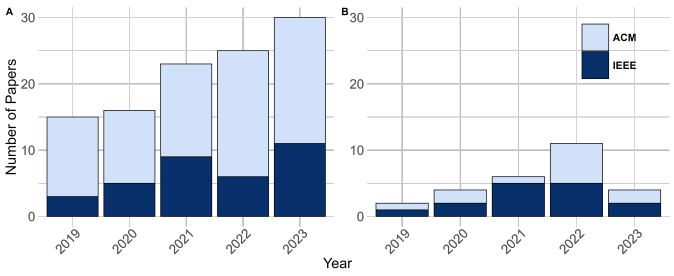

Fig. 1: Stacked bar plot representing the number of papers published over the study period. Panel A is the 109 papers found using the search strings in Table I. Panel B is the 27 papers after application of the inclusion criteria in Table II.

1) Hardware Acceleration: Seven papers discuss custom TEEs and their features as they are applied to accelerators and acceleration. TEEs emphasizing hardware acceleration primarily feature memory security, enclaves, RoT, and attestation (Figure 2). *ShEF* implements a unique Shield module for secure data access [14]. Meanwhile, *TACC* separates memory management for in-package (internal) and off-package (external) memory [15]. *AccGuard* separates and isolates memory regions for use in multi-tenant cloud environments, whereas *AccShield* supports unified virtual memory across multiple accelerators, allowing them to securely share memory resources [16]. Paper [17] used a Software-Defined Interconnect block, a hardware block that dynamically controls and sets specific boundaries for memory regions. While secure memory is the most widespread hardware acceleration feature, other features are discussed in the literature.

Papers [14], [15], and [18] differ on cloud-specific use cases, but all take an enclave-based approach to TEEs. Papers [14], [18], and [16] required attestation with a root of trust for verification purposes. Other features were less prevalent across papers focused on hardware acceleration (e.g., Secure Boot, Security Monitor [SM], Key Monitoring, Physical Unclonable Functions) but are still important for securing hardware accelerators. Developers and researchers pursue these different features to secure TEEs focused on hardware acceleration.

2) Cloud and Remote Computing: Papers on hardware acceleration almost always also focus on accelerators in a cloud computing environment (see references in Acceleration and Cloud Computing rows in Table III). Papers already discussed in Section III-A1 are re-mentioned but specific features are only discussed again here where relevant. Seven papers discuss custom-designed TEEs that implement security for cloud or remote-based FPGAs. Though applications are numerous for FPGA-based cloud computing, most papers found a need for security in a multi-tenant cloud environment. Key features such as attestation, memory security, enclaves, and RoTs are used to secure cloud environments that house accelerators [19].

Two papers discuss cloud and remote computing independent of hardware acceleration. Papers on *MeetGo* [20] and *Operon* [21] both provide TEEs for cloud and remote computing environments. *MeetGo* is a hardware-centric solution to

insider threats in cloud computing. *MeetGo* implements a TEE that operates independently of the host systems architecture, restricting the administrator's access to users' data in the cloud. *MeetGo's* modularity was demonstrated when it was implemented as a cryptocurrency wallet and General-Purpose Graphics Processing Unit [20]. *Operon* [21] aims to provide secure, encrypted database operations while maintaining compatibility with existing SQL applications. Papers [14], [16], [18], [22], [23] also are applied to cloud and remote computing, but have already been discussed in Section III-A1.

3) Attack Mitigation: Trusted Execution Environments play a critical role in attack mitigation. Almost one-fifth of the literature focuses on attack-specific mitigation through custom TEE implementation. Side channel attacks (SCAs) are a significant threat to TEEs. *ChaosINTC* [24] and *REHAD* [25] both focus on SCA mitigation, interrupt-based and cache-based, respectively. *ChaosINTC* implements a dynamic interrupt delay mechanism alongside an interrupt handler to protect their TEE [24]. *REHAD* uses reconfigurable hardware to mitigate cached SCAs [25]. While SCAs are a threat to TEEs specifically, TEEs are also used to defend against other threats.

The remaining TEEs discussed in the literature focused on preventing diverse attack vectors. *TrustToken* features isolated execution and trusted user interaction to combat software-based assaults seeking information and unauthorized access [22]. Yet another TEE seeks to combat unauthorized access, specifically through Trojans, by implementing a Hardware Trojan detection, identification, and recovery mechanism [26]. Another attack vector, fault attacks, is mitigated by *SecWalk*, which protects virtual and physical memory against fault attacks [27]. From fault attacks to information leakage, TEEs often provide a first line of defense against bad actors.

4) IP Licensing: Of the papers that do not discuss hardware accelerators, cloud computing, and attack mitigation, there are a few niche applications. Intellectual Property (IP) protection and licensing is a concern for [28] and [22] because of multi-tenant environments. These multi-tenant FPGA environments present new security risks; current solutions necessitate third-party involvement for key-programming and encryption. The aforementioned *TrustToken* [22] only permits trustworthy connections between third-party IP and the rest of the SoC, while Khan et el. [28] propose a Security framework for handling key storage and security monitoring.

5) Smart Grid Security: Smart Grid Security [29] is a niche application that implements a TEE with dual-core isolation and secure boot based on a RoT. The niche applications of IP and grid security advance the field of SoC-FPGA-based TEEs, opening the door to apply TEEs to other computing areas. Applications of TEEs are slowly expanding as demonstrated by the papers centered around hardware accelerators, cloud computing, and attack mitigation.

The application of TEEs across various domains, from hardware acceleration to cloud computing and attack mitigation, showcases their versatility and growing importance in securing modern computing environments. The innovative use of enclaves, attestation, and memory isolation in these environments highlights the challenges associated with maintaining security in dynamic, resource-shared settings. Meanwhile, the application of TEEs in attack mitigation, particularly against SCAs and hardware Trojans, underscores the necessity of security mechanisms that can preempt and neutralize threats.

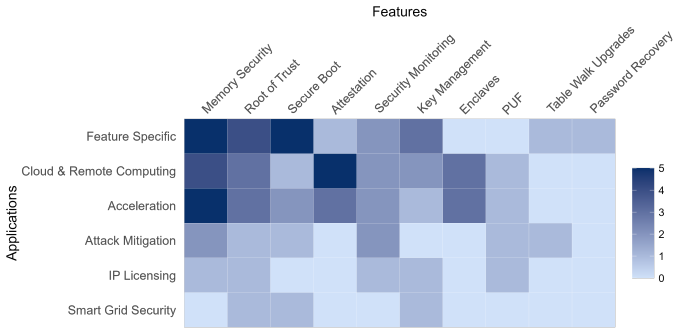

Fig. 2: Heatmap of applications and their respective features in the pool of papers. Blue hue denotes number of papers discussing the features and applications indicated on axes.

Although the focus on niche applications like IP licensing and smart grid security may seem specialized, these examples illustrate the broadening scope of TEE deployment. This trend reflects a growing recognition of the need for secure environments across all facets of computing, driving innovation and expansion in TEE capabilities.

B. Feature Specific

Nine papers focused on feature-specific TEEs. While all application-specific TEEs require a cadre of features, some researchers designed their TEEs with specific features in mind. These researchers put forth new contributions to the features TEEs can provide, however, not all features are implemented in tandem. These nine feature-specific papers focus on RoTs, attestation, memory security, secure boot, key management, and password recovery (Table III). Some papers focus on a singular feature, while others focus on multiple (Figure 2).

A RoT is a foundational element in most TEE designs, providing an anchor point for other security features, such as attestation and secure boot. Attestation ensures that the software and hardware components of a system are trustworthy. Mutual attestation allows devices of the same rank and type to verify their mutual interaction; Turan and Verbauwhede [30] use a RoT to facilitate cryptographic verification, network communication, and decision-making, thus providing mutual attestation. Paper [31] employs a RoT, implemented using a Trusted Platform Module (TPM), as the basis for a protection-dedicated core in a multi-core RISC-V system. These feature-specific implementations show how RoTs provide the foundation for critical security features.

Memory security is crucial to the isolation of a TEE. Unrestricted or compromised memory access threatens entire system security. A notable memory-focused paper, *ARES*, implements a security mechanism designed for non-volatile memory (NVM) in embedded systems [32]. [32] aims to

TABLE III: Features and applications of Open-source, SoC-Based Trusted Execution Environments

Topic	Paper
Applications	
Hardware Acceleration	[14]–[18], [22], [23]
Attack Mitigation	[22], [24]–[27]
Cloud and Remote Computing	[14], [16], [18], [20]–[23]
Feature Specific	[30]–[38]
IP Licensing	[22], [28]
Smart Grid Security	[29]
Features	
Attestation	[14], [16], [18], [20], [21], [30]
Enclaves	[14], [15], [18], [21]
Key Management	[18], [21], [28], [29], [33], [34], [37]
Memory Security	[14]–[17], [20], [22], [27], [32]–[36]
Page Table Walk Upgrades	[27], [35]
Password Recovery	[38]
Physical Unclonable Function	[22]
Root of Trust	[14], [18], [22], [29]–[31], [34], [37]
Secure Boot	[14], [15], [24], [29]–[31], [33], [34], [37]
Security Monitoring	[18], [22], [26], [28], [31], [38]

combat common memory attacks and issues with Non-Volatile Memory (NVM) by implementing a novel Bonsai Merkle Tree (BMT) scheme and leveraging parallel recovery in FPGAs. Another NVM-focused TEE, [33], proposes a methodology for securely booting from NVM in insecure environments, leveraging the reconfigurable logic of the FPGA as a secure anchor point. The Trusted Memory-Interface Unit sits in the reconfigurable logic region of the FPGA and performs integrity and authenticity verifications of NVM data prior to executing any user application, ensuring a secure boot process. The focus on NVM-based solutions highlights the importance of secure memory access in ensuring the integrity of data.

Apart from NVM, memory encryption was the focus of a single paper [36]. [36] uses a special memory encryption unit that integrates directly with RISC-V architecture to encrypt memory using the lightweight ChaCha stream cipher which encrypts and decrypts quickly using the add-rotate-XOR (ARX) structure. Paper [36] also utilizes the RISC-V Physical Memory Protection (PMP) unit to check load/store physical addresses against access restrictions. A spin-off of PMP presented by [35], Hybrid Physical Memory Protection (HPMP), blends segment-based memory protection with a permission table, combining the strengths of both approaches. This hardware-software co-design dynamically manages memory protection and allocates segments and permission tables. These memory security approaches highlight the essential role of protecting memory in ensuring TEE security and integrity.

Secure boot is a critical feature in TEEs, ensuring that the system starts in a trusted state by verifying the authenticity and integrity of the bootloader and other essential components. The aforementioned [33] securely boots from NVM where the boot image is decrypted using the dynamically generated encryption key, and its integrity is verified by comparing the calculated hash against the stored token. Uniquely, [37] focuses on mitigating the threat of quantum computers on TEEs by implementing Secure Boot. The authors implement post-quantum secure boot using the eXtended Merkle Signature Scheme (XMSS) to protect the system's boot process from quantum computing attacks that could compromise traditional asymmetric cryptographic algorithms. This establishes a secure boot chain-of-trust from the RoT up to the operating system kernel ensuring the integrity of each boot stage [37].

In conjunction with secure boot, proper key management is essential to the security of TEE environments. Paper [34] proposes a novel approach to key management within the TEE by utilizing a flexible and secure boot procedure, complete isolation from the TEE domain, and exclusive secure storage for root keys. This ensures enhanced security and flexibility in key generation and maintenance.

A few papers focus on less mainstream features such as password recovery and page table walk upgrades. [38] implement a RISC-V processor, a secure coprocessor, and a password recovery engine connected through an AXI bus. The secure coprocessor includes an instruction set architecture (ISA) monitor and secure cache for secure computing tasks, especially those involving sensitive data like passwords [38].

The diverse range of features explored across the literature highlights the components necessary for the deployment of TEEs in various computing contexts. The emphasis on foundational elements like RoTs and secure boot mechanisms underscores their role as the bedrock of secure system initialization and operation. These features establish and maintain trust, especially in environments where the integrity of both hardware and software must be assured.

Memory security, with its various implementations, is particularly crucial given the pervasive risk of unauthorized access or data breaches that could compromise the entire TEE. However, the focus on specific features like password recovery and page table walk upgrades, though less common, reflects the growing complexity and specialization of TEE functionalities as they are adapted to meet the needs of increasingly diverse and demanding applications. This progression suggests that future research will push what TEEs can achieve.

C. Extensible TEEs

The *HECTOR-V* and *Keystone* approaches provide modular and well-rounded TEEs, enabling users to plug and play rather than mix and match features and applications [6], [11].

HECTOR-V, concerns itself with side-channel attacks, arguing that, "TEEs, such as *Intel SGX* or *ARM TrustZone*, implemented on the main application processor, are insecure" [11]. Focusing on combating SCAs, these authors implement a heterogeneous multicore architecture that embeds a dedicated processor into the system to separate the secure and non-secure domains. Their RISC-V Secure Co-Processor (RVSCP) restricts I/O access and provides control-flow integrity (CFI) for secure applications. This TEE provides secure I/O using identifier-based secure communication channels between different devices in the system, which ensures that only authorized entities can access sensitive peripherals. The RVSCP processor employs hardware-enforced CFI to safeguard applications running in *HECTOR-V* using a specialized hardware unit to monitor the control flow of applications. Overall, *HECTOR-V* aims to provide a secure architecture for trusted

execution by combining a heterogeneous CPU architecture with secure coprocessor features, hardware control-flow integrity, and secure communication channels.

Lee et al. made a significant contribution to the TEE landscape when they created *Keystone*, "the first open-source framework for building customized TEEs" [6]. *Keystone* provides a comprehensive framework for implementing a modular TEE on an FPGA using RISC-V architecture. *Keystone* TEEs use enclaves and PMP to isolate different computing modes from accessing data. While memory security is critical, it is not the only feature *Keystone* TEEs provide. *Keystone* TEEs also provide a configurable security monitor (SM) that adds a trusted layer below the OS that can be configured to enforce TEE guarantees (e.g., policies and security primitives). In addition to the SM, the secure boot and attestation capabilities measure and verify the integrity of the SM and enclaves. The myriad features are accompanied by SCA mitigation as *Keystone* TEEs incorporate cache partitioning and other techniques to defend against side-channel attacks. In sum, Lee et al.'s comprehensive, open-source approach allows developers to have modularity and freedom when implementing and modifying a TEE created using the *Keystone* framework.

While both [11] and [6] present similar frameworks for TEEs, only one has been validated. The *Keystone* framework, implemented by [12], served as the architecture for a trusted IoT sensing system. The sensing system features *Keystone* and employs two types of Physically Unclonable Functions (PUFs)–one for the main device and one for the subordinate sensor. In this application, *Keystone* provides isolation from potentially untrusted operating systems and applications using its enclave system. The *Keystone* TEE integrates with a PUF, which serves as a hardware RoT that generates a unique, device-specific key for secure key management. This implementation of *Keystone* illustrates how its modularity and feature-rich build allow multi-application realization.

Keystone and *HECTOR-V* are easily adaptable to any chip using the RISC-V instruction set, though not without foibles. *Keystone*, while highly modular and customizable, heavily relies on specific RISC-V hardware features, i.e. PMP. Physical Memory Protection also limits the number of memory regions that can be protected based on PMP entries. *HECTOR-V's* multicore architecture is complex to design and implement, particularly regarding the two communication between the cores. Along with the complex design, the hardware architecture and required resources of *HECTOR-V* could limit its adaptability. Though these two TEEs use non-chip-specific features that can be implemented across FPGA vendors, there are still some constraints when it comes to these frameworks.

Additionally, despite these advancements, several critical limitations persist that must be addressed. Performance overhead, particularly in memory encryption and secure boot processes, can slow down system operations, making TEEs less viable for resource-constrained environments like IoT devices and embedded systems, where efficiency is critical. Integration complexity, especially in heterogeneous architectures, complicates the seamless coordination between secure and non-secure domains, risking potential security gaps or performance bottlenecks. Additionally, while TEEs are designed to protect against many known threats, they remain vulnerable to emerging challenges such as quantum computing and advanced side-channel attacks. These limitations are crucial because they not only constrain the current utility of TEEs but also underscore the urgent need for ongoing research to develop more efficient, adaptable, and resilient security solutions.

IV. Threats to Validity

We examine three potential threats to validity based on the classification scheme of [39] and [40].

Construct validity refers to how well the study identifies and categorizes TEEs. The search strings may have failed to capture relevant papers. This threat was mitigated by checking references of the included papers for potential oversights. Another threat is the manual categorization of papers (e.g., application-specific or feature-specific TEEs). This relies on subjective judgment. To mitigate this, possible features and applications were reviewed and rechecked.

Content validity may be affected in two ways. First, if the inclusion criteria used to select the final 27 papers were too restrictive, this would result in excluding papers that offer theoretical frameworks or nascent areas of research. This threat was minimized by reading the abstracts of all 109 papers to ensure no relevant studies were excluded. Second, only IEEE or ACM were searched, possibly excluding relevant papers published elsewhere. This is not a significant threat because IEEE and ACM conference proceedings and journals are the primary outlets for publications on edge-computing security.

External validity relates to the ability to generalize the findings of this study. We do not perceive significant threats to the external validity of this study. The scope of our study is on SoC-FPGA TEEs. Within this scope, our research captures the state of the published research. However, extrapolating or generalizing findings beyond this scope to the broader landscape of edge computing is not advised.

V. Conclusion

This study systemizes SoC-FPGA TEEs, highlighting research gaps. Through the analysis of 109 papers sourced from IEEE Xplore and ACM Digital Library, a pool of 27 papers represented the current state of SoC-FPGA-based TEEs. These papers demonstrated the research challenges of implementing a robust, multi-featured, multi-application TEE, illustrated by the emphasis on application and feature-based TEEs. A robust, modular approach emerged in two papers combining critical features for a non-application-specific approach. The lack of publications related to SoC-FPGA-based TEEs that do not rely on third-party technology reveals a gap in the literature and an opportunity for researchers and developers (Figure 1). Many papers emphasize specific applications or features, but few combine features to create extensible TEEs. These insights hold significance for future development of TEEs and emphasize the importance of secure computing across applications and platforms.

VI. Acknowledgments

NASA and Resilient Computing, LLC supported this research under award number 80NSSC23CA147, and subcontract number 4W9082, respectively. This research was conducted with the U.S. Department of Homeland Security (DHS) Science and Technology Directorate (S&T) under contract 70RSAT24KPM000022. Any opinions contained herein are those of the authors and do not necessarily reflect those of NASA, Resilient Computing, LLC or DHS S&T. Thank you to Yvette Hastings at MSU for assisting with Figs. 1 and 2.

References

[1] Pearson *et al.*, "Privacy, security and trust issues arising from cloud computing," in *2010 IEEE Second International Conf on Cloud Computing Technology and Science*, 2010, pp. 693–702.

[2] Xiao *et al.*, "Edge computing security: State of the art and challenges," *Proceedings of the IEEE*, vol. 107, no. 8, pp. 1608–1631, 2019.

[3] Zeyu *et al.*, "Survey on edge computing security," in *2020 International Conf on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE)*, 2020, pp. 96–105.

[4] Zhang *et al.*, "Data security and privacy-preserving in edge computing paradigm: Survey and open issues," *IEEE Access*, vol. 6, pp. 18 209–18 237, 2018.

[5] D. Kaplan, "Protecting vm register state with sev-es," Advanced Micro Devices, Inc., February 2017, white Paper. [Online]. Available: https://www.amd.com/content/dam/amd/en/documents/epyc-business-docs/white-papers/Protecting-VM-Register-State-with-SEV-ES.pdf

[6] Lee *et al.*, "Keystone: an open framework for architecting trusted execution environments," in *Proceedings of the Fifteenth European Conf on Computer Systems*, ser. EuroSys '20. New York, NY, USA: ACM, 2020. [Online]. Available: https://doi.org/10.1145/3342195.3387532

[7] 2024. [Online]. Available: https://developer.arm.com/documentation/PRD29-GENC-009492/latest/

[8] Arm, "What is fpga?" [Online]. Available: https://www.arm.com/glossary/fpga

[9] J. Schneider and I. Smalley, "What is a field programmable gate array (fpga)?" May 2024. [Online]. Available: https://www.ibm.com/think/topics/field-programmable-gate-arrays

[10] [Online]. Available: https://www.intel.com/content/www/us/en/products/programmable/fpga-vs-structured-asic.html

[11] Nasahl *et al.*, "Hector-v: A heterogeneous cpu architecture for a secure risc-v execution environment," in *Proceedings of the 2021 ACM Asia Conf on Computer and Communications Security*, ser. ASIA CCS '21. New York, NY, USA: ACM, 2021, p. 187–199. [Online]. Available: https://doi.org/10.1145/3433210.3453112

[12] Yoshida *et al.*, "Towards trusted iot sensing systems: Implementing puf as secure key generator for root of trust and message authentication code," in *Proceedings of the 10th International Workshop on Hardware and Architectural Support for Security and Privacy*, ser. HASP '21. New York, NY, USA: ACM, 2022. [Online]. Available: https://doi.org/10.1145/3505253.3505258

[13] B. A. Kitchenham, *et al.*, "Using mapping studies as the basis for further research–a participant-observer case study," *Information and Software Technology*, vol. 53, no. 6, pp. 638–651, 2011.

[14] Zhao *et al.*, "Shef: Shielded enclaves for cloud fpgas," in *Proceedings of the 27th ACM International Conference on Architectural Support for Programming Languages and Operating Systems*, 2022, pp. 1070–1085.

[15] Zhu *et al.*, "Tacc: a secure accelerator enclave for ai workloads," in *Proceedings of the 15th ACM International Conf on Systems and Storage*, ser. SYSTOR '22. New York, NY, USA: ACM, 2022, p. 58–71. [Online]. Available: https://doi.org/10.1145/3534056.3534943

[16] Ren *et al.*, "Accshield: a new trusted execution environment with machine-learning accelerators," in *2023 60th ACM/IEEE DAC*, 2023, pp. 1–6.

[17] Kolimbianakis *et al.*, "Software-defined hardware-assisted isolation for trusted next-generation iot systems," in *Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing*, ser. SAC '22. New York, NY, USA: ACM, 2022, p. 139–146. [Online]. Available: https://doi.org/10.1145/3477314.3508378

[18] Ren *et al.*, "Accguard: Secure and trusted computation on remote fpga accelerators," in *2021 IEEE iSES*, 2021, pp. 378–383.

[19] Jasti *et al.*, "Security in multi-tenancy cloud," in *44th Annual 2010 IEEE International Carnahan Conf on Security Technology*, 2010, pp. 35–41.

[20] Oh *et al.*, "Meetgo: A trusted execution environment for remote applications on fpga," *IEEE Access*, vol. 9, pp. 51 313–51 324, 2021.

[21] Wang *et al.*, "Operon: an encrypted database for ownership-preserving data management," *Proc. VLDB Endow.*, vol. 15, no. 12, p. 3332–3345, Aug. 2022. [Online]. Available: https://doi.org/10.14778/3554821.3554826

[22] Ahmed *et al.*, "Trusted ip solution in multi-tenant cloud fpga platform," in *2022 IEEE 8th WF-IoT*, 2022, pp. 1–6.

[23] Ince *et al.*, "Token-based authentication and access delegation for hw-accelerated telco cloud solution," in *2022 IEEE 11th International Conf on CloudNet*, 2022, pp. 109–117.

[24] Zhu *et al.*, "Chaosintc: A secure interrupt management mechanism against interrupt-based attacks on tee," in *2023 60th ACM/IEEE DAC*, 2023, pp. 1–6.

[25] Mao *et al.*, "Rehad: Using low-frequency reconfigurable hardware for cache side-channel attacks detection," in *2020 IEEE EuroS&PW*, pp. 704–709.

[26] Malekpour *et al.*, "Hardware trojan detection and recovery in mpsocs via on-line application specific testing," in *2019 IEEE 22nd International Symposium on DDECS*, 2019, pp. 1–6.

[27] Schilling *et al.*, "Secwalk: Protecting page table walks against fault attacks," in *2021 IEEE International Symposium HOST*, pp. 56–67.

[28] Khan *et al.*, "Utilizing and extending trusted execution environment in heterogeneous socs for a pay-per-device ip licensing scheme," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 2548–2563, 2021.

[29] Chen *et al.*, "A risc-v system-on-chip based on dual-core isolation for smart grid security," in *2022 IEEE 6th Conf EI2*, 2022, pp. 1278–1283.

[30] Turan *et al.*, "Propagating trusted execution through mutual attestation," in *Proceedings of the 4th Workshop on System Software for Trusted Execution*, ser. SysTEX '19. New York, NY, USA: ACM, 2019. [Online]. Available: https://doi.org/10.1145/3342559.3365334

[31] Stoyanov *et al.*, "Secure heterogeneous architecture based on risc-v and root-of-trust," in *Proceedings of the 24th International Conf on Computer Systems and Technologies*, ser. CompSysTech '23. New York, NY, USA: ACM, 2023, p. 19–23. [Online]. Available: https://doi.org/10.1145/3606305.3606312

[32] Zou *et al.*, "Ares: Persistently secure non-volatile memory with processor-transparent and hardware-friendly integrity verification and metadata recovery," *ACM Trans. Embed. Comput. Syst.*, vol. 21, no. 1, feb 2022. [Online]. Available: https://doi.org/10.1145/3492735

[33] Streit *et al.*, "Secure boot from non-volatile memory for programmable soc architectures," in *2020 IEEE International Symposium HOST*, 2020, pp. 102–110.

[34] Hoang *et al.*, "Trusted execution environment hardware by isolated heterogeneous architecture for key scheduling," *IEEE Access*, vol. 10, pp. 46 014–46 027, 2022.

[35] Du *et al.*, "Accelerating extra dimensional page walks for confidential computing," in *Proceedings of the 56th Annual IEEE/ACM International Symposium on Microarchitecture*, ser. MICRO '23. New York, NY, USA: ACM, 2023, p. 654–669. [Online]. Available: https://doi.org/10.1145/3613424.3614293

[36] Cilardo *et al.*, "Memory encryption support for an fpga-based risc-v implementation," in *2021 16th International Conf on Design & Technology of Integrated Systems in Nanoscale Era (DTIS)*, 2021, pp. 1–5.

[37] Kumar *et al.*, "Post-quantum secure boot," in *Proceedings of the 23rd Conf on Design, Automation and Test in Europe*, ser. DATE '20. San Jose, CA, USA: EDA Consortium, 2020, p. 1582–1585.

[38] Xi *et al.*, "A heterogeneous risc-v soc for confidential computing and password recovery," in *2022 7th International Conf on Integrated Circuits and Microsystems (ICICM)*, 2022, pp. 500–504.

[39] Cook *et al.*, *Quasi-experimentation: Design & Analysis Issues for Field Settings*. Houghton Mifflin, 1979. [Online]. Available: https://books.google.com/books?id=BFNqAAAAMAAJ

[40] Campbell *et al.*, *Experimental and Quasi-experimental Designs for Research*. R. McNally, 1966. [Online]. Available: https://books.google.com/books?id=kFtqAAAAMAAJ