The OSCAR PUBLISHING Services
American Journal of Applied Science and Technology

VOLUME Vol.05 Issue 10 2025
PAGE NO.303-308

Multi-Tenant Cloud Architectures Utilizing FPGAs: Security Challenges, Design Methodologies, and Proposed Paradigms

Dr. Arjun Mehta
Department of Computer Engineering, Global University, Toronto, Canada

Received: 01 October 2025; **Accepted:** 16 October 2025; **Published:** 31 October 2025

Abstract: As cloud computing continues its rapid ascent, the integration of reconfigurable hardware such as Field-Programmable Gate Arrays (FPGAs) into multi-tenant cloud infrastructures presents both tremendous opportunities and formidable security challenges. This paper synthesizes contemporary academic and technical literature to examine the underlying threats, architectural solutions, and trust models in FPGA-augmented cloud systems. We analyze attack vectors including side-channel leakage, remote power analysis, thermal covert channels, and hardware-level tampering, as documented in seminal works. In response, we propose a conceptual methodology integrating hardware isolation, trusted execution environments, netlist-level obfuscation, self-attestation, and secure resource management to achieve robust security in multi-tenant environments. Through an in-depth theoretical analysis and comparative assessment of existing designs, we outline a comprehensive architecture that balances performance, flexibility, and security. Our findings highlight design trade-offs, limitations of current approaches, and future research directions including improved attestation protocols and dynamic isolation mechanisms.

Keywords: Cloud security; FPGA; multi-tenancy; trusted execution environment; side-channel attacks; hardware isolation; self-attestation

INTRODUCTION:

The evolution of cloud computing over the past decade has transformed how computational resources are provisioned and consumed. Cloud models, particularly multi-tenant architectures, afford scalability, cost-efficiency, and flexibility, enabling multiple disparate clients to share hardware resources. With the augmentation of cloud infrastructures by hardware accelerators such as Field-Programmable Gate Arrays (FPGAs), there arises the potential to dramatically enhance performance for tasks like machine learning inference, cryptographic operations, and data processing. However, this integration brings unprecedented security challenges. Unlike traditional virtualized cloud setups, FPGA-based cloud systems blur the lines between hardware and software isolation, posing risks not just at virtualization boundaries, but deep within programmable hardware logic itself.

While virtualization security in traditional cloud infrastructures has been extensively studied (Pek et al., 2013; Brooks et al., 2012; Al-Jahdali et al., 2013), the novel integration of reconfigurable logic introduces new and more insidious vulnerabilities. Multi-tenancy in the cloud demands strict isolation, but reconfigurable fabrics shared across tenants may allow covert channels, side-channel leakage, and unauthorized access to shared hardware primitives. Recent years have witnessed a nascent but growing body of research focusing on securing FPGA-augmented clouds. In this context, our work seeks to integrate these disparate findings into a unified conceptual framework.

The core problem addressed in this paper is: How can cloud providers deliver FPGA-based acceleration in multi-tenant environments while ensuring strong security guarantees comparable to—or exceeding—those in traditional virtualization frameworks? Specifically, we examine threats at the hardware level (power analysis, thermal channels, hardware tampering), weaknesses in existing trust models, and propose a holistic, secure architecture that addresses these vulnerabilities. Our contribution is not an empirical measurement, but rather a rigorous theoretical analysis grounded in the extant literature, offering a roadmap toward secure, production-ready

FPGA cloud systems.

In the sections that follow, we: (a) survey the landscape of attacks and security challenges in FPGA cloud contexts; (b) analyze existing defensive architectures, including hardware isolation, trusted execution environments, netlist obfuscation, and self-attestation; (c) propose a comprehensive methodology for secure multi-tenant FPGA deployment; (d) discuss the trade-offs and limitations of our proposed design; and (e) chart directions for future research.

METHODOLOGY

Given the theoretical nature of this study, our methodology comprises a systematic literature review and comparative architectural analysis, synthesizing insights across prominent works to identify recurring threats and defenses, then constructing an integrated security architecture.

Literature Review and Threat Taxonomy

Our first step was to examine the literature for documented threats and vulnerabilities in FPGA-based systems within multi-tenant or cloud-like environments. We identified key classes of attacks:

- Side-channel leakage via power analysis or thermal channels, as demonstrated by Schellenberg et al. (2018) in remote power analysis attacks on FPGAs, and by Tian & Szefer (2019) in their work on temporal thermal covert channels in cloud FPGAs.
- Hardware-level threats including oscillator-based attacks targeting FPGA timing and clocking mechanisms in data centers (Sugawara et al., 2019).
- Tampering, unauthorized reconfiguration, and netlist-level attacks, as explored by Chakraborty & Bhunia (2008) in netlist-level obfuscation, and by Vliegen et al. (2019) in self-attestation of configurable hardware (SACHa).
- Trust and isolation deficiencies in FPGA clouds, as critically reviewed by Turan & Verbauwhede (2020), along with early foundational definitions of trusted execution environments (Sabt, Achemlal & Bouabdallah, 2015).
- Resource-sharing risks, including memory sharing across FPGA and CPU platforms, addressed by Vogel, Marongiu & Benini (2019) in their configurable IOMMU-based shared virtual memory design.

Simultaneously, we surveyed architectures and proposals intended to mitigate these threats: hardware isolation in FPGA-accelerated embedded systems (Saha & Bobda, 2020), multi-level security using open-source rooted trust (Saha et al., 2023), netlist obfuscation, self-attestation, and cluster-based FPGA cloud designs (Tarafdar et al., 2017; Skhiri et al., 2019).

Comparative Architectural Analysis

In the second phase, we analyzed these defensive strategies along key dimensions: efficacy against different threat classes; feasibility in cloud-scale deployment; impact on performance; and complexity of implementation. This involved abstracting each defense into functional capabilities (e.g., power isolation, attestation, isolation, resource management), then evaluating overlaps and gaps.

Synthesis into an Integrated Architecture

Drawing on the strengths of existing work and seeking to mitigate their limitations, we assembled a conceptual architecture that combines hardware isolation, trusted execution base, netlist obfuscation, self-attestation, configurable IOMMU, and tenant-aware resource orchestration. We describe how each component interacts, outline secure workflows for FPGA allocation and de-allocation among tenants, and explain how safety is maintained dynamically across reconfiguration cycles.

Threat Modeling and Security Analysis

Finally, we apply a threat-modeling perspective: for each threat vector identified, we describe how the proposed architecture would detect, prevent, or mitigate the attack, and where residual risk remains. We also outline limitations and future enhancements.

This methodology, although non-empirical, yields a detailed, theoretically grounded design and evaluation that is ready for future empirical validation.

RESULTS

Our analysis yields a detailed mapping between known FPGA-cloud threats and defensive mechanisms, revealing gaps in current literature and architectures. Below we describe key findings and the resulting conceptual design, followed by an assessment of how this design mitigates specific threat vectors.

Threat–Defense Mapping

1. Power and Thermal Side-Channel Attacks

The literature confirms that remote adversaries can monitor power usage or thermal variations in shared FPGA resources to infer sensitive computations (Schellenberg et al., 2018; Tian & Szefer, 2019). These attacks exploit shared power rails, cooling systems, and thermal coupling between hardware blocks.

Existing cloud FPGA systems often lack hardware-level isolation of power and thermal domains, allowing cross-tenant leakage. Defensive proposals

rarely address side-channel isolation explicitly—highlighting a crucial gap.

2. Hardware Timing/Clock Attacks

As shown by Sugawara et al. (2019), oscillator design flaws or malicious oscillator configurations can create timing anomalies exploitable in centralized data center FPGAs. Since many FPGA clouds rely on shared clock trees and centralized clock distribution, the risk is amplified.

Literature has not yet fully addressed secure clock isolation or dedicated oscillator management per tenant in cloud FPGA environments.

3.Netlist-Level Tampering and Unauthorized Reconfiguration

Hardware IP theft, backdoors, or unauthorized netlist alterations are long-standing concerns in reconfigurable computing (Chakraborty & Bhunia, 2008). These attacks may persist across reconfiguration cycles if netlists are not obfuscated or verified.

Meanwhile, self-attestation frameworks like SACHa (Vliegen et al., 2019) offer promising methods for verifying FPGA configuration integrity on deployment. However, they have limitations in dynamic cloud environments, especially when reconfiguration occurs frequently and by multiple tenants.

4. Resource Sharing and Memory Isolation Issues

The use of shared memory spaces, or insufficient isolation between CPU and FPGA memory, can enable side-channel or data leakage (Vogel, Marongiu & Benini, 2019). In traditional cloud virtualization, IOMMU and memory virtualization help; yet in FPGA clouds, configurable IOMMUs are still not widespread.

For secure multi-tenant operation, resource sharing must be governed by strict isolation, enforced via hardware and firmware mechanisms.

5. Trust Base Deficiency

The definition and nuances of trusted execution environments (TEEs) for reconfigurable hardware, as articulated by Sabt, Achemlal & Bouabdallah (2015), indicates that a TEE must include secure boot, attestation, and isolation. However, most FPGA-based cloud proposals either omit TEEs or rely solely on software-based isolation, insufficient against hardware-level threats (Turan & Verbauwhede, 2020).

Without a rooted hardware trust base, the entire multi-tenant cloud FPGA architecture remains vulnerable to advanced adversaries.

Conceptual Architecture for Secure Multi-Tenant FPGA Cloud

Based on the above mapping, we propose an integrated architecture comprising the following core components:

● Dedicated Hardware Isolation Modules

Each tenant's FPGA logic is instantiated within a dedicated isolation zone, with separate power rails, dynamic voltage and frequency scaling (DVFS) domains, and isolated cooling/heat sinks. Physical partitioning within the FPGA ensures thermal and power independence, aiming to eliminate cross-tenant side-channel leakage.

● Tenant-Specific Clock & Oscillator Management

Rather than a shared global clock distribution, each tenant receives a dedicated programmable oscillator or Phase-Locked Loop (PLL) instance. Clock configuration is restricted to a constrained set of safe parameters; only firmware signed by the cloud provider can alter oscillator settings, preventing timing-based attacks like those described by oscillator manipulation research (Sugawara et al., 2019).

● Netlist Obfuscation & Secure Bitstream Encryption

Before deployment, tenant bitstreams undergo netlist-level obfuscation (as per Chakraborty & Bhunia, 2008), and bitstreams are encrypted using strong cryptographic primitives. This prevents IP theft, unauthorized netlist tampering, or backdoor insertion. Key management is handled by a centralized cloud key management service, with keys never exposed to tenant environments.

● Rooted Trust and Trusted Execution Environment (TEE) for FPGA

On hardware initialization, the FPGA bootloader verifies the bitstream signature using a manufacturer-provisioned root of trust. Once verified, the reconfigurable logic is instantiated in a secure enclave. This TEE ensures that neither the tenant nor the cloud operator can inject malicious logic post-deployment without detection (Sabt, Achemlal & Bouabdallah, 2015). The enclave also restricts access to configuration interfaces and power/clock controls.

● Self-Attestation Mechanisms Before & After Reconfiguration

Borrowing from the self-attestation model of SACHa (Vliegen et al., 2019), the FPGA periodically attests its configuration status to the cloud resource manager. On every reconfiguration or tenant swap, attestation ensures that only approved bitstreams are loaded,

and that no unauthorized modifications have occurred. This process uses cryptographic hashes and timestamps to detect anomalies.

- Configurable IOMMU and Memory Isolation

To prevent data leakage via shared memory or suspicious DMA behavior, all memory accesses—CPU or FPGA initiated—are mediated by a configurable IOMMU (as proposed by Vogel, Marongiu & Benini, 2019). Each tenant receives a dedicated virtual memory region; cross-tenant memory access attempts are blocked. Furthermore, caching and shared memory primitives are disabled unless explicitly permitted in isolated zones.

- Secure Resource Orchestration & Tenant-aware Scheduling

The cloud orchestration layer enforces that FPGAs are exclusively allocated per tenant; dynamic reallocation triggers a secure teardown process that wipes bitstreams, power cycles isolation zones, and reinitializes the TEE. Scheduling algorithms are augmented to avoid simultaneous neighbor execution of tenants with differing trust levels or conflicting workloads—reducing risk of side-channel spill-over.

Security Analysis of Proposed Architecture

Applying threat modeling:

- Against Power/Thermal Side-Channel Attacks: By providing isolated power rails, independent cooling, and dedicated clock domains, cross-tenant leakage paths via shared hardware physical resources are effectively severed. Thermal variations from one tenant cannot propagate into another's domain, neutralizing temporal thermal covert channels (Tian & Szefer, 2019).

- Against Oscillator/Clock-based Attacks: Dedicated clock domains and controlled oscillator configuration eliminate adversary control over timing primitives, mitigating the oscillator-based vulnerabilities demonstrated by Sugawara et al. (2019).

- Against Netlist Tampering and IP Theft: Bitstream encryption and netlist obfuscation ensure that even if an attacker intercepts configuration traffic, they cannot reverse-engineer or inject malicious logic. Self-attestation further ensures configuration integrity before and after reconfiguration, preventing unauthorized modifications between tenant cycles.

- Against Memory-based Leaks or DMA Abuse: The configurable IOMMU enforces strong memory isolation akin to mature virtualization platforms; shared memory hazards between CPU and FPGA are neutralized. Tenant DMA is constrained to allocated memory segments, preventing covert data exfiltration.

- Against Root-of-Trust and Configuration Integrity Attacks: The TEE rooted in hardware boot ensures bitstream provenance, and dynamic attestation guards against runtime tampering. Since reconfiguration triggers a secure teardown and re-initialization, any residual configuration or state is erased before another tenant gains access.

DISCUSSION

Our proposed architecture represents a significant advance in conceptualizing secure multi-tenant FPGA cloud deployment. It synthesizes best practices and emerging defensive strategies into a unified framework—providing a theoretically robust foundation for real-world implementation. However, this architecture is not without limitations, and several challenges remain before it can be widely adopted.

Performance Overheads and Resource Utilization

Isolating each tenant in terms of power rails, cooling, and clock domains incurs overhead. In a large-scale data center, dedicating separate hardware resources per tenant reduces the benefits of resource multiplexing and pooling that make cloud computing economically attractive. The addition of dedicated oscillators or PLLs, independent cooling systems, and physical partitioning may substantially increase costs. Moreover, the act of secure teardown and re-initialization between tenant allocations will add latency and reduce resource utilization efficiency.

Netlist obfuscation and bitstream encryption may also degrade performance; resource utilization could diminish due to obfuscation-related inefficiencies. Further, configurable IOMMU and memory isolation layers introduce extra layers of address translation and access control, possibly affecting throughput and latency of memory and I/O operations.

Complexity and Manageability

Implementing such an architecture requires deep integration between firmware, hardware, orchestration layers, and security infrastructure. Cloud providers would need to redesign FPGA-based offerings from the ground up. The management overhead—tracking per-tenant isolation zones, attestation logs, key management, encryption, and secure resource scheduling—could be substantial. Maintaining this complexity at scale across thousands of tenants and devices may introduce new operational vulnerabilities or performance bottlenecks.

Attestation Scalability and Freshness

While self-attestation (as in SACHa) is a promising

mechanism, ensuring freshness of attestation and preventing replay attacks in a dynamic multi-tenant environment is nontrivial. Frequent reconfiguration demands that attestation be both efficient and secure. Cryptographic overhead, key management, secure timestamping, and prevention of replay or rollback attacks all present non-trivial engineering challenges.

Residual Side-Channel and Covert Channel Risks

Although our architecture aims to eliminate power and thermal leakage across tenants, absolute elimination may not be feasible. Subtle leakage via shared facility-level resources—power distribution networks, building cooling, shared ambient temperature zones—might still be exploited by sophisticated adversaries. Additionally, other covert channels may exist, such as electromagnetic emanations, acoustic channels, or routing-based timing channels, which our current architecture does not explicitly address.

Trust in Cloud Provider and Supply-Chain Risks

The entire security model depends on the cloud provider correctly implementing isolation, key management, and attestation systems. If any component of the supply chain—FPGA vendor, firmware developers, hardware manufacturers—is compromised, the root-of-trust may be weakened. Such supply-chain risks are especially critical when hardware components originate from multiple vendors or when third-party IP cores are incorporated.

Future Research Directions

Given these limitations, further research is required to turn this conceptual architecture into a deployable system. Key areas include:

- Empirical evaluation of isolation mechanisms: Prototype implementations are needed to measure actual side-channel leakage (power, thermal, electromagnetic) under the proposed isolation schemes. This will help quantify residual risks and guide refinements.
- Efficient self-attestation protocols: Develop lightweight yet secure attestation mechanisms with freshness guarantees, minimal performance overhead, and robust defenses against replay or rollback.
- Dynamic resource orchestration algorithms: Design scheduling policies that balance security with utilization efficiency, perhaps by grouping tenants with similar trust profiles or workloads.
- Advanced covert-channel analysis: Extend threat modeling to include electromagnetic, acoustic, and routing-based channels; design isolation or detection mechanisms accordingly.
- Supply-chain security frameworks: Investigate approaches for verifying hardware provenance, ensuring firmware integrity, and securing third-party IP cores within a cloud FPGA ecosystem.

CONCLUSION

The integration of FPGAs into multi-tenant cloud environments holds immense promise for accelerating diverse workloads while offering flexibility. However, it also exposes novel hardware-level vulnerabilities that traditional virtualization-based security models are ill-equipped to handle. Through a comprehensive literature review and architectural analysis, this paper has identified key threats—power and thermal side-channels, clock-based attacks, netlist tampering, memory leakage, and root-of-trust weaknesses—and demonstrated that no singular existing technique suffices to defend against them in isolation.

Our proposed architecture represents a holistic, defense-in-depth blueprint combining hardware isolation, secure clock management, netlist obfuscation, bitstream encryption, root-of-trust-based TEEs, self-attestation, configurable IOMMU-based memory isolation, and tenant-aware resource scheduling. While theoretically robust, practical deployment will inevitably involve trade-offs in cost, performance, complexity, and manageability. As such, this work should be viewed not as a final, deployable product, but as a foundational design paradigm that guides future empirical and engineering efforts.

We believe that by pursuing this integrated approach—and by rigorously evaluating, refining, and extending it in real-world settings—the research community and industry practitioners can realize secure, efficient, and trustworthy FPGA-based multi-tenant cloud services that meet the high standards of modern cloud security.

REFERENCES

1. Sandip Ray and Yier Jin. 2015. Security policy enforcement in modern SoC designs. In Proceedings of the IEEE/ACM International Conference on Computer-Aided Design (ICCAD ’15), IEEE Press, Piscataway, NJ, 345–350.
2. Sujoy Sinha Roy, Furkan Turan, Kimmo Järvinen, Frederik Vercauteren, and Ingrid Verbauwhede. 2019. FPGA-based High-Performance Parallel Architecture for Homomorphic Computing on Encrypted Data. Cryptology ePrint Archive, Paper 2019/160.

3. M. Sabt, M. Achemlal, and A. Bouabdallah. 2015. Trusted execution environment: What it is, and what it is not. In Trustcom/BigDataSE/ISPA, IEEE, Vol. 1, 57–64.
4. Sujan Kumar Saha and Christophe Bobda. 2020. FPGA accelerated embedded system security through hardware isolation. In Proceedings of the Asian Hardware Oriented Security and Trust Symposium (AsianHOST ’20), IEEE, 1–6.
5. Sujan Kumar Saha, Abigail N. Butka, Muhammed Kawser Ahmed, and Christophe Bobda. 2023. OpenTitan based multi-level security in FPGA system-on-chips. In Proceedings of the International Conference on Field Programmable Technology (ICFPT ’23), 302–303.
6. Falk Schellenberg, Dennis R. E. Gnad, Amir Moradi, and Mehdi B. Tahoori. 2018. An inside job: Remote power analysis attacks on FPGAs. In Proceedings of the Design, Automation and Test in Europe Conference and Exhibition (DATE ’18), 1111–1116.
7. Rym Skhiri, Virginie Fresse, Jean Paul Jamont, Benoit Suffran, and Jihene Malek. 2019. From FPGA to support cloud to cloud of FPGA: State of the art. International Journal of Reconfigurable Computing 2019, 8085461.
8. Hayden Kwok-Hay So and Robert W. Brodersen. 2007. BORPH: An Operating System for FPGA-Based Reconfigurable Computers. Ph.D. Dissertation, EECS Department, University of California, Berkeley.
9. Rajat Subhra Chakraborty and Swarup Bhunia. 2008. Hardware protection and authentication through netlist level obfuscation. In Proceedings of the 2008 IEEE/ACM International Conference on Computer-Aided Design, 674–677.
10. T. Sugawara, K. Sakiyama, S. Nashimoto, D. Suzuki, and T. Nagatsuka. 2019. Oscillator without a combinatorial loop and its threat to FPGA in data centre. Electronics Letters 55, 11 (2019), 640–642.
11. Naif Tarafdar, Nariman Eskandari, Thomas Lin, and Paul Chow. 2018. Designing for FPGAs in the cloud. IEEE Design & Test 35, 1 (2018), 23–29.
12. Naif Tarafdar, Thomas Lin, Eric Fukuda, Hadi Bannazadeh, Alberto Leon-Garcia, and Paul Chow. 2017. Enabling flexible network FPGA clusters in a heterogeneous cloud data center. In Proceedings of the 2017 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays (FPGA ’17), 237–246.
13. Shanquan Tian and Jakub Szefer. 2019. Temporal thermal covert channels in cloud FPGAs. In Proceedings of the 2019 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays (FPGA ’19), ACM, New York, NY, 298–303.
14. Furkan Turan and Ingrid Verbauwhede. 2020. Trust in FPGA accelerated cloud computing. ACM Computing Surveys 53, 6, Article 128.
15. Victor Costan and Srinivas Devadas. 2016. Intel SGX Explained. Cryptology ePrint Archive.
16. Jo Vliegen, Md Masoom Rabbani, Mauro Conti, and Nele Mentens. 2019. SACHa: Self-attestation of configurable hardware. In Proceedings of the Design, Automation & Test in Europe Conference & Exhibition (DATE ’19), 746–751.
17. Pirmin Vogel, Andrea Marongiu, and Luca Benini. 2019. Exploring shared virtual memory for FPGA accelerators with a configurable IOMMU. IEEE Transactions on Computers 68, 4 (2019), 510–525.
18. Zane Weissman, Thore Tiemann, Daniel Moghimi, Evan Custodio, Thomas Eisenbarth, and Berk Sunar. 2019. JackHammer: Efficient rowhammer on heterogeneous FPGA-CPU platforms. arXiv:1912.11523.
19. Hariharan, R. 2025. Zero trust security in multi-tenant cloud environments. Journal of Information Systems Engineering and Management 10.