SAFE-SiP: Secure Authentication Framework for System-in-Package Using Multi-party Computation

Ishraq Tashdid
Department of Electrical and Computer Engineering
University of Central Florida
Orlando, FL, USA
ishraq.tashdid@ucf.edu

Tasnuva Farheen
Department of Computer Science and Engineering
Louisiana State University
Baton Rouge, LA, USA
tfarheen@lsu.edu

Sazadur Rahman
Department of Electrical and Computer Engineering
University of Central Florida
Orlando, FL, USA
mohammad.rahman@ucf.edu

Abstract

The emergence of chiplet-based heterogeneous integration revolutionizes semiconductor, AI, and high-performance computing systems by enabling modular design and enhanced scalability. However, the post-fabrication assembly of chiplets from multiple vendors introduces a complex supply chain, raising critical security concerns such as counterfeiting, overproduction, and unauthorized access. Existing solutions rely on dedicated security chiplets or modifications to the timing flow that inherently assumes a trusted SiP integrator, exposing chiplet signatures to other vendors and introducing additional attack surfaces. This work addresses these vulnerabilities by leveraging Multi-party Computation (MPC), which ensures zero-trust authentication without revealing sensitive information to any party. We introduce *SAFE-SiP*, a scalable authentication framework that garbles chiplet signatures and employs MPC for integrity verification, preventing unauthorized access and adversarial inference. *SAFE-SiP* eliminates the need for a dedicated security chiplet while ensuring authentication remains secure, even in untrusted integration environments. We evaluated *SAFE-SiP* across five RISC-V-based SiPs. Our experimental results shows that, *SAFE-SiP* achieves minimal power overhead, incurs an average area overhead of only 3.05%, and maintains a computational complexity of 2^{192}, providing a highly efficient and scalable security solution.

ACM Reference Format:
Ishraq Tashdid, Tasnuva Farheen, and Sazadur Rahman. 2025. SAFE-SiP: Secure Authentication Framework for System-in-Package Using Multi-party Computation. In *Great Lakes Symposium on VLSI 2025 (GLSVLSI '25), June 30-July 2, 2025, New Orleans, LA, USA.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3716368.3735248

1 Introduction

The shift from monolithic System-on-Chip (SoC) architectures to System-in-Package (SiP) based heterogeneous integration (HI) is crucial for sustaining yield in advanced nodes, co-integrating diverse technologies, and enabling 3D stacking for improved performance and efficiency [9, 21]. By modularizing functionalities into chiplets, SiP enhances design flexibility at the cost of challenges in co-design, reliability, and security [8, 30]. Unlike traditional SoCs, SiP requires post-fabrication assembly of multi-vendor chiplets, often in untrusted environments, making it vulnerable to counterfeiting, overproduction, spoofing and unauthorized modifications [16, 17]. The reliance on untrusted foundries for interposer fabrication and SiP integration further exacerbates risks such as hardware trojan insertion and backdoor attacks [12]. Additionally, counterfeit chiplets infiltrating the supply chain threaten functional integrity and security. As shown in Fig. 1, ensuring chiplet authenticity pre- and post-fabrication is vital to system trustworthiness [33]. Without a scalable authentication mechanism, SiP-based HI remains susceptible to exploitation at supply chain stages.

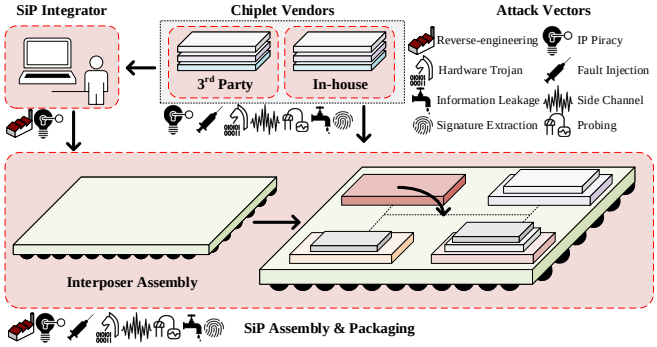

Figure 1: SiP design flow and life-cycle using third-party and in-house chiplets. Untrusted and trusted parties are marked by red and green boxes respectively.

Recent research highlights the unique security challenges in SiP but also exposes the limitations of existing approaches. Traditional techniques such as logic locking [17, 23], watermarking [4, 19], and IC metering [3] are effective for SoCs but struggle with the multi-vendor complexity of SiP [2, 24]. Recent initiatives, such as, GATE-SiP [16], PQC-HI [33], and SECT-HI [15] improve security through test access port (TAP) modifications, quantum-resistant authentication, and hardware security modules but introduce significant area overhead. Fabrication-level techniques such as split manufacturing and network-on-interconnect (NoI) obfuscation mitigate risks like cloning and tampering [29, 32]. However, split manufacturing is costly and depends on trusted BEOL foundries, facing yield and scalability issues [13, 24], while NoI obfuscation increases design complexity. These limitations underscore the need for scalable, cost-effective security solutions tailored to heterogeneous systems.

Multi-party Computation (MPC) is a cryptographic protocol that enables multiple parties to collaboratively compute a function over their private inputs while ensuring data privacy and correctness, even in zero-trust environments. The SiP assembly fits well with

GLSVLSI '25, New Orleans, LA, USA

ACM ISBN 979-8-4007-1496-2/2025/06
https://doi.org/10.1145/3716368.3735248

Table 1: Attack Vectors in SiP Supply Chain.

Scenario \ Entity	SiP Integrator	3rd Party Vendors	Interposer & Packaging Foundry	Attack Vector
Scenario A	Trusted	Untrusted	Untrusted	Side-Channel Attack Unauthorized Probing
Scenario B	Untrusted	Trusted	Untrusted	Signature Extraction
Scenario C	Trusted	Trusted	Untrusted	Hardware Trojan Overproduction

MPC as the integrator can use a function to evaluate the authenticity of the chiplets without knowing the actual signatures from the integrated chiplets. Moreover, garbling circuits offer a robust solution to obfuscate signatures in an HI environment by balancing security resiliency and cost. Recent works like TinyGarble [31] and MPCircuits [25] enhance efficiency and scalability in secure computations and multi-party protocols while maintaining low area overhead. Inspired by these, we consider garbling circuits to obscure chiplet signatures and employ secure MPC, safeguarding operational logic even if intercepted. Hence, this paper introduces *SAFE-SiP*, an MPC-based framework that integrates seamlessly with chiplets, adding a universal garbling scheme for authentication.

Contributions. Our main contributions are summarized below.

(1) We propose *SAFE-SiP*, a multi-party computation-based authentication framework using garbling circuits to secure chiplet integration in 2.5D/3D SiP assemblies, ensuring authentication, and secure testing while preserving data confidentiality. It is designed for broader compatibility, seamlessly integrating with diverse signature processes and supporting a secure boot mechanism. **To the best of our knowledge, this is the first chiplet authentication scheme in a zero-trust threat model.**
(2) We perform a comprehensive security analysis demonstrating *SAFE-SiP*'s resilience against threats, with garbling and SHA-256 protecting against tampering, spoofing, and replay attacks.
(3) We evaluate *SAFE-SiP* on five RISC-V-based benchmarks, achieving as low as 1.84% area overhead while maintaining a computational complexity of 2^{192} for a 64 bit security parameter.
(4) To encourage community collaboration and industry adoption, we plan to open-source our implementation at https://github.com/IshraqAtUCF/safe_sip following publication.

The rest of the paper is organized as follows—Section 2 outlines the threat model, reviews existing authentication methods, and motivates a secure, low-overhead SiP framework. Section 3 presents *SAFE-SiP*, detailing its MPC-based authentication, garbling circuits, and SHA-256 watermark protection. Section 4 analyzes security against removal, replay, tampering, and forgery. Section 5 evaluates in real-world settings, and Section 6 concludes the paper.

2 Background and Motivation

In this section, we discuss the complex SiP supply chain, the threat model, and existing system-level authentication mechanisms.

2.1 SiP Supply Chain and Threat Model

Fig. 1 illustrates the multifaceted security threats in SiP-based heterogeneous integration, highlighting vulnerabilities across chiplet vendors, SiP integrators, and foundries under a zero-trust paradigm. Third-party vendors are particularly susceptible to unauthorized probing and side-channel attacks (Scenario A, Table 1), where adversarial chiplets leverage information leakage techniques to extract

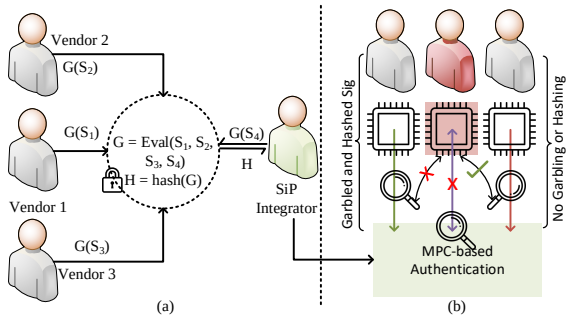

Figure 2: (a) Multi-party Computation scheme and (b) adversarial probing from malicious chiplet or 3rd-party foundry mitigated by *SAFE-SiP*.

proprietary signatures and compromise vendor confidentiality. As depicted in Fig. 1, adversarial chiplets can inject malicious modifications during fabrication, embedding covert circuits capable of intercepting authentication processes. The SiP integrator, despite being trusted in some scenarios, may also pose an untrusted environment (Scenario B), where it attempts signature extraction attacks through reverse engineering or unauthorized tampering to reuse or overproduce chiplets. This scenario is exacerbated by the presence of a compromised interposer or packaging foundry, which can facilitate the overproduction of counterfeit chiplets or introduce hardware Trojans to manipulate authentication mechanisms (Scenario C). Fig. 1 further illustrates that during SiP assembly and packaging, the interposer can serve as a conduit for various attack vectors, including hardware trojans, fault injection, and probing. The foundry, which plays a critical role in the integration process, poses significant threats by embedding backdoors within the interposer or modifying circuit layouts to facilitate unauthorized surveillance of inter-chiplet communication. These attack surfaces collectively underscore the necessity of a zero-trust authentication model that prevents any entity—whether vendor, integrator, or foundry—from gaining unauthorized access to chiplet authentication data.

2.2 Existing Works and their Drawbacks

Recent research on chiplet security focuses on fabrication-level techniques such as split manufacturing (SM) and NoI obfuscation to mitigate tampering and unauthorized access. SM obscures chiplet interconnections, reducing security risks [29], while secure routing disrupts predictable NoI paths to prevent DDoS attacks [32]. However, reliance on trusted BEOL foundries limits scalability, and SM introduces FEOL-BEOL alignment challenges that may impact yield and functionality [24]. These constraints highlight the need for holistic security solutions in chiplet-based systems. Researchers have explored integrating Chiplet Hardware Security Modules (CHSM) and Chiplet Security Intellectual Property (CSIP) into System-in-Package (SiP) architectures [12]. As shown in Tab. 2, PQC-HI employs post-quantum cryptography for chiplet authentication and key exchange, protecting against probing and unauthorized data extraction from active interposers [33]. SECT-HI, meanwhile, secures the SiP testing phase by encrypting scan chain outputs and embedding watermarks, ensuring only verified SiPs are deployed [15]. However, CHSM and CSIP add design complexity and cost, potentially attracting attackers if these modules become bypass targets, raising concerns about their practical adoption.

Table 2: Comparison with State-of-the-art Solutions (SOTA).

Existing Technique	Limitations	*SAFE-SiP*
GATE-SiP [16]	TAP-based authentication; Vulnerable to MITM attacks	No modification to the TAP; Unperturbed from MITM attacks
PQC-HI [33]	High computational overhead; Susceptible to probing attacks	Lightweight authentication with; Strong signature obfuscation
SECT-HI [15]	Limited to test encryption only; Restricts vendor security control	Considers vendor's stake in SiP; Security for both vendor & integrator
Know Time to Die [11]	Prone to challenge-response pair attacks; Sensitive to environmental variations	Environment-agnostic; Cryptographic authentication

** Unlike SOTA except [11],* SAFE-SiP *operates without dedicated security chiplet.*

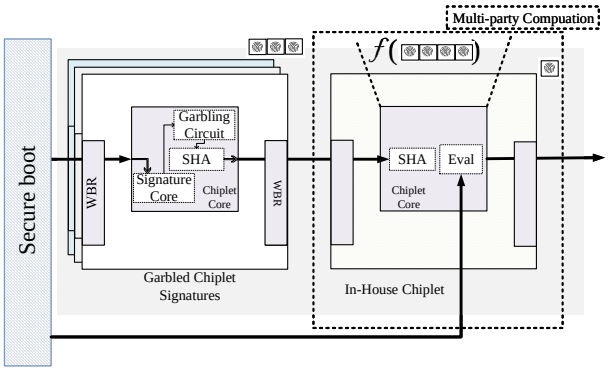

Figure 3: Detailed authentication framework using *SAFE-SiP*.

3 SAFE-SiP Methodology

In this section, we provide an overview of the *SAFE-SiP*, discuss the detailed framework, and secure-boot driven communication flow.

3.1 Security Objectives: Zero-trust Threat Model

A zero-trust threat model follows 'trust but verify' paradigm, where no entity within the SiP supply chain can be inherently trusted. Based on the threat model discussed in Sec. 2.1 we propose the following two security objectives for chiplet authentication.

(SecObj1) The interposer, facilitating communication between heterogeneous chiplets, is a critical vulnerability point due to the lack of a secure perimeter, exposing data to probing, spoofing, and man-in-the-middle attacks [11]. Securing this layer is essential to ensure system integrity, especially when chiplets from untrusted sources interact without guaranteed security features. Hence, authentication must occur without exposing sensitive data, necessitating cryptographic techniques, such as, garbling circuits. Garbled circuit is a cryptographic protocol enabling secure MPC, where two parties jointly evaluate a function on private inputs without revealing them [7, 20]. As shown in Fig. 2, vendors encode signatures into garbled values ($G(S_i)$), which the SiP integrator processes using a trusted chiplet without direct access to secret signatures [6, 35]. This enforces encrypted authentication, preventing both integrators and foundries from accessing vendor signatures. Thus mitigating side-channel attacks, probing (Scenario A, Table 1), snooping, malicious modifications, and other attack scenarios (Scenario B, C).

(SecObj2) Even with Garbling circuits in place for encrypted authentication, unauthorized chiplet modifications by the foundry may violate the integrity of garbled signatures ($G(S_i)$), jeopardizing the secure MPC. A hash operation can ensure the integrity of garbled signature. Unauthorized chiplet modifications by the foundry cause hash mismatches, making them detectable. However, if a foundry inserts a Trojan without altering the garbled circuit or signature generation, authentication alone cannot detect it—such tampering is only identifiable through dedicated post-fabrication testing. Conversely, if an integrator and foundry collude against a vendor, the vendor's signature remains protected through *SAFE-SiP*. While authentication prevents signature leakage, hardware modifications remain undetectable unless they alter garbling or signature generation, in which case testing reveals discrepancies.

By embedding security within core chiplet functionality instead of dedicated security IPs, *SAFE-SiP* minimizes attack surfaces like Trojan insertion and signature extraction. Additionally, cryptographic transformation ensures intercepted authentication data remains infeasible to reverse-engineer, preventing unauthorized overproduction. However, *SAFE-SiP* strictly verifies provenance rather than vendor trustworthiness as compromised chiplet detection is beyond authentication's scope.

3.2 Overview of *SAFE-SiP*

Building on insights from Sec. 2.2 and 3.1, the semiconductor industry demands cost-effective security solutions that integrate seamlessly with existing signature generation methods. *SAFE-SiP* addresses this need by combining a garbling circuit and a hash core (SHA-256) to enable secure and verifiable chiplet authentication. The garbling circuit obfuscates chiplet signatures, ensuring confidentiality, while the hash core generates fixed-size outputs to maintain integrity and prevent reverse engineering [16]. *SAFE-SiP* also integrates with IEEE 1500 [28], 1687 [26], and 1838 [27]-compliant Design-for-Testability (DfT) components, such as the wrapper boundary register (WBR) and wrapper instruction register (WIR), enabling structured authentication without added design complexity. Moreover, while effective for digital IPs, authenticating analog IPs poses unique challenges due to their continuous nature. To bridge this gap, an Analog-to-Digital Wrapper (ADC-W) can be introduced within *SAFE-SiP* to convert unique analog characteristics into secure digital signatures, extending authentication to mixed-signal systems. Hence, the chiplet vendor embeds the signature core, garbling circuit, and SHA-256, while the integrator performs verification on a trusted chiplet. By using *SAFE-SiP*, existing signatures achieve:

- Signature obfuscation without requiring a separate security chiplet or IP that could become an adversarial target.
- Protection against removal, replay, tampering, DDoS, and forging attacks while complying with IEEE 1500 [1].
- Data integrity and confidentiality across chiplet communications with unique, irreversible outputs in untrusted SiPs, while incurring additional low overhead in terms of area and power.

3.3 Process Flow

① The *SAFE-SiP* methodology begins with the SiP integrator sourcing physical chiplets from vendors, each embedded with a watermark generation circuit capable of producing unique signatures. These signatures, provided by the vendors, act as baselines for future authentication. The garbling circuit takes these signatures as input, generates a garbled version that is provided to the SHA-256 unit for attestation before sending out of the chiplet boundary.

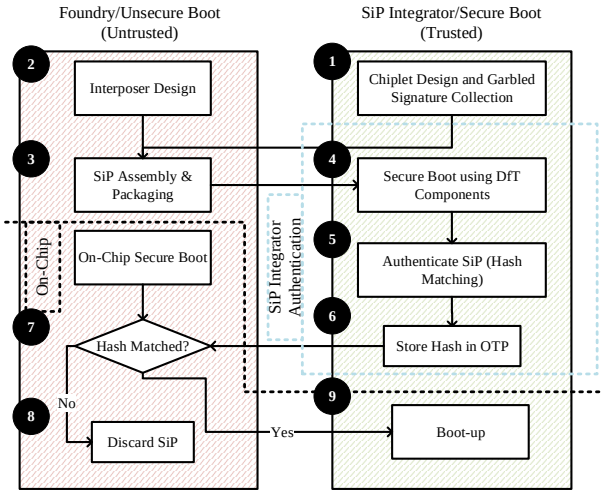

Figure 4: *SAFE-SiP* process flow with supply chain and on-chip secure boot.

② The integrator designs the interposer layout, ensuring compatibility with chiplet configurations while mitigating risks of bypass or tampering. This independent interposer design reduces vulnerabilities during the external assembly process. ③ Once chiplets and interposers are sent to a potentially untrusted packaging facility, the SiP is assembled and sent back to the SiP integrator for testing and provisioning.

④ Upon return, the SiP undergoes a secure boot mechanism, leveraging the DfT components like the WBR and WIR to initiate the authentication process, following standardized IEEE protocols [26–28]. The WBR facilitates direct interaction with chiplets, providing essential handshake signals and inputs for signature generation. Simultaneously, the WIR interprets control instructions to trigger authentication, ensuring seamless integration into the boot sequence.

⑤ During authentication, each chiplet generates garbled outputs that are validated through Multi-party Computation by comparing them against vendor-provided signatures, provided prior by the vendor themselves. Any mismatch, caused by circuit alterations or tampering, generates a signature that is different that the one provided by the chiplet vendor in step 1, leads to a different garbled and hashed output than expected, and flags the chiplet as compromised one during evaluation process by in-house chiplet. ⑥ Verified outputs are hashed and stored in one-time programmable (OTP) memory to facilitate future boot cycles.

⑦ During subsequent secure boot cycles, the system reauthenticates chiplets by comparing newly generated hashes with stored values. Any mismatch disables the compromised chiplet, preventing it from impacting the system. ⑧ This iterative reauthentication ensures operational security, with only authenticated SiPs remaining active. ⑨ Following each successful secure boot, it can be ensured that the SiP is authenticated and ready.

It is to note that, Fig. 4 illustrates the SAFE-SiP *process flow from the perspective of the SiP integrator, where the rightmost box signifies the trusted execution of the SiP integrator's authentication and secure boot process within the supply chain, and hence, marked as trusted.*

3.3.1 Garbling Circuit. The *SAFE-SiP* framework employs a garbling circuit to ensure the confidentiality and integrity of input signatures by transforming them into encrypted outputs. This transformation utilizes cryptographic labels and random bits generated by the chiplet's true random number generator, enhancing unpredictability and security [5, 18].

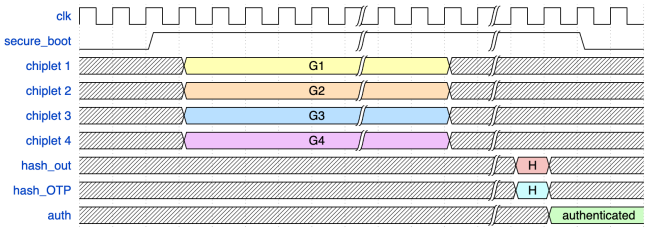

Figure 5: *SAFE-SiP* waveform showcasing secure boot protocol.

The process begins by masking each bit b^i of a W-bit signature S with a random bit r and label L, where one state is represented by r and its complement $\bar{r}$, ensuring obfuscation:

$$\tilde{b}_0^i = r_0^i \| L_0, \tag{1}$$

$$\tilde{b}_1^i = r_1^i \| L_1, \quad \text{where } r_1^i = \bar{r}_0^i \tag{2}$$

The randomness of r and $\bar{r}$ makes each bit indistinguishable without the correct masking key. The garbling transformation is expressed:

$$S \in (0,1)^W \rightarrow G \in (0,1)^g, \quad \text{where } g = W \cdot \kappa \tag{3}$$

where κ is the security parameter, defining the length of labels r and L. This process ensures strong encryption, preventing unauthorized data extraction and reverse engineering [23]. The integration of cryptographic masking and cyclic encryption secures data throughout the lifecycle, making *SAFE-SiP* both robust and efficient.

3.3.2 SHA-256 Core. The SHA-256 hashing algorithm plays a crucial role in *SAFE-SiP*, ensuring data integrity and authenticity. Recognized for its resistance to collision attacks, SHA-256 generates fixed-size hashes, making it computationally infeasible to retrieve original messages or produce identical hash outputs [14].

Within *SAFE-SiP*, SHA-256 secures the garbled outputs $G(S_i)$, verifying their integrity:

$$H = \text{Hash}(G(S_1), G(S_2), G(S_3), G(S_4)) \tag{4}$$

$$E = \text{Eval}(H) \tag{5}$$

Any tampering with $G(S_i)$ alters H, making unauthorized modifications detectable. By leveraging SHA-256, *SAFE-SiP* prevents data manipulation and ensures secure communication all over [10, 22].

4 Security Analysis

SAFE-SiP utilizes built-in True Random Number Generators (TRNGs) in modern chiplets to securely generate cryptographic labels (L_0, L_1) and masking bits $(r, \bar{r})$ for garbling input signatures as discussed in Sec. 3.3.1. Unlike pseudo-random generators, TRNGs derive randomness from unpredictable physical phenomena, ensuring secure and independent parameter creation. This unpredictability is vital for the garbling process, as compromised or predictable RNGs could expose garbled outputs, enabling adversarial inferences. The following section discusses the various adversarial scenarios in SiP assembly and post-distribution and *SAFE-SiP*'s resilience to them.

4.1 Removal Attacks

Removal attacks pose a significant threat in chiplet-based systems, where an adversarial foundry may capture the authentication mechanism, record its outputs, remove the underlying logic, and insert

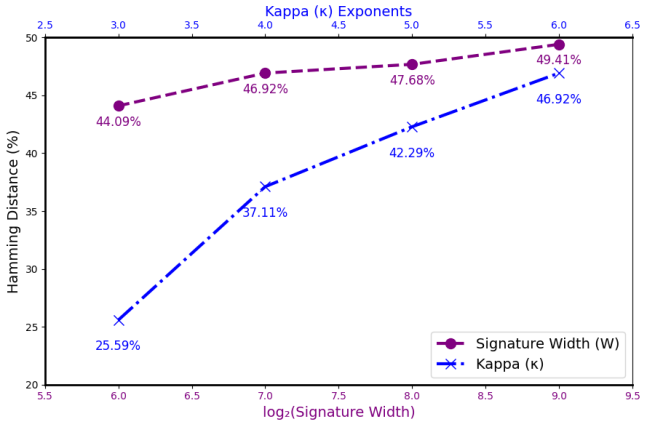

Figure 6: *SAFE-SiP*'s resistance against fault injection attacks. *Blue*: fixed signature width (*64*), and *Purple*: fixed κ (*64*).

malicious modifications into the chiplet. This attack risks compromising the integrity of the chiplet and may lead to unauthorized functionality or leakage of sensitive information. The *SAFE-SiP* framework is inherently resistant to removal attacks due to the integration of its garbling circuit with the watermarking circuitry of the chiplet. The garbled circuit outputs are indistinguishable from normal circuit outputs, making it extremely challenging for an attacker to identify and isolate the authentication mechanism. This obfuscation is further enhanced by the seamless embedding of the authentication process into the chiplet's operational flow, ensuring that the garbling logic is intertwined with the functional logic in a manner that does not expose a distinct authentication structure. Moreover, additional security enhancements such as logic locking and circuit camouflaging, can provide layers of protection against removal or reverse engineering.

4.2 Replay Attacks

Replay attacks occur when adversaries reuse stored outputs or employ brute force methods [34]. The *SAFE-SiP* framework counters these attacks with layered defenses, yielding a computational complexity as show in the equation below.

$$TC = g \cdot 2^{64} \cdot 2^{128} = g \cdot 2^{192} \tag{6}$$

Here, 2^{64} corresponds to the 64-bit TRNG in garbling and 2^{128} reflects the brute-force resistance of SHA-256. g denotes the signature guessing complexity determined by its width. This exponential complexity makes brute force infeasible, while the TRNG's unique, non-deterministic outputs and SHA-256's irreversible transformations further bolster protection against replay attacks.

4.3 Fault Injection or Tampering

Tampering attacks use fault injection techniques—such as power glitches or clock disruptions—to bypass chiplet authentication, posing serious risks to SoC security. *SAFE-SiP* mitigates these threats by ensuring authentication fails when faults alter the process. As shown in Fig. 6, high Hamming Distance (HD) values indicate strong fault tolerance, with disruptions causing significant output variation. The garbled circuit uses TRNG labels (r, L) and signature (S) to produce output G; faults yield incorrect G, with HD values reaching 49.41%. SHA-256 amplifies sensitivity, where even single-bit changes in G cause hash mismatches with H_{expected}. Fig. 6 shows both fixed-signature and fixed-κ scenarios maintain high HD with increasing size. Secure boot protocols further enhance robustness, ensuring tampered chiplets are reauthenticated and disabled. This layered defense ensures integrity under tampering.

Table 3: Area Overhead Analysis of *SAFE-SiP*.

Design Name	Design Size (μm^2)	Sec. Param. (κ) (#)	Net Area Overhead (%)	Cell Count Overhead (%)	Cell Area Overhead (%)
Ariane	5263k	32	0.45	1.91	1.19
		64	0.73	3.01	1.91
CVA6	1271k	32	2.15	6.46	4.94
		64	3.59	10.13	7.92
OR1200	5472k	32	0.38	2.54	1.15
		64	0.64	3.99	1.84
NVDLA	1991k	32	1.29	5.34	3.15
		64	2.16	8.38	5.05
RISC-V	4815k	32	0.45	2.84	1.30
		64	0.76	4.45	2.09

4.4 Denial of Service Attacks

Denial attacks exploit the hierarchical structure of 2.5D and 3D chiplet-based designs, where one chiplet may block the authentication of another. In 3D integrations, this can involve a bottom chiplet denying the authentication of a top chiplet, compromising the entire stack. For 2.5D structures, the *SAFE-SiP* framework addresses this by independently authenticating each chiplet through parallel testing, using its garbled signature and TRNG-generated random outputs. This ensures system-wide reliability, as no single chiplet can obstruct authentication. In more interdependent 3D stacks, if there are any man-in-the-middle (MITM) attacks, the resulting output would be different, revealing potential intruders.

4.5 Bypass and Forging

The *SAFE-SiP* framework ensures vigorous protection against bypass and forging attacks, safeguarding chiplet integrity and authenticity. In bypass attacks, adversaries attempt to circumvent authentication to enable unauthorized chiplets. *SAFE-SiP* uses garbled circuits and SHA-256 hashing to tightly bind the chiplet's signature to its authentication output, flagging mismatches as unauthorized. Secure boot protocols further reinforce security by reauthenticating chiplets at every system boot. Forging attacks, where adversaries implant false watermarks or alter authentication, are mitigated by the randomness introduced during the garbling process, which obfuscates signature patterns and prevents replication.

5 Design Overhead Analysis

The following section provides an overview of *SAFE-SiP*'s practicality and analyses the area, timing and power overhead.

5.1 Experimental Setup

We implemented *SAFE-SiP* in Verilog and synthesized it using Synopsys Design Compiler using the SAED 14 nm standard cell library. Power, area, and timing were extracted from the post-synthesis netlist. Experiments ran on a dual-socket Intel Xeon system with 32 physical cores, 190 GB RAM, based on the Skylake microarchitecture. The design was verified for functionality and synthesized to meet timing at 100 MHz under typical PVT conditions.

Table 4: Timing Analysis for *SAFE-SiP*.

Sec. Param. (κ)	Critical Path Length (ns)	WNS (ns)	TNS (ns)	Violated Paths (#)	Latency (cc)
16	8.50	0.00	0.00	0	96
32	8.50	0.00	0.00	0	160
64	8.50	0.00	0.00	0	192

5.2 Area Overhead Analysis

The area overhead analysis of the *SAFE-SiP* framework, summarized in Table 3, underscores its efficient hardware utilization while maintaining high security. Across various designs, cell area overhead remains consistently low, reinforcing the framework's suitability for resource-constrained environments. For a security parameter of $\kappa = 32$, OR1200 and Ariane exhibit minimal overheads of 1.15% and 1.19%, respectively, with RISC-V following closely at 1.30%. NVDLA and CVA6, despite their increased complexity, maintain moderate overheads of 3.15% and 4.94%, respectively. At the higher security level ($\kappa = 64$), the highest recorded overhead remains within 7.92% (CVA6), while other designs, including OR1200 (1.84%) and RISC-V (2.09%), continue to demonstrate efficient area usage. These results highlight that *SAFE-SiP* achieves an optimal balance between hardware efficiency and cryptographic security, ensuring chiplet authentication without significant area penalties.

5.3 Timing Overhead Analysis

The *SAFE-SiP* framework ensures exceptional timing integrity across varying security parameters, as summarized in Table 4. For all evaluated configurations ($\kappa = 16, 32, 64$), the design consistently achieves a critical path length of 8.50 *ns* with no instances of Worst Negative Slack (WNS), Total Negative Slack (TNS), or timing violations. This demonstrates strict adherence to timing constraints, ensuring reliable operation at a reasonable scan clock frequency of 1 GHz. The absence of violations across all configurations establishes *SAFE-SiP* as a robust and predictable solution for secure computations. Additionally, the incurred authentication latency—ranging from 96 to 192 clock cycles depending on κ—is inherently parallelizable within different SiPs and among chiplets in a single SiP, allowing efficient distribution of authentication computations across multiple processing units. This flexibility enables seamless integration into high-performance and real-time systems, as parallel execution mitigates latency impact while maintaining security guarantees. The clean timing profile further reinforces *SAFE-SiP*'s adaptability to higher clock frequencies, making it a scalable and efficient solution for secure chiplet authentication in modern SoC architectures.

5.4 Power Overhead Analysis

The power overhead analysis of the *SAFE-SiP* framework, summarized in Table 5, highlights its scalability across different chiplet architectures and security parameters (κ). The results indicate that larger designs, such as NVDLA and OR1200, exhibit significantly lower relative power overhead compared to smaller designs like CVA6. For instance, NVDLA incurs only 2.37% and 2.11% overhead for $\kappa = 64$ and 32, respectively, while OR1200 maintains similarly low values of 4.12% and 3.67%. This trend underscores *SAFE-SiP*'s efficiency in leveraging higher baseline power in larger designs, minimizing the relative computational overhead. Conversely, in the smaller CVA6 design, the power overhead is more pronounced, reaching 34.08% and 30.34% for the same security parameters due to its lower baseline power, yet it remains within practical limits for resource-constrained deployments. RISC-V demonstrates a balanced efficiency, with overhead values of 7.43% and 6.61%, further reinforcing the framework's adaptability. These findings confirm that *SAFE-SiP* achieves better energy efficiency as chiplet size increases, making it an optimal choice for integrating secure authentication in high-performance architectures while maintaining reasonable overhead for smaller systems.

Table 5: Power Overhead Analysis for *SAFE-SiP*.

Design Name	Baseline Power (mW)	Sec. Param. (κ) (#)	Overhead (%)
Ariane	94.157	32	4.16
		64	4.67
CVA6	12.896	32	30.34
		64	34.08
OR1200	106.610	32	3.67
		64	4.12
NVDLA	185.140	32	2.11
		64	2.37
RISC-V	59.164	32	6.61
		64	7.43

6 Conclusion

This work presents *SAFE-SiP*, a scalable authentication framework for 2.5D/3D SiP assemblies using Multi-party Computation and garbled circuits. By garbling and hashing chiplet signatures, *SAFE-SiP* ensures confidentiality for vendors and detects tampering by foundries. It integrates with diverse signature schemes and secure boot processes without requiring dedicated security hardware. Security analysis confirms resilience to spoofing, tampering, and replay attacks using SHA-256 and garbling. Evaluations on five benchmark designs show minimal area, power, and latency overhead. The implementation is open-sourced to promote collaboration.

References

[1] 2022. IEEE Standard Testability Method for Embedded Core-based Integrated Circuits. *IEEE Std 1500-2022 (Revision of IEEE Std 1500-2005)* (2022), 1–168. doi:10.1109/IEEESTD.2022.9916221

[2] Bulbul Ahmed et al. 2024. SeeMLess: Security Evaluation of Logic Locking using Machine Learning oriented Estimation. In *Proceedings of the Great Lakes Symposium on VLSI 2024* (Clearwater, FL, USA) *(GLSVLSI '24)*. Association for Computing Machinery, New York, NY, USA, 489–494. doi:10.1145/3649476.3660382

[3] Yousra Alkabani, Farinaz Koushanfar, et al. 2007. Active Hardware Metering for Intellectual Property Protection and Security.. In *USENIX security symposium*, Vol. 20. 1–20.

[4] N. Nalla Anandakumar et al. 2022. Rethinking Watermark: Providing Proof of IP Ownership in Modern SoCs. *IACR Cryptol. ePrint Arch.* (2022).

[5] D. Beaver, S. Micali, and P. Rogaway. 1990. The round complexity of secure protocols. In *Proceedings of the Twenty-Second Annual ACM Symposium on Theory of Computing* (Baltimore, Maryland, USA) *(STOC '90)*. Association for Computing Machinery, New York, NY, USA, 503–513. doi:10.1145/100216.100287

[6] Mihir Bellare, Viet Tung Hoang, Sriram Keelveedhi, and Phillip Rogaway. 2013. Efficient Garbling from a Fixed-Key Blockcipher. In *2013 IEEE Symposium on Security and Privacy*. 478–492. doi:10.1109/SP.2013.39

[7] Assaf Ben-David, Noam Nisan, and Benny Pinkas. 2008. FairplayMP: a system for secure multi-party computation. In *Proceedings of the 15th ACM Conference on Computer and Communications Security* (Alexandria, Virginia, USA) *(CCS '08)*. Association for Computing Machinery, New York, NY, USA, 257–266. doi:10.1145/1455770.1455804

[8] Anshuman Chandra et al. 2023. A Case Study on IEEE 1838 Compliant Multi-Die 3DIC DFT Implementation. In *2023 IEEE International Test Conference (ITC)*. 11–20. doi:10.1109/ITC51656.2023.00011

[9] Ming-Fa Chen et al. 2019. System on Integrated Chips (SoIC(TM) for 3D Heterogeneous Integration. In *2019 IEEE 69th Electronic Components and Technology*

Conference (ECTC). 594–599. doi:10.1109/ECTC.2019.00095

[10] Nicolas T. Courtois et al. 2014. Optimizing SHA256 in Bitcoin Mining. In *Cryptography and Security Systems*, Zbigniew Kotulski, Bogdan Księżopolski, and Katarzyna Mazur (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 131–144.

[11] Aleksa Deric and Daniel Holcomb. 2022. Know time to die–integrity checking for zero trust chiplet-based systems using between-die delay PUFs. *IACR Transactions on Cryptographic Hardware and Embedded Systems* (2022), 391–412.

[12] Nidish Vashistha et al. 2022. ToSHI - Towards Secure Heterogeneous Integration: Security Risks, Threat Assessment, and Assurance. Cryptology ePrint Archive, Paper 2022/984. https://eprint.iacr.org/2022/984

[13] Tasnuva Farheen et al. 2023. A Twofold Clock and Voltage-Based Detection Method for Laser Logic State Imaging Attack. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 31, 1 (2023), 65–78. doi:10.1109/TVLSI.2022.3214724

[14] Niels Ferguson, Bruce Schneier, and Tadayoshi Kohno. 2010. *Cryptography Engineering: Design Principles and Practical Applications*. Wiley Publishing.

[15] Galib Ibne Haidar et al. [n. d.]. SECT-HI: Enabling Secure Testing for Heterogeneous Integration to Prevent SiP Counterfeits. ([n. d.]).

[16] Galib Ibne Haidar et al. 2024. GATE-SiP: Enabling Authenticated Encryption Testing in Systems-in-Package. In *Proceedings of the 61st ACM/IEEE Design Automation Conference* (San Francisco, CA, USA) *(DAC '24)*. Association for Computing Machinery, New York, NY, USA, Article 299, 6 pages. doi:10.1145/3649329.3656527

[17] Md Saad Ul Haque et al. 2023. SHI-Lock: Enabling Co-Obfuscation for Secure Heterogeneous Integration Against RE and Cloning. In *2023 IEEE Physical Assurance and Inspection of Electronics (PAINE)*. IEEE, 1–7.

[18] Carmit Hazay, Peter Scholl, and Eduardo Soria-Vazquez. 2020. Low Cost Constant Round MPC Combining BMR and Oblivious Transfer. *Journal of Cryptology* 33, 4 (01 Oct 2020), 1732–1786. doi:10.1007/s00145-020-09355-y

[19] Zahin Ibnat et al. 2023. ActiWate: Adaptive and Design-agnostic Active Watermarking for IP Ownership in Modern SoCs. In *2023 60th ACM/IEEE Design Automation Conference (DAC)*. 1–6. doi:10.1109/DAC56929.2023.10247688

[20] Maya Larson, Chunqiang Hu, Ruinian Li, Wei Li, and Xiuzhen Cheng. 2015. Secure Auctions without an Auctioneer via Verifiable Secret Sharing. In *Proceedings of the 2015 Workshop on Privacy-Aware Mobile Computing* (Hangzhou, China) *(PAMCO '15)*. Association for Computing Machinery, New York, NY, USA, 1–6. doi:10.1145/2757302.2757305

[21] Vasilis F. Pavlidis and Eby G. Friedman. 2008. *Three-dimensional Integrated Circuit Design*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

[22] D Rachmawati, J T Tarigan, and A B C Ginting. 2018. A comparative study of Message Digest 5(MD5) and SHA256 algorithm. *Journal of Physics: Conference Series* 978, 1 (mar 2018), 012116. doi:10.1088/1742-6596/978/1/012116

[23] M Tanjidur Rahman et al. 2020. Defense-in-depth: A recipe for logic locking to prevail. *Integration* 72 (2020), 39–57.

[24] Sazadur Rahman et al. 2023. Lle: Mitigating ic piracy and reverse engineering by last level edit. In *International Symposium for Testing and Failure Analysis*, Vol. 84741. ASM International, 360–369.

[25] M. Sadegh Riazi, Mojan Javaheripi, Siam U. Hussain, and Farinaz Koushanfar. 2019. MPCircuits: Optimized Circuit Generation for Secure Multi-Party Computation. In *2019 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*. 198–207. doi:10.1109/HST.2019.8740831

[26] IEEE SA. 2024. IEEE Standard for Access and Control of Instrumentation Embedded within a Semiconductor Device. https://standards.ieee.org/ieee/1687/3931/ Accessed: November 2024.

[27] IEEE SA. 2024. IEEE Standard for Test Access Architecture for Three-Dimensional Stacked Integrated Circuits. https://standards.ieee.org/ieee/1838/5073/ Accessed: November 2024.

[28] IEEE SA. 2024. IEEE Standard Testability Method for Embedded Core-based Integrated Circuits. https://standards.ieee.org/ieee/1500/7704/ Accessed: November 2024.

[29] Yousef Safari, Pooya Aghanoury, Subramanian S. Iyer, Nader Sehatbakhsh, and Boris Vaisband. 2023. Hybrid Obfuscation of Chiplet-Based Systems. In *2023 60th ACM/IEEE Design Automation Conference (DAC)*. 1–6. doi:10.1109/DAC56929.2023.10247738

[30] Farhana Sheikh et al. 2021. 2.5D and 3D Heterogeneous Integration: Emerging applications. *IEEE Solid-State Circuits Magazine* 13, 4 (2021), 77–87. doi:10.1109/MSSC.2021.3111386

[31] Ebrahim M. Songhori, Siam U. Hussain, Ahmad-Reza Sadeghi, Thomas Schneider, and Farinaz Koushanfar. 2015. TinyGarble: Highly Compressed and Scalable Sequential Garbled Circuits. In *2015 IEEE Symposium on Security and Privacy*. 411–428. doi:10.1109/SP.2015.32

[32] Ebadollah Taheri et al. 2024. SCRIPT: A Multi-Objective Routing Framework for Securing Chiplet Systems against Distributed DoS Attacks. In *Proceedings of the Great Lakes Symposium on VLSI 2024* (Clearwater, FL, USA) *(GLSVLSI '24)*. Association for Computing Machinery, New York, NY, USA, 78–85. doi:10.1145/3649476.3658763

[33] Md Sami Ul Islam Sami et al. 2024. PQC-HI: PQC-enabled Chiplet Authentication and Key Exchange in Heterogeneous Integration. In *2024 IEEE 74th Electronic Components and Technology Conference (ECTC)*. 464–471. doi:10.1109/ECTC51529.2024.00079

[34] Taha Selim Ustun, Shaik Mullapathi Farooq, and S. M. Suhail Hussain. 2019. A Novel Approach for Mitigation of Replay and Masquerade Attacks in Smartgrids Using IEC 61850 Standard. *IEEE Access* 7 (2019), 156044–156053. doi:10.1109/ACCESS.2019.2948117

[35] Andrew Chi-Chih Yao. 1986. How to generate and exchange secrets. In *27th Annual Symposium on Foundations of Computer Science (sfcs 1986)*. 162–167. doi:10.1109/SFCS.1986.25