On a Static Analysis Methodology for Confidentiality and Security Signoff of RISC-V Crypto Core

[Author names removed for blind review]

Abstract

The exponential growth of cloud computing, IoT, and smart devices has led to a significant increase in cyber-attacks, many of which exploit vulnerabilities in hardware security. Architectural and RTL design optimizations that were previously deemed clever often unintentionally expose hardware to unforeseen security risks. Traditional countermeasures such as simulation are constrained by the limited scope of explicit stimuli and the creativity of test cases, while formal verification faces scalability challenges, especially with large designs.

This situation highlights an opportunity for the application of static analysis techniques, which algorithmically analyze design properties without relying on input stimuli or Boolean evaluations. Static analysis offers an innovative approach to security verification by detecting failure-specific behaviors, ensuring valid secure data paths, and identifying illegal data leaks and interference paths.

In this paper, we present a transformative approach to hardware security verification, leveraging advanced static analysis methods to address key vulnerabilities. Experimental results are demonstrated on the OpenTitan Earl Grey chip, showcasing the detection and verification of secure data paths and the identification of potential information leaks under various scenarios. This work establishes a robust framework for early-stage security validation, addressing critical hardware risks efficiently and at scale.

Introduction

The complexity and sophistication of today's hardware security threats have escalated significantly, driven by advanced cyberattacks that exploit vulnerabilities at both hardware and software layers [1]. Safeguarding confidentiality and security is essential to protect sensitive information from unauthorized access and leakage. Critical speculative execution vulnerabilities, such as Spectre [2] and Meltdown [3], exacerbate these risks by using mispredicted data paths to expose secure information to unprotected areas.

RISC-V, being open source, flexible, and highly customizable, presents unique advantages but also introduces susceptibility to security vulnerabilities. The typical security mitigation cycle begins with public identification of vulnerabilities, followed by analysis and patching—primarily at the software level initially, and eventually incorporating hardware countermeasures over time.

This paper proposes a static analysis-based verification methodology to address hardware security path vulnerabilities at the RTL stage. The approach enables early detection with minimal user intervention and scalability to system-on-chip (SoC) level flows. We specifically analyze security challenges associated with the OpenTitan Earl Grey chip at the RTL level, presenting our findings and a solution using a state-of-the-art tool for static security signoff.

Static Signoff for Security Verification

Static verification [4] is a new paradigm in the context of security verification, which, until now, has typically been addressed via simulation, which lacks coverage guarantees, or via formal verification, which lacks capacity to work at SoC levels.

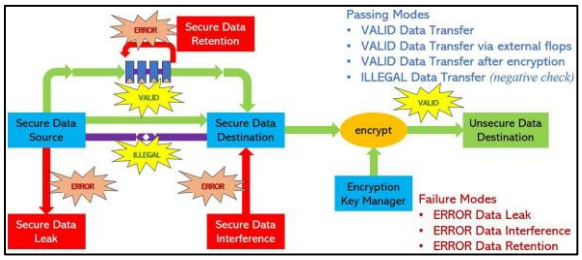

Figure 1: Pass/fail error modes in static security verification.

We present SS[1], a new commercial static security verification tool in this paper. Figure 1 provides an overview of passing and failure modes in our static security verification tool. We model the problem as an information flow problem and solve it via path verification techniques. Passing modes can be formulated as a positive check, verifying valid data transfers in different contexts, or as a negative check, verifying that certain data transfers are illegal. The ability to formulate a negative check at SoC level is a key contribution of our methodology. Failure modes are of the nature of secure data leaks, unauthorized data interfering with secure data transfers, and secure data retention in registers outside of the secure transaction timeframe.

Static Security Verification in OpenTitan Earl Grey Chip

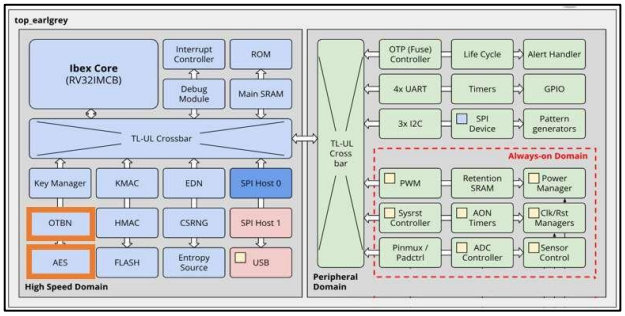

Figure 2: Block diagram of the OpenTitan Earl Grey chip.

The OpenTitan Earl Grey chip [5] is a low-power secure microcontroller that is designed for several use cases requiring hardware security. As shown in Figure 2, the system is split into a fast processor core and a slower clocked peripheral domain. We specifically focus on OTBN, the OpenTitan Big Number co-processor, and AES encryption blocks as part of the experiments in this paper (marked in Figure 2).

Illegal Data Leak After a Misprediction

OpenTitan Big Number (OTBN) co-processor is specialized for the execution of security-sensitive asymmetric (public-key) cryptography code, such as RSA or ECC. It is dominated by wide integer arithmetic, with 256b wide data-path, registers, and instructions, as well as a dedicated 32b wide control-path, with built-in access to random numbers.

Protected Secure Data Transfer in Multi- Core Systems

In our final experiment with OpenTitan, we created a 4-core Earl Grey chips communicating via an AXI interface (as shown in Figure 5). In this case the valid transactions are that each core *n* can only communicate with its own key manager, and all communication transactions with the key managers of any of the other 3 cores are illegal. We modeled this as 1 valid transaction check and 3 illegal transaction checks per core, for a total of 16 checks in our system. We were able to statically verify that these transactions hold under all conditions post reset.

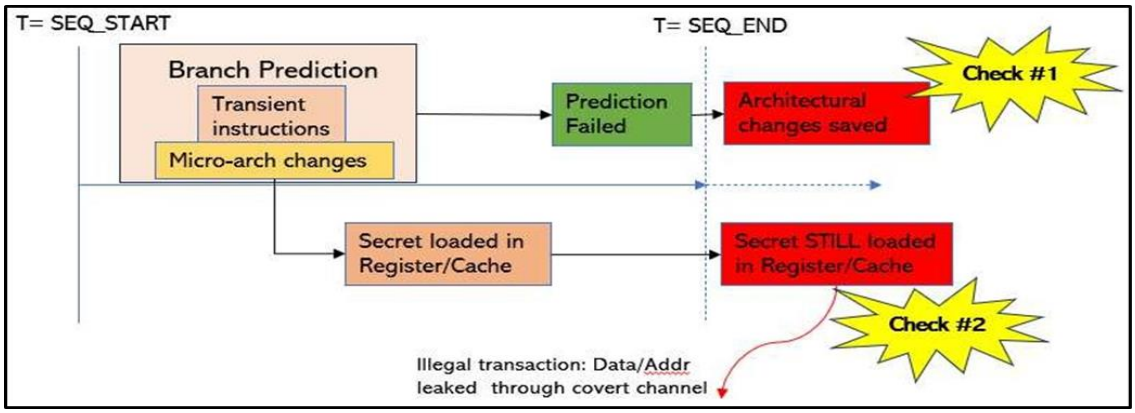

Figure 3: Mispredicted branch sequence in OTBN can be verified by checking for registers restoring fetch sequence control, as well as for illegal secure data leaks.

In this case, we modified the RTL to trigger a misprediction in the form of a fetch fail. This triggers a Spectre-like vulnerability within the RISC-V architecture. Under this situation, the security verification required is two-fold: one, architectural changes are saved, and post-misprediction, data in secure assets are modified to allow fetching next instruction(s); and two, any secure data that got fetched during the mispredict phase, should not leak to unsecure areas of the chip. We model both these checks (as shown in Figure 3) as commands and statically verify them.

Our results show that prefetch address registers leaking to unsecure areas is illegal, effectively catching sensitive information leaks in RISC-V implementations.

Confidentiality Preservation in AES

In this experiment, we used OpenTitan's AES hardware IP block within the RISC-V framework, to verify the correctness of encrypting confidential signals. The signals in the block were classified into three domains, *viz.* Confidential, Exempt, and Exposed. Confidential signals hold secure data, Exempt signals hold post-encryption data, and Exposed signals are world readable. Valid and Illegal data transactions are then defined as show in Figure 4. Within our tool we support the addition of clear data transfer trigger conditions, using which we were able to statically verify all the above situations, modeled as 8 separate checks in our system.

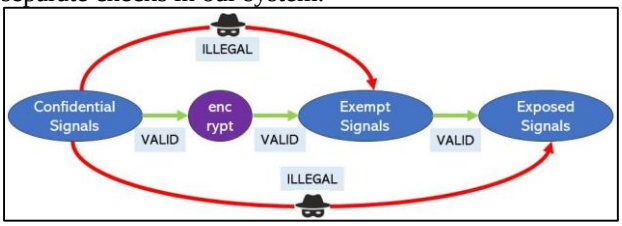

Figure 4: AES signal classification and their interactions.

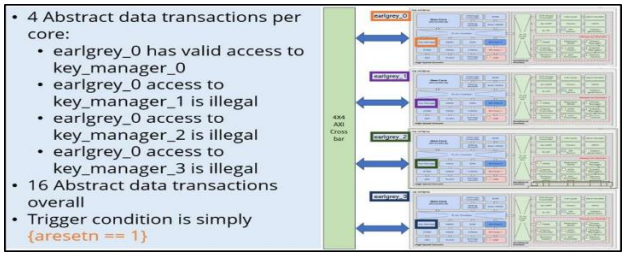

Figure 5: Multi-Core Earl Grey chips connected via AXI.

Summary and Conclusions

Table 1 provides an overview of the security verification results conducted using SS, our advanced static analysis framework. A notable strength of our modeling approach lies in its capability to identify negative scenarios (e.g., illegal data transfers) by providing explicit examples of such occurrences under specific signal conditions over time. This enables comprehensive insight into failure points. Additionally, the framework includes a state-of-the-art, intuitive, and interactive debugging environment that facilitates precise diagnosis and resolution of root causes contributing to security violations.to precisely diagnose and fix the issues leading to failure.

Table 1: OpenTitan Security Verification Results.

Design	GateCount	#Checks	Time	Mem
OTBN	352K	2	<1m	1M
AES	118K	8	<1m	568K
4-Core	13.3M	16	45m	58G

The experiments clearly demonstrate the efficiency and scalability of the tool, even for designs at the SoC level. By combining security-path-specific static analysis techniques with targeted formal verification strategies, this approach delivers signoff-level verification of critical security paths, ensuring robust and reliable design validation.

References

[1] Common Weakness Enumeration (CWE), *MITRE*, 2014.
[2] P. Kocher, J. Horn, A. Fogh, D. Genkin, D. Gruss, W. Haas, M. Hamburg, M. Lipp, S. Mangard, T. Prescher, M. Schwarz, and Y. Yarom, "Spectre attacks: Exploiting speculative execution," Communications of the ACM, vol. 62, no. 7, pp. 93–101, 2019.
[3] C. Canella, M. Schwarz, M. Lipp, and D. Gruss, "Meltdown-rw: Exploiting speculative write loads for data leakage," in Proceedings of the 28th USENIX Security Symposium (USENIX Security 19), 2019.
[4] P. Ashar and V. Viswanath, "Closing the Verification Gap with Static Sign-off," 20th International Symposium on Quality Electronic Design (ISQED), Santa Clara, CA, USA, 2019.
[5] OpenTitan Earl Grey chip datasheet. https://opentitan.org/book/hw/top_earlgrey/doc/specification.html