TEEcorrelate

An Information-Preserving Defense against Performance-Counter Attacks on TEEs

Hannes Weissteiner[1] Fabian Rauscher[1] Robin Leander Schröder[3, 4] Jonas Juffinger[1]
Stefan Gast[1] Jan Wichelmann[2] Thomas Eisenbarth[2] Daniel Gruss[1]

TU Graz
SCIENCE
PASSION
TECHNOLOGY

ARTIFACT EVALUATED usenix ASSOCIATION AVAILABLE
ARTIFACT EVALUATED usenix ASSOCIATION FUNCTIONAL
ARTIFACT EVALUATED usenix ASSOCIATION REPRODUCED

Motivation [1, 5]

CounterSEVeillance:
Performance-Counter Attacks on AMD SEV-SNP

Stefan Gast*, Hannes Weissteiner*, Robin Leander Schröder†‡ and Daniel Gruss*
*Graz University of Technology, Austria
†Fraunhofer SIT, Darmstadt, Germany
‡Fraunhofer Austria, Vienna, Austria

Abstract—Confidential virtual machines (VMs) promise higher security by running the VM inside a trusted execution environment (TEE). Recent AMD server processors support confidential VMs with the SEV-SNP processor extension. SEV-SNP provides guarantees for integrity and confidentiality for confidential VMs despite running them in a shared hosting environment.

In this paper, we introduce CounterSEVeillance, a new side-channel attack leaking secret-dependent control flow and operand properties from performance counter data. Our attack is the first to exploit performance counter side-channel leakage with single-instruction resolution from SEV-SNP VMs and works on fully patched systems. We systematically analyze performance counter events in SEV-SNP VMs and find that 228 are exposed to a potentially malicious hypervisor. CounterSEVeillance builds on this analysis and records performance counter traces with an instruction-level resolution by single-stepping the victim VM using APIC interrupts in combination with page faults. We match CounterSEVeillance traces against binaries, precisely recovering the outcome of any secret-dependent conditional branch and inferring operand properties. We present four attack case studies, in which we exemplarily showcase concrete exploitable leakage with 6 of the exposed performance counters. First, we use CounterSEVeillance to extract a full RSA-4096 key from a single MbedTLS signature process in less than 8 minutes. Second, we present the first side-channel attack on TOTP verification running in an AMD SEV-SNP VM, recovering a 6-digit TOTP with only 31.1 guesses on average. Third, we show that CounterSEVeillance can leak the secret key from which the TOTPs are derived from the underlying base32 decoder. Fourth and finally, we show that CounterSEVeillance can also be used to construct a plaintext-checking oracle in a divide-and-surrender-style attack. We conclude that moving an entire VM into a setting with a privileged adversary increases the attack surface, given the vast amounts of code not vetted for this specific security setting.

Virtualization (SEV) [11], Intel Trust Domain Extensions (TDX) [45], or ARM Confidential Compute Architecture (CCA) [13] promise higher security by running the VM inside a trusted execution environment (TEE), designed to protect sensitive data from physical and privileged software access. In contrast to other TEEs, like Intel Software Guard Extensions (SGX) [46], or ARM TrustZone [14], which are designed to run only specialized, hardened code, AMD SEV and Intel TDX are designed to run an entire software stack supplied by the customer, including a fully-featured, general-purpose operating system. Customer software stacks tend to be complex and often consist of components that are not designed with the stronger attacker model in mind, where the physical host is compromised.

Numerous works [16], [77], [84], [91], [85], [80], [5], [88], [52], [55], [92], [98], [68], [69], [97], [95], [56] show attacks on TEEs. In particular, multiple works focus on side-channel attacks, leaking critical information from the processor's memory interface [91], processor caches [66], [78], [35], [17], [94], branch predictors [49], [43], memory disambiguation [65], and power consumption information [58]. Several of these attacks exploit that the threat model of trusted execution environments allows for much more powerful adversaries than a regular attack in a native or virtualized attack setting: For instance, Xu et al. [99] control the steps of an enclave by controlling the page mappings, effectively leading to fine-granular stepping through the victim program, comparable to attaching a debugger. Moghimi et al. [65] slow down the victim dramatically and amplify the leakage consequently. Van

Robust and Scalable Process Isolation against Spectre in the Cloud

Martin Schwarzl[1], Pietro Borrello[2], Andreas Kogler[1], Kenton Varda[3], Thomas Schuster[1], Michael Schwarz[4], and Daniel Gruss[1]

[1]Graz University of Technology, Austria [2]Sapienza University of Rome, Italy
[3]Cloudflare Inc. [4]CISPA Helmholtz Center for Information Security, Germany

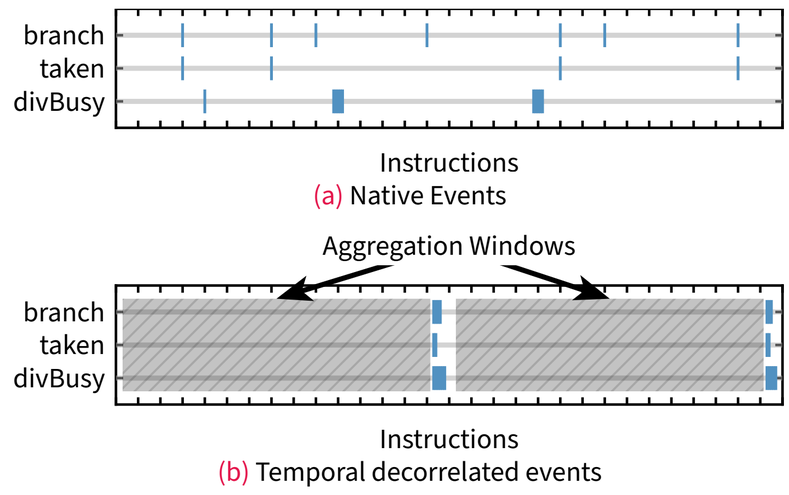
Abstract. In the quest for efficiency and performance, edge-computing providers replace process isolation with sandboxes, to support a high number of tenants per machine. While secure against software vulnerabilities, microarchitectural attacks can bypass these sandboxes.
In this paper, we present a Spectre attack leaking secrets from co-located tenants in edge computing. Our remote Spectre attack, using amplification techniques and a remote timing server, leaks 2 bit/min. This motivates our main contribution, *DyPrIs*, a scalable process-isolation mechanism that only isolates suspicious worker scripts following a lightweight detection mechanism. In the worst case, DyPrIs boils down to process isolation. Our proof-of-concept implementation augments real-world cloud infrastructure used in production at large scale, *Cloudflare Workers*. With a false-positive rate of only 0.61 %, we demonstrate that DyPrIs outperforms strict process isolation while statistically maintaining its security guarantees, fully mitigating cross-tenant Spectre attacks.

Performance counters are valuable tools for benign hypervisors to optimize resource utilization and detect malicious behavior. Malware inside enclaves is especially hard to detect [4, 3, 2] and affects cloud security. However, unmitigated performance counters enable fine-grained, instruction-level information leakage from confidential virtual machines (CVM) by a malicious hypervisor.

Overview

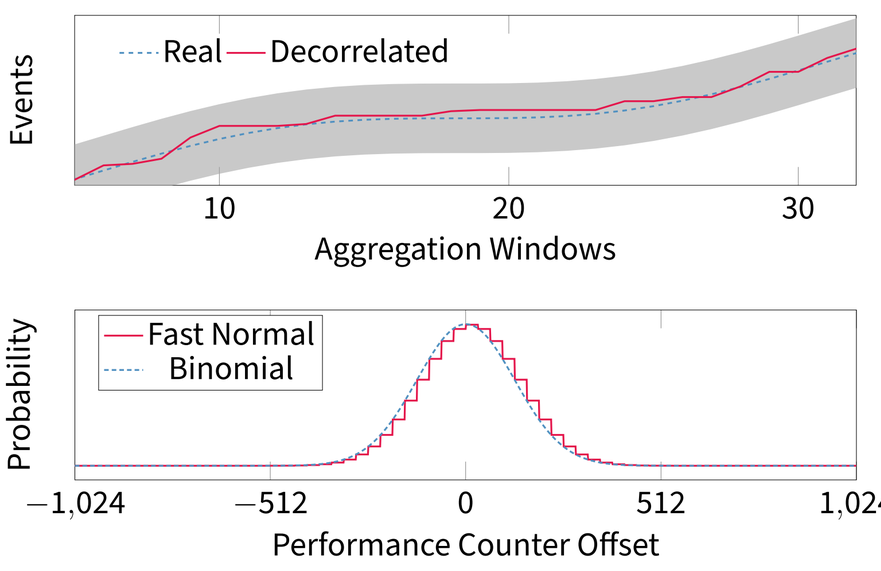
TEEcorrelate mitigates fine-grained performance-counter leakage by decorrelating real and reported performance counter values. The mitigation consists of two main components:
Temporal decorrelation aggregates performance counter values and only exposes them after a certain number of instructions, hiding exact increment timings.
Value Decorrelation allows reported performance counter values to deviate from real values, hiding, the exact number of increments.

Temporal Decorrelation

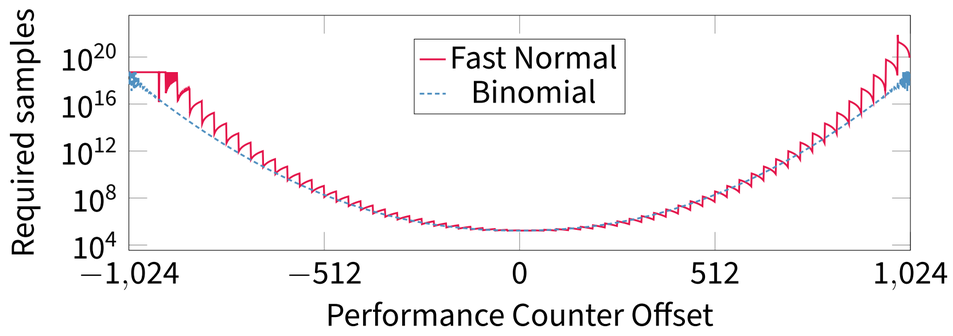


(a) Native Events
(b) Temporal decorrelated events

Value Decorrelation



Results



The required number of traces to recover 1 bit of information, depending on the current deviation between real and reported performance counter values.

Case study [1]	No mitigation	TEEcorrelate	Overhead
String Comparison	18.14 s	34.7 days	× 160 thousand
Lookup Table	0.58 s	285.4 days	× 40 million
RSA	7.15 min	824.6 days	× 160 thousand
HQC[1]	6.13 min	27.0 days	× 6 thousand

Numbers based on our recommended deviation window size of 2048.
Estimated runtime overhead of TEEcorrelate is approximately 0.09%.

Conclusion

- TEEcorrelate mitigates known performance-counter attacks on TEEs [1].
- It imposes a minimal runtime overhead of 0.09%.
- Increases number of required traces to infeasible levels.
- Applicable to SEV-SNP, TDX, CCA, and RISC-V CVMs.
- Enables secure use of performance counters in cloud environments.

Contact

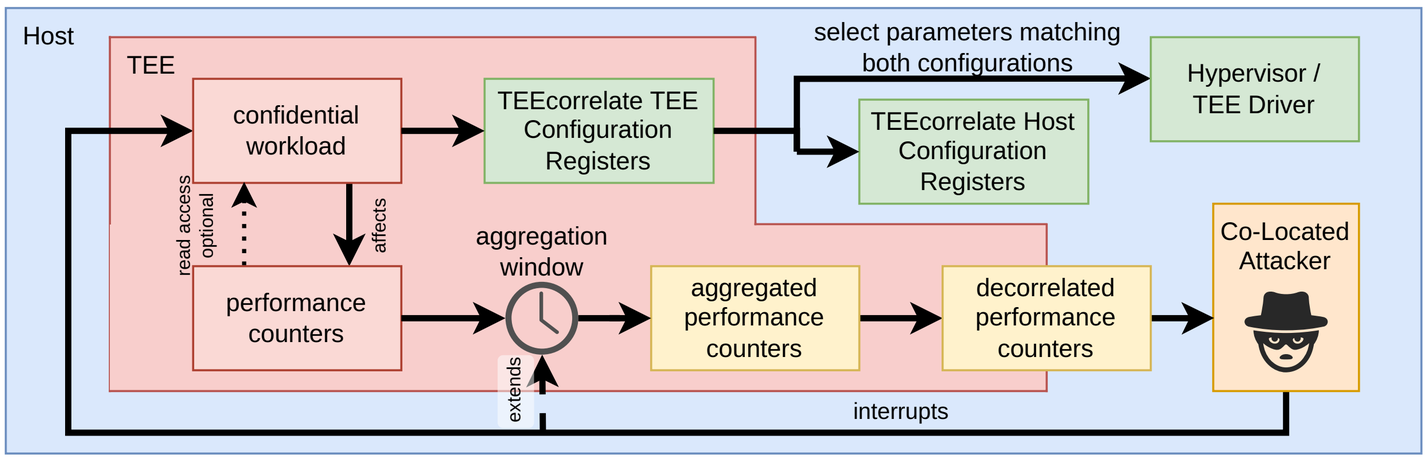
hannes.weissteiner@tugraz.at
hweissi@infosec.exchange
hannesweissteiner.com

Bibliography

[1] Stefan Gast et al. **CounterSEVeillance: Performance-Counter Attacks on AMD SEV-SNP.** NDSS. 2025.
[2] Daniel Gruss et al. **Another Flip in the Wall of Rowhammer Defenses.** S&P. 2018.
[3] Yeongjin Jang et al. **SGX-Bomb: Locking Down the Processor via Rowhammer Attack.** SysTEX. 2017.
[4] Michael Schwarz et al. **Malware Guard Extension: Using SGX to Conceal Cache Attacks.** DIMVA. 2017.
[5] Martin Schwarzl et al. **Dynamic Process Isolation.** ESORICS. 2021.

Acknowledgements

This research is supported in part by the European Research Council (ERC project FSSec 101076409), the Austrian Science Fund (FWF SFB project SPyCoDe 10.55776/F85), Deutsche Forschungsgemeinschaft (project ReTEE) and the National Research Center for Applied Cybersecurity ATHENE as part of the PORTUNUS project in the research area Crypto. Additional funding was provided by generous gifts from Red Hat, Google, and Intel.



[1]Graz University of Technology [2]University Luebeck [3]Fraunhofer SIT, Darmstadt, Germany [4]Fraunhofer Austria, Vienna
USENIX 2025 | hannes.weissteiner@tugraz.at
ISEC | Institute of Information Security
› isec.tugraz.at