PowSpectre: Powering Up Speculation Attacks with TSX-based Replay

Md Hafizul Islam Chowdhuryy
University of Central Florida
Orlando, Florida, USA

Zhenkai Zhang
Clemson University Clemson
South Carolina, USA

Fan Yao
University of Central Florida
Orlando, Florida, USA

Computer Architecture and Systems Research Lab (CASRL)
University of Central Florida

19th ACM ASIA Conference on Computer and Communications Security
(**ASIACCS 2024**)
July 1st – July 5th, 2024

ACM ASIACCS 2024

Hardware-based Side Channel Problems

- Hardware-based side channel leak information in *software* through monitoring *hardware resource*.
  - *Hardware-maintained states (persistent)*: uArch states (e.g. caches, TLBs, branch predictors).
  - ***Implicit & ephemeral states (transient)***: uArch contention, physical properties (e.g., EM, power).
- TEE threat model (aka, malicious OS) enables *privileged adversaries*.
  - Enable fine-grained victim execution control and **instruction replay**.
  - Particularly useful for transient side channels (replay enhances and denoises observations) .

The practicality of ***replaying speculative execution attacks*** is not well explored

Are existing replay techniques sufficient for speculation attacks exploiting coarse grained transient channels (i.e., Power)?

UCF

This work

Investigation of *power side channel during speculation* through exploiting novel *hardware transactional memory based instruction replay*.

Software-based Power Monitoring

- Intel and AMD systems provide *software interface for power measurements*.
  - *Running average power limit* (**RAPL**).
- Required for core functionality such as: *thermal management*, *turbo boost*.

- Prior works exploit RAPL as *non-speculative side channel*.
  - Platypus [S&P'21], Red Alert [ASIACCS'21], Collide+Power [USENIX'23].
- **Intel's mitigation**: Downgrade the power reporting to a model-based approach.
  - Claims to defeat state-of-the-art RAPL side channels.

Hardware Transactional Memory

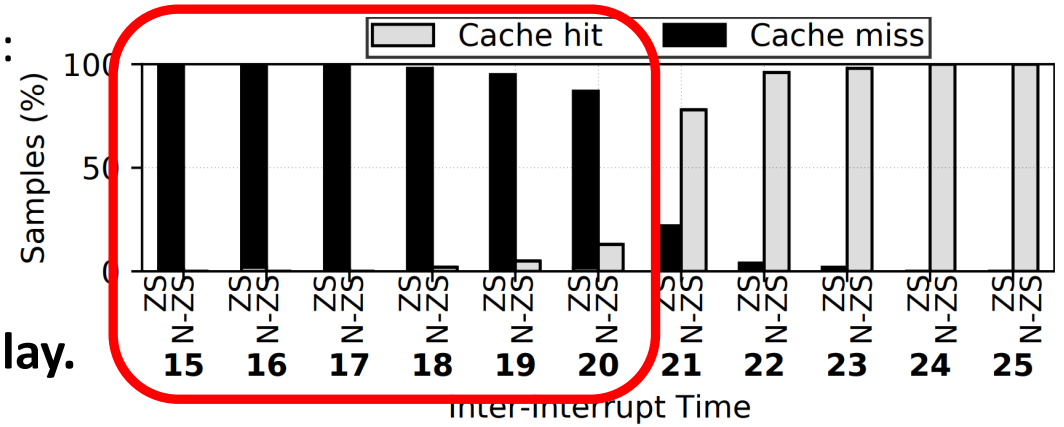
- High-performance alternative to explicit locks (i.e., MUETX) for concurrent memory accesses.
- Transactional memory primitives:
  - Executes group of memory operations as *one transactions*.
  - Transaction is completed when *all memory operations* are completed.
  - If *any one of the memory operations* fails, transaction is aborted (and potentially retried).
- **Intel TSX:** Intel's implementation of hardware transactional memory.
  - Implemented through extension of *cache coherence*.
  - Tracks transactionally accessed blocks (***readset***) and modified blocks (***writeset***) in hardware.

- **A class of defense against SGX page-fault based attacks utilizes Intel TSX**

Existing Replay Mechanisms: Timer Interrupt-based

Microbenchmark to evaluate timer interrupt-based replay:

- MOVQ scheduled to execute after ERESUME.
- Cache-hit → MOVQ has executed.
- **Cache-hit + ZS (Zero-stepping) → Valid instruction replay.**



Observation: Consistent cache miss in all successful Zero-stepping samples

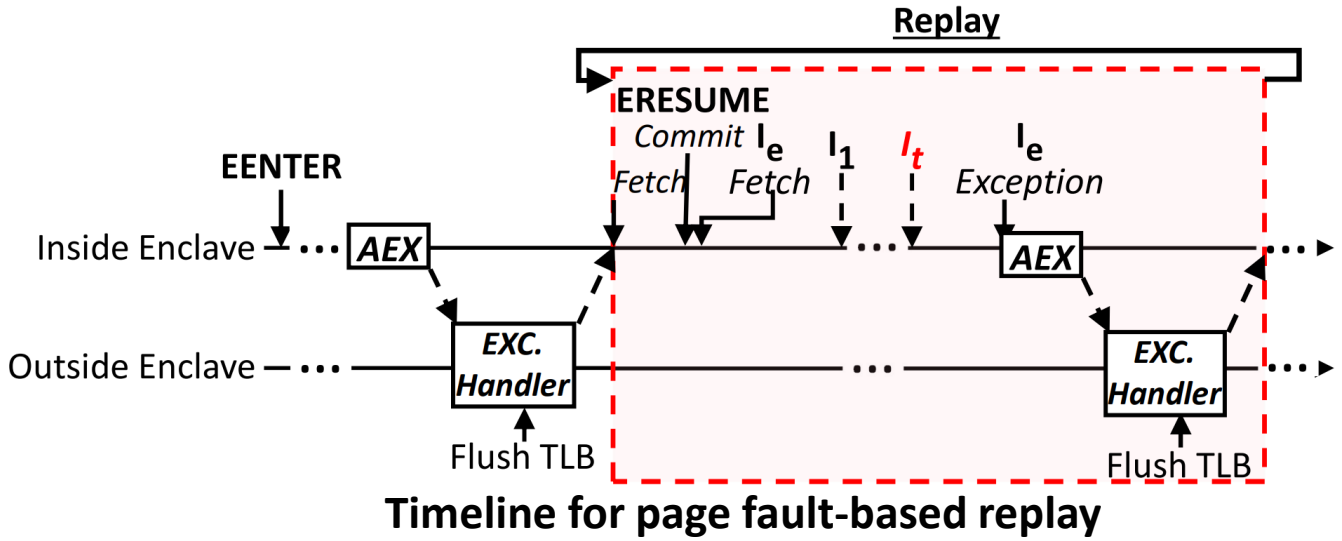
Instructions after ERESUME do not execute until it is retired.

Zero-stepping serves as an *execution rewind (i.e., perform ERESUME restoration)* and ***cannot replay instructions***.

UCF

Existing Replay Mechanisms: Page Faults-based

Privileged attacker can deliberately set page faults to induce replay.



Timeline for page fault-based replay

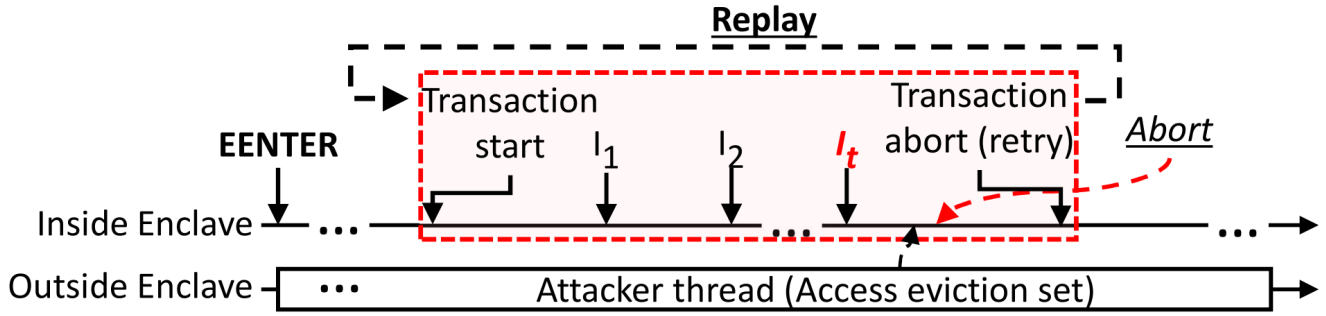
Page faults trigger TLB misses and page table walk → ***Non-deterministic noise*** *to an elastic timing channel.*

Trigger the page fault handler (potential partial page table walks) → ***Large system-level footprint***.

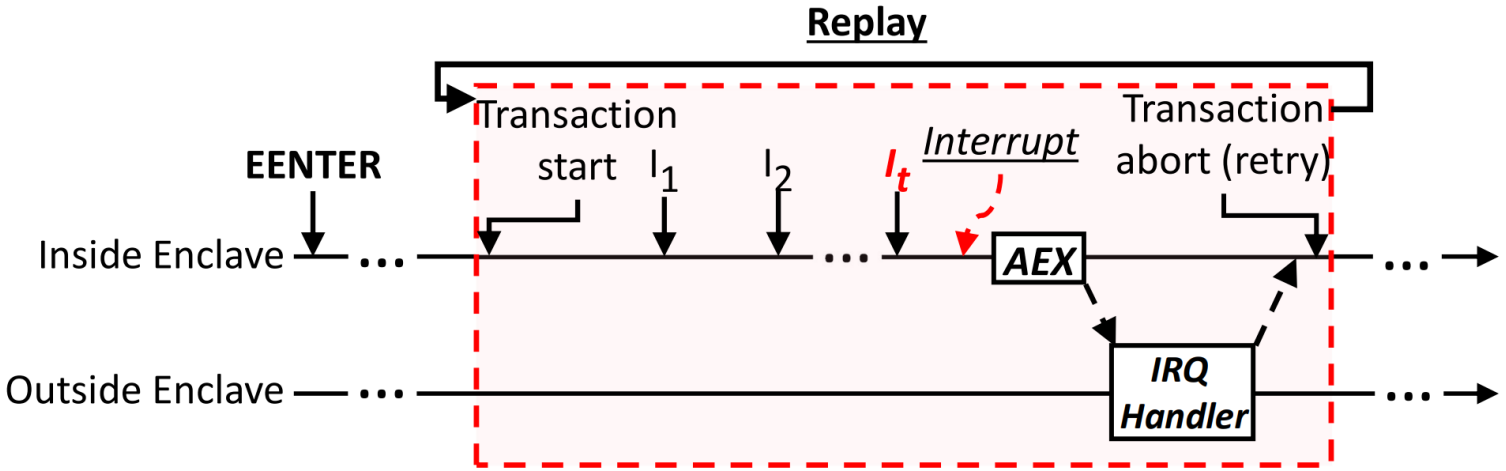
UCF

TMPlayer: TSX For Instruction Replay

TMPlayer-E: Perform readset/writeset *eviction* before transaction commits (results in TSX abort).

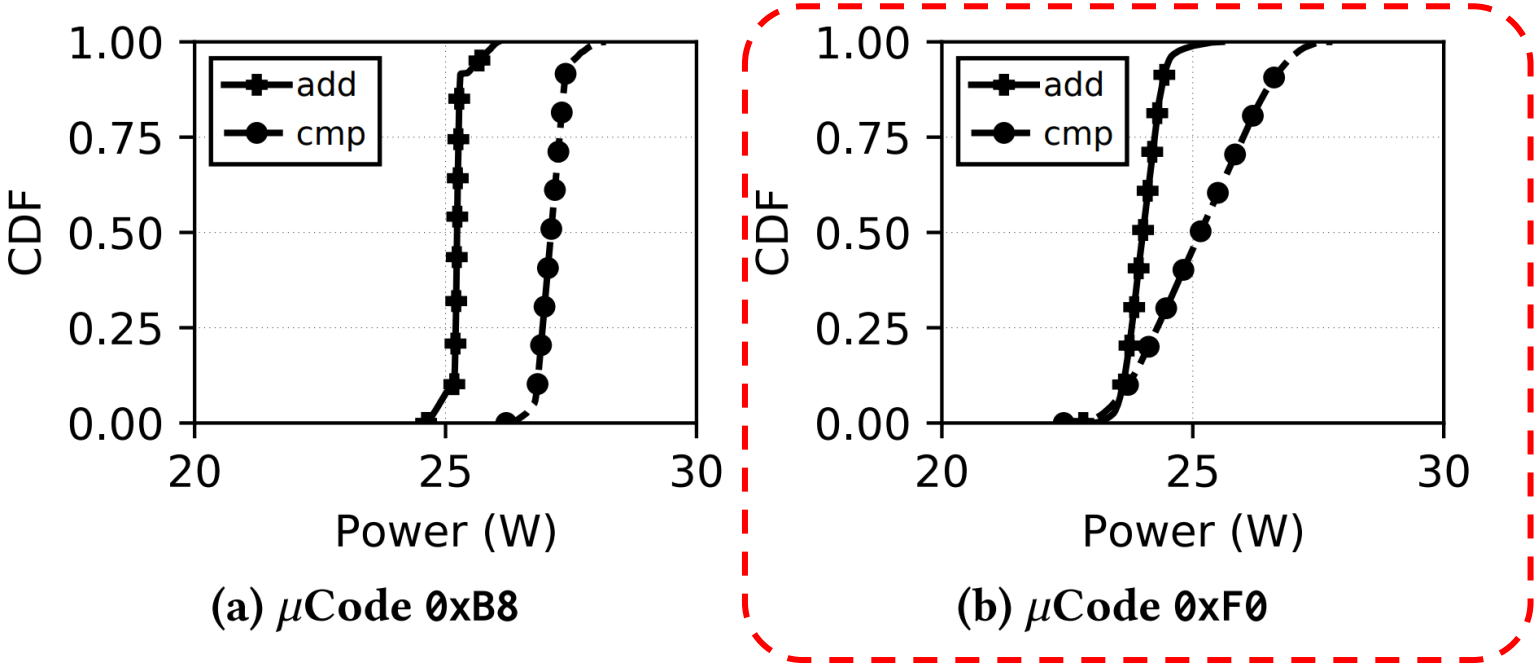


TMPlayer-I: Perform *interrupt* before transaction commits (results in TSX abort).



UCF

Power Side Channel in Speculative Execution



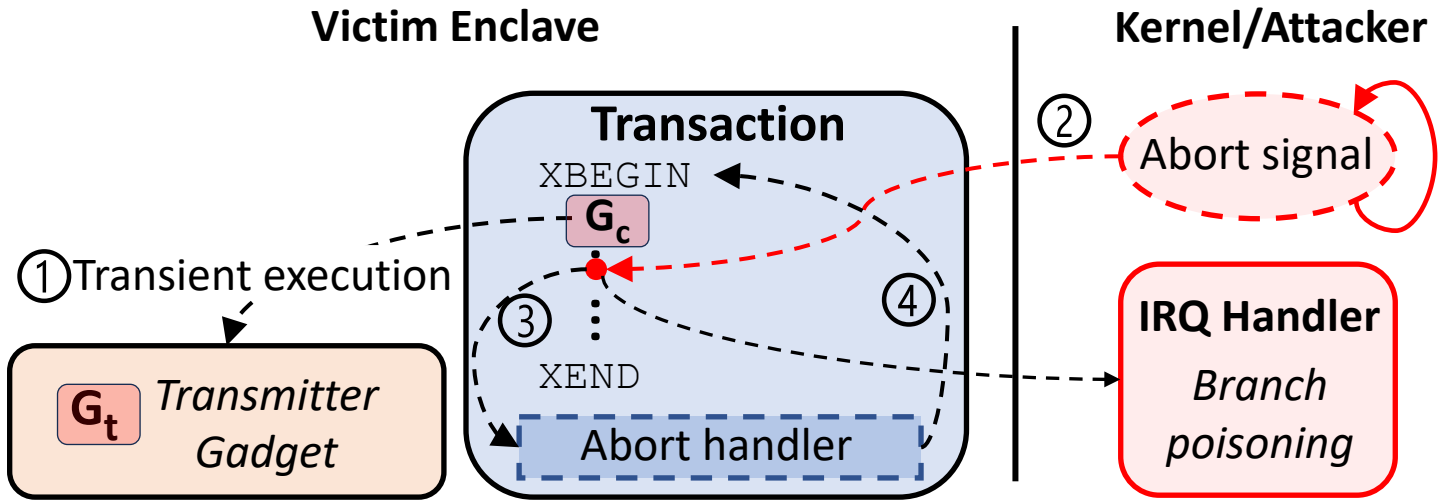
CDF of power consumption for two different execution paths

Latest Intel μcode patch against RAPL attacks

Even with ***model-based power reporting***, an adversary can still ***observe different power signatures*** for different instructions

UCF

Overview of PowSpectre Attack



① Identify *power differentiable* ***transmitter gadget (G_t)***

- ***G_t*** is outside of TSX transaction – executed speculatively through branch poisoning from TSX.

② Setup a *TSX abort trigger* executed before the end of transaction.

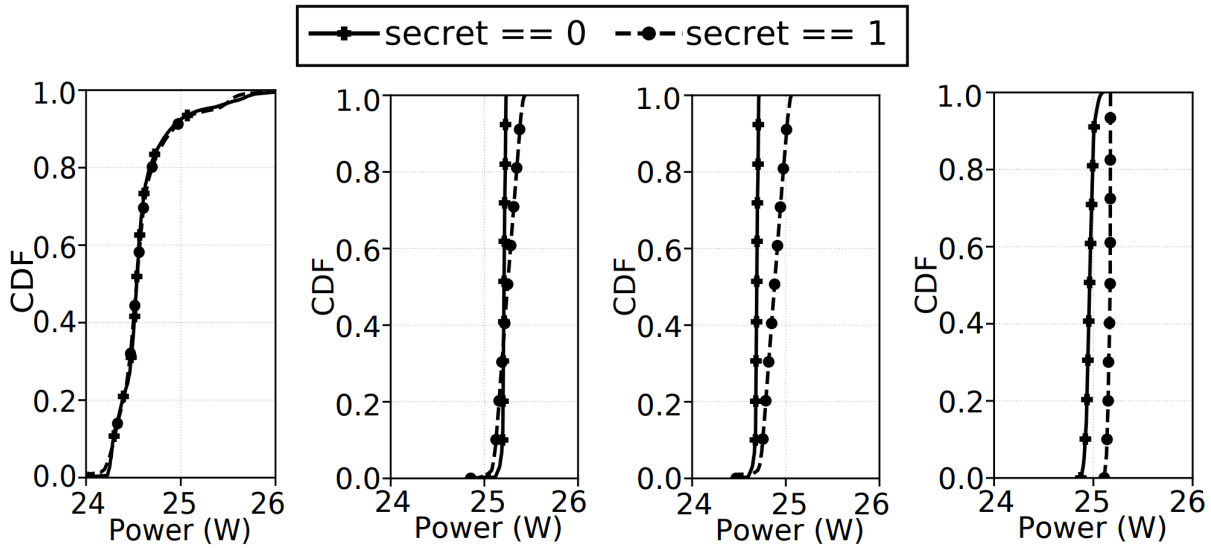
- **TMPlayer-E**: A memory block executed after the poisoned branch execution.
- **TMPlayer-I**: An APIC interrupt timer (periodic) arriving before transaction commits.

③ Once abort trigger is executed, the transaction is terminated, and the abort handler is executed.

④ The transaction is then re-executed from the abort handler.

UCF

Evaluation of TMPlayer



(b) *APIC Interrupt* (c) *Page Fault* (d) *TMPlayer-E* (e) *TMPlayer-I*

Accuracy of the control-flow detection under different Instruction Replay primitive

- APIC interrupt-based instruction replay exhibits almost no power differentiability.
- Page fault-based instruction replay demonstrates some differentiability (63% accuracy).

- TMPlayer variants demonstrate extremely high differentiability (**95%+ accuracy**)

UCF

Case Studies with PowSpectre

Platform: Intel Core i7 9700K
Target application: Intel SGX SSL (based on OpenSSL v1.1.0o)

```
6c590 <EC_KEY_METHOD_free>:
6c594: test [rdi+0x8], 0x1
6c598: jne 6c5a0
6c59a: ret                          ❶
6c59b: nop [rax+rax*1+0x0]
6c5a0: mov edx, 0xaa                ❷
6c5a5: lea rsi, [rip+...]
6c5ac: jmp 8ae50
...
8ae59: endbr64
8ae54: mov rax, [rip+...]
```

Case study-1: Gadget in elliptic curve key management

```
d6910 <asn1_enc_init>:
...
d692a: test [rdx+0x8], 0x2
d692e: je d694d
d6930: movsxd rdx, [...]            ❶
d6934: add rax, rdx
d6937: mov [rax], 0x0
d693e: mov [rax+0x8], 0x0
d6945:
d6946: mov [rax+0x10], 0x1
d694d: ret                          ❷
```

Case study-2: Gadget in abstract syntax encoding of keys

Attack	TMPlayer-Type	0xB8	0xF0
Case study 1	*TMPlayer-E*	93.2%	81.4%
	TMPlayer-I	92.7%	82.4%
Case study 2	*TMPlayer-E*	92.7%	83.6%
	TMPlayer-I	94.3%	82.4%

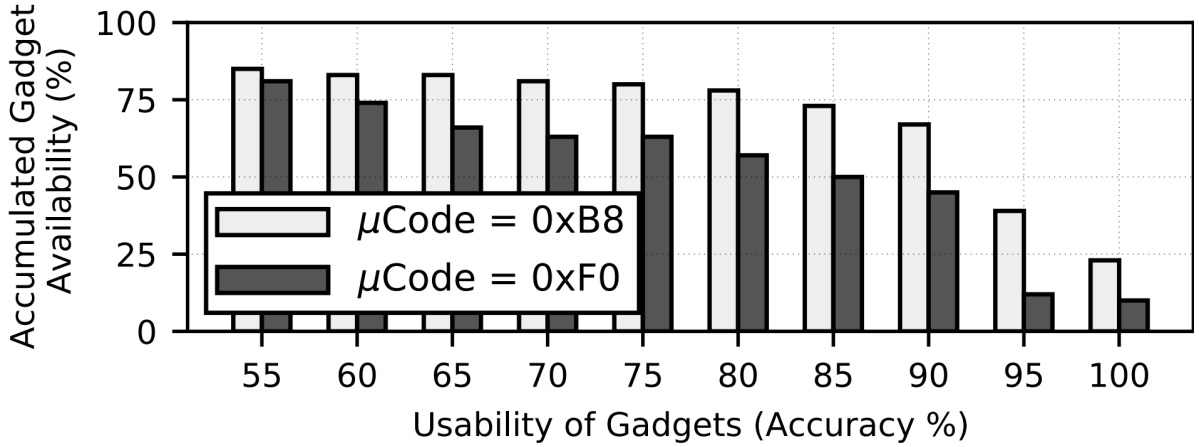
Attack accuracy under different microcodes

PowSpectre Gadget Analysis

	RAX	RBX	RCX	RDX	RDI	RSI	RSP	RBP	Total
TEST	3371	0	2	179	0	24	19	14	3609
CMP	1672	8	3	168	374	1	2	398	2626
SHR+ TEST/CMP	3122	100	409	459	5	20	0	70	4185
Mask	0xD3	0xD5	0x91	0xDD	0xE1	0x91	0xD1	0x8B	0xFF

Availability of *potential* PowSpectre *transmitter gadget*

Ability to leak **all 8 bits** of the first byte of a register



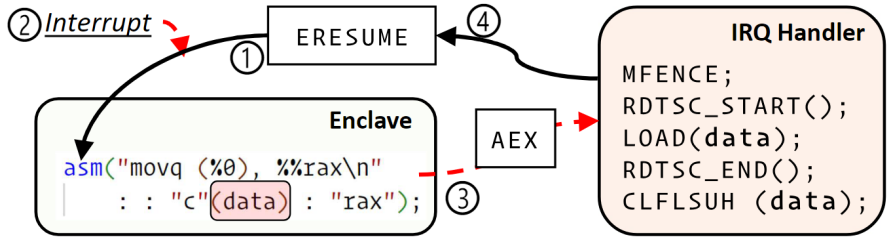
Tradeoff between gadget availability and usability

73% of the gadgets have 80%+ bit leakage accuracy

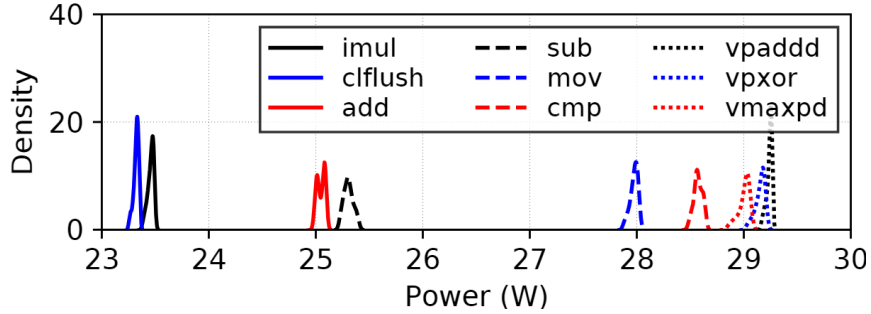
UCF

More on Paper

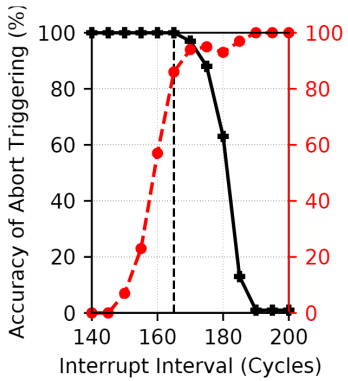
- Further analysis on Instruction replay primitives.
- Investigation of power differentiability of instructions.
- Further evaluation of TMPlayer and PowSpectre.
- More analysis on PowSpectre gadgets.
- And more…



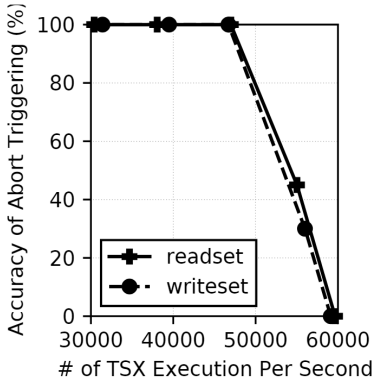
(a) Illustration of the microbenchmark under test.



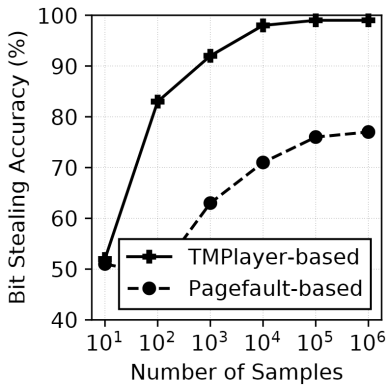
Power signature of representative instructions



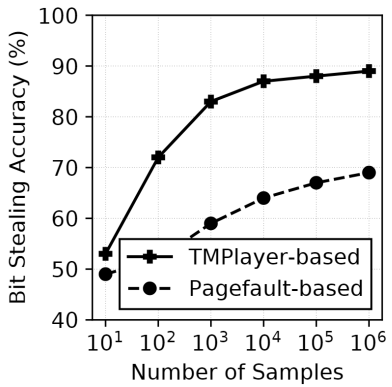
(a) *TMPlayer-I*



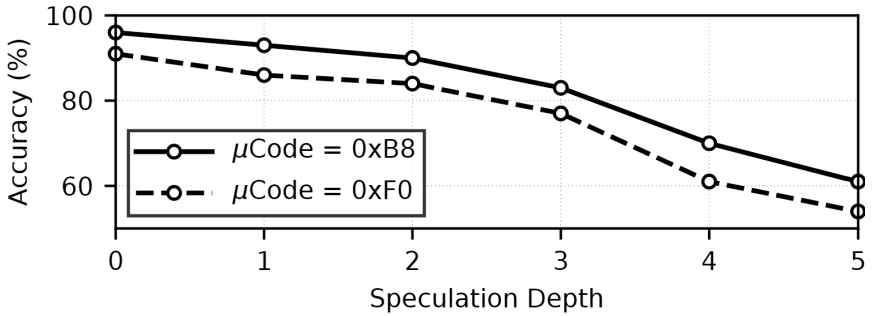
(b) *TMPlayer-E*



(a) μCode 0xB8



(b) μCode 0xF0



Sensitivity of gadgets

UCF

Conclusion

- We present PowSpectre- a software-based power side channel for the speculative domain.
- First work to highlights the secret leakage capabilities of power side channel in speculation.
- We investigate prior instruction replay techniques and discover their shortcoming in transient execution.
- We design TMPlayer- an instruction replay technique utilizing the Intel TSX.
- PowSpectre can be used to exfiltrate enclave secretive data in the speculative domain with very high accuracy.
- We demonstrate the wide availability of the power distinguishable gadgets for PowSpectre.

Thanks! Questions?

Md Hafizul Islam Chowdhuryy
CASR Lab (https://casr.ece.ucf.edu)
Email: hafizul.islam@ucf.edu