Detecting Spectre Attacks by identifying Cache Side-Channel Attacks using Machine Learning

Presented at **WAMOS 2018**

Jonas Depoix
RheinMain University of Applied Sciences
Wiesbaden, Germany
jonas.depoix@student.hs-rm.d

Philipp Altmeyer
RheinMain University of Applied Sciences
Wiesbaden, Germany
philipp.b.altmeyer@student.hs-rm.de

Agenda

Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim Geisenheim

1. **Motivation**
2. **Approach**
   - Concept
   - Selected HPCs
   - Data set
   - Implementation
3. **Results**
4. **Demo**
5. **Conclusion**
6. **Future Work**

Motivation

Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim Geisenheim

- Spectre and Meltdown pose considerable threats
- Spectre software mitigations have drawbacks
  - Mitigations have to be implemented and deployed for every individual application
    - Vendors have to provide updates
    - Users have to keep their applications up to date
  - Considerable performance impact
  - Customers of cloud providers are at great risk
    - An up to date VM can be vulnerable to other VMs running on the same hypervisor

Motivation

Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim Geisenheim

Real-time detection system for Spectre attacks could ...

- … keep users safe, even when they use vulnerable software
- … potentially be run with less performance impact, than mitigations
- … be run on hypervisors by cloud providers, to identify malicious VMs

➢ We wanted to implement a **real-time detection system for Spectre attacks**, to test if it is a feasible alternative/supplement to software mitigations

Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim Geisenheim

Approach

Concept

Hochschule RheinMain
University of Applied Sciences
Wiesbaden Rüsselsheim Geisenheim

How does our real-time detection system for Spectre work?

- We mitigate Spectre, by shutting down cache side-channel attacks
- We exploit that cache side-channel attacks have side-effects as well
  - Memory has to be accessed frequently and repeatedly
  - Side-channel attacks induce distinct cache usage patterns

Concept

How does our real-time detection system for Spectre work?

1. Track cache usage with Hardware Performance Counters
   - Special purpose CPU registers
   - Counts occurrences of certain CPU events
     - E.g. clock cycles, L3 cache hits, L3 cache misses, …
   - Can be attached to individual threads, processes, or the entire CPU
2. Classify processes as malicious or benign using a neural network
   - Good at detecting patterns in data
   - Effective at solving classification tasks

Selected HPCs

- L3 cache misses (L3_TCM)
  - Side-channel attacks cause high cache miss rates
- L3 cache accesses (L3_TCA)
  - Reference for total cache activity
- Total instructions (TOT_INS)
  - Reference for CPU load
  - Malicious process has higher rate of cache misses ⇔ executen instructions

Data set

Hochschule RheinMain
University of Applied Sciences
Wiesbaden Rüsselsheim Geisenheim

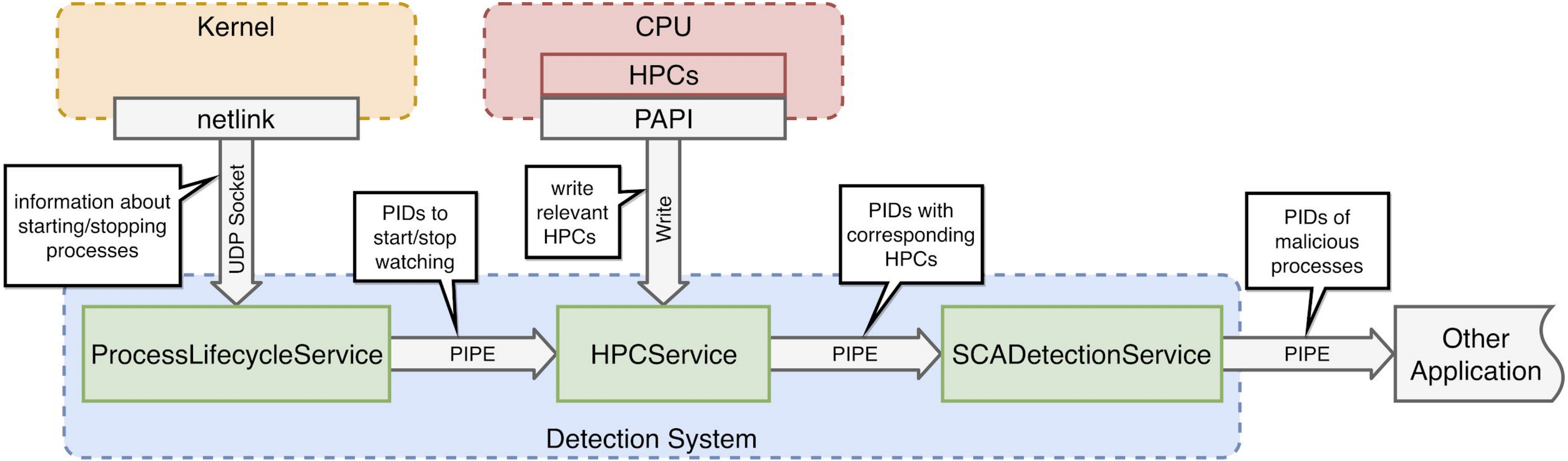
11 scenarios for training and validation

- Server workloads
  - Wordpress CMS (PHP)
  - Ghost CMS (Node.js)
- Desktop workloads
  - stress
  - Web browsing
- Spectre implementations
  - Variants 1 and 2 implementations written in C
  - Variant 1 JavaScript implementation

➢ Selected HPCs are recorded over 60 seconds in 100 ms intervals

Implementation

Hochschule RheinMain
University of Applied Sciences
Wiesbaden Rüsselsheim Geisenheim

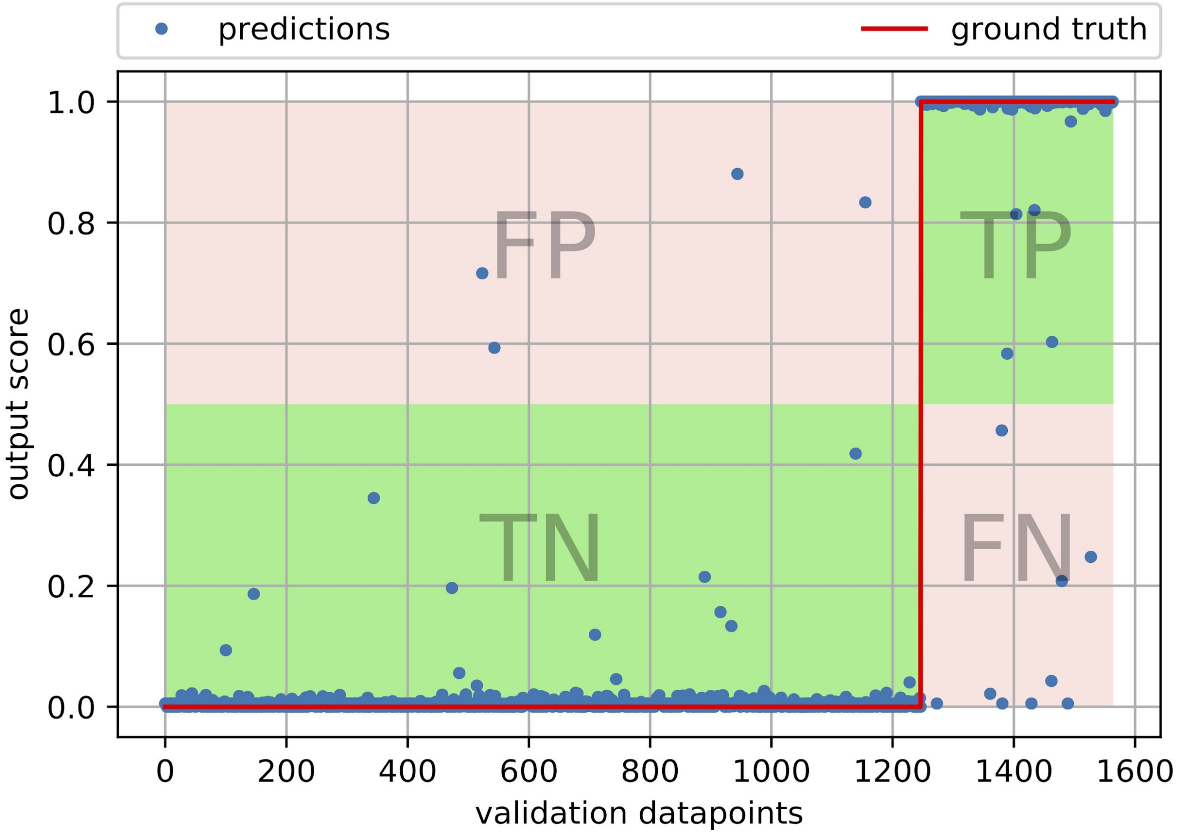


Results

Hochschule RheinMain
University of Applied Sciences
Wiesbaden Rüsselsheim Geisenheim

Ground truth	
# datapoints	**1564**
# positives	317
# negatives	1247

Predictions	
Accuracy (total)	**99.23%**
Accuracy (positives)	97.16%
Accuracy (negatives)	99.67%
F-score	**0.9716**
# true positives (TP)	308
# false positives (FP)	4
# true negatives (TN)	1243
# false negatives (FN)	9



Demo

Conclusion

Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim Geisenheim

- We have successfully implemented a real-time detection system for Spectre attacks
  - Accuracy above 99%
  - Detecting side-channel attacks by analyzing HPCs with neural networks shows great potential

➢ **Real-time detection is a feasible alternative/supplement to software mitigations**

Future Work

Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim Geisenheim

- Performance could be improved
- Neural network could be trained on a wider variation of Spectre implementations
- Neural network should be trained on a wider variation of CPU models
- Test if detection system is applicable to Meltdown
- Test if detection system is applicable to cross-VM attacks

Thank you for your attention!

Any questions?