There’s Always a Bigger Fish: A Clarifying Analysis of a Machine-Learning-Assisted Side-Channel Attack

Jack Cook, Jules Drean, Jonathan Behrens, Mengjia Yan
ML-Assisted Side-Channel Attacks

• Are highly effective and even work with noise
• Work as a black box and are hard to interpret

Bigger Fish is a detailed analysis of a misunderstood side-channel attack
Agenda

1. Background
2. A Surprising Experiment
3. In Depth Security Analysis
4. Findings & Conclusion
Timing Side Channels

Shared Resources

- CPU
- Branch Predictor
- Cache
- DRAM
- ...

"Victim Secret"
Website Fingerprinting Attacks

• Very serious privacy implications
• Can be mounted from JavaScript
• Good benchmark for side channels
Website Fingerprinting: Machine-Learning Classifier

TRACE COLLECTION

TRAIN

"a.com"

INFER

"b.com"
A Cache-Occupancy Attack*

```javascript
loop {
    start = time();
    counter = 0;
    while (time() - start < 5ms) {
        counter++;
        SWEEP_CACHE();
    }
    Trace[start] = counter;
}
```

A Surprising Experiment

ATTACKER'S CODE

```c
loop {
    start = time();
    counter = 0;
    while (time() - start < 5ms) {
        counter++;
        SWEEP_CACHE();
    }
    Trace[start] = counter;
}
```

Sweep-Counting Attack vs Our Attack:

<table>
<thead>
<tr>
<th>Browser</th>
<th>Sweep-Counting Attack</th>
<th>Our Attack</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chrome on Linux</td>
<td>91%</td>
<td>97%</td>
</tr>
<tr>
<td>Chrome on Windows</td>
<td>80%</td>
<td>93%</td>
</tr>
<tr>
<td>Safari on macOS</td>
<td>73%</td>
<td>97%</td>
</tr>
</tbody>
</table>

97% 97% 93% 97% 97% 97% 91% 91% 80% 80% 73% 73%
What is the primary side channel?
ML-Assisted Side-Channel Attacks

• Work as a black box and are hard to interpret

Bigger Fish is a detailed analysis of a misunderstood side-channel attack
Scientific method

Observation

Background Research

Hypothesis

Experiment

Analyze data

Conclusion
Scientific method

What could cause throughput to change over time?

Frequency scaling

Train machine learning model → 94.2% accuracy

Little to no signal from frequency scaling

Repeat experiment with frequency scaling disabled

95.2% accuracy when memory accesses removed

95.2% accuracy when memory accesses removed

Train machine learning model → 94.2% accuracy

Little to no signal from frequency scaling

Repeat experiment with frequency scaling disabled
What could cause throughput to change over time?

- Frequency scaling
  - Repeat experiment with frequency scaling disabled
  - Train machine learning model → 94.2% accuracy

- Little to no signal from frequency scaling

- What else?
  - Little to no signal from CPU core contention
  - CPU core contention

- Repeat experiment with attacker + victim pinned to separate cores
  - 94.0% accuracy
  - 94.2% accuracy when memory accesses removed

Scientific method
What could cause throughput to change over time?

Frequency scaling

Repeat experiment with frequency scaling disabled

Train machine learning model

→ 94.2% accuracy

95.2% accuracy when memory accesses removed

Little to no signal from frequency scaling

What else?

Little to no signal from CPU core contention

Scientific method

94.0% accuracy with separate cores

94.0% accuracy

Repeat experiment with attacker + victim pinned to separate cores

Interference from interrupts

94.0% accuracy with separate cores

What else?
System Interrupts

- Used to deal with asynchronous events
  - e.g. Graphics interrupts render content on a display
- Some can be “pinned” to specific cores, some can’t
What could cause throughput to change over time?

- Frequency scaling
  - Repeat experiment with frequency scaling disabled
  - Train machine learning model → 94.2% accuracy

- CPU core contention
  - Repeat experiment
  - 94.0% accuracy with no frequency scaling
  - 94.2% accuracy when memory accesses removed

- Interference from interrupts
  - Repeat experiment
  - 88.2% accuracy with separate cores

There’s still a lot of signal!
Movable interrupts

- 16: Mouse
- 23: Keyboard
- 27: Network card
- 30: Graphics card
Non-movable interrupts

- Timer interrupts
- IRQ work interrupts
Non-Movable Interrupts

- Can’t be isolated from any cores
- Are necessary for the operating system to function
- Have not been studied in detail for side channels
What could cause throughput to change over time?

Frequency scaling

Repeat experiment with frequency scaling disabled

Train machine learning model → 94.2% accuracy

95.2% accuracy when memory accesses removed

Little to no signal from frequency scaling

What else?

CPU core contention

Repeat experiment

94.0% accuracy

94.2% accuracy with no frequency scaling

Little to no signal from CPU core contention

Scientific method

20

There’s still a lot of signal!

88.2% accuracy

What else?

Interference from interrupts

Repeat experiment

94.0% accuracy with separate cores

88.2% accuracy with movable interrupts isolated

There’s still a lot of signal!

Non-movable interrupts

Repeat experiment

88.2% accuracy with movable interrupts isolated

This experiment is impossible!
eBPF

• Allows instrumentation of the Linux kernel at runtime

• We developed a tool to monitor interrupt characteristics

• Records time at beginning and end of interrupt handlers
Interrupt Handling Time

Counter Value

Time

amazon.com

weather.com
99% of gaps can be explained by the presence of interrupts.
More in the paper!

• Randomized timer countermeasure
• Cache + interrupt noise experiments
• Virtual machine isolation
• Further discussion of non-movable interrupts
• Analysis of web browser timers
• And more!
Findings and Conclusion

- Machine-learning-assisted attacks are powerful but hard to interpret
- Sweep-counting “cache-occupancy” attack* primarily exploits system interrupts
- Non-movable interrupts have strong security implications
- We release our analysis toolset at https://github.com/jackcook/bigger-fish

Demo

jackcook.github.io/bigger-fish

process isolation

cache side channel

interrupt side channel