

Browser Cache Attacks in 2026

Advisor: **Roland Czerny**

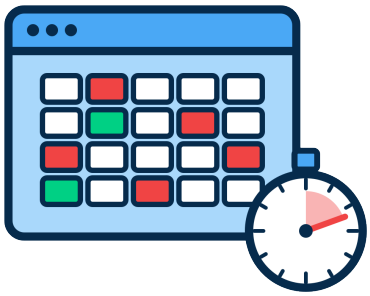
Motivation

About ten years ago, researchers showed that cache side-channel attacks work from JavaScript. A malicious website could spy on other tabs and programs simply by measuring how long memory accesses take. Browser vendors reacted, especially after Spectre: they reduced the precision of timers, restricted `SharedArrayBuffer`, and moved websites into separate processes. Attackers in turn found new timers and attacks that need fewer and fewer browser features. Since then, browsers, WebAssembly, and CPUs have all changed. The question is: can cache attacks still be done from the browser in 2026?

In this project, you will revisit known browser-based cache attacks, test them on current browsers and hardware, and find out which ones still work and why.

Goals and Tasks

- Learn about cache side-channel attacks and the defenses in current browsers.
- Reproduce known attacks (e.g., Prime+Probe, cache occupancy) in JavaScript or WebAssembly.
- Evaluate which attacks still work on which browsers and CPUs, and what stops the others.



Literature

- M. Schwarz et al.
  Fantastic Timers and Where to Find Them: High-Resolution Microarchitectural Attacks in JavaScript
  FC
- A. Shusterman et al.
  Prime+Probe 1, JavaScript 0: Overcoming Browser-based Side-Channel Defenses
  USENIX Security

Courses & Deliverables

- ☑ **Introduction to Scientific Working**
  Short report on background
  Short presentation
- ☑ **Bachelor Project**
  Project code and documentation
- ☑ **Bachelor's Thesis**
  Project code
  Thesis
  Final presentation

Recommended if you're studying

☑ CS ☑ ICE ☑ SEM

Prerequisites

- Basic knowledge of CPU caches
- Programming experience in JavaScript or C
- Interest in side-channel security

Advisor Contact

roland.czerny@tugraz.at