Analysis of Methods for Background Execution in Modern Web Applications

Analyse von Verfahren für Hintergrundausführung in modernen Webanwendungen
Bachelor-Thesis von Yannick Reifschneider
Tag der Einreichung:

1. Gutachten:
2. Gutachten:

TECHNISCHE UNIVERSITÄT DARMSTADT

Fachbereich Informatik
Security Engineering

Analysis of Methods for Background Execution in Modern Web Applications
Analyse von Verfahren für Hintergrundausführung in modernen Webanwendungen

Vorgelegte Bachelor-Thesis von Yannick Reifschneider

1. Gutachten:
2. Gutachten:

Tag der Einreichung:

Erklärung zur Bachelor-Thesis

Hiermit versichere ich, die vorliegende Bachelor-Thesis ohne Hilfe Dritter nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die aus Quellen entnommen wurden, sind als solche kenntlich gemacht. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Darmstadt, den May 19, 2019

(Yannick Reifschneider)

Contents

1 Introduction

1.1 Motivation

- Possible side channel attack (i.e. via sensor readings)
- A XSS vulnerability in a popular website could use the visitors as a botnet for a DDOS attack

1.2 The JavaScript execution model and the event loop

Why a simple infinite loop is not feasable: Web page becomes unresponsive. If you don’t yield back to the event loop, you can no longer react to changes in the environment, for example to detect, if the browser is now in the foreground again. This is at least true in the main loop, have to check for service worker or web worker context.

2 Analysis of different background execution methods

2.1 Timers

Standard method for scheduling a recurring function in JavaScript. The setInterval function allows to specify a function and an interval in milliseconds after which a function is repeatedly called until the interval is cancelled.

2.2 Web workers

Using the worker-timers[1] library to run a scheduler on a web worker, which calls a callback on the main loop. This circumenvents the setInterval throttling, when a browser tab is in the background.

2.3 Service workers

Service workers have advantages. They run independent of the browser tab. They stay can stay aliver after the browser tab, which installed the service worker is closed.

- Multiple methods for background execution:
- Simple set interval after activation
- In response to network request (corresponding website has to be open to trigger a network call)
- Website push notifications (has to be allowed by user)
- Web Background Synchronization API[2]

2.4 Sensor readings

The Sensors API allows to read from device sensors like accelerometer, gyroscope oder ambient light sensor. You can define the frequency in which you want to receive new sensor readings.

2.5 Websocket connection

Opening a websocket connection and leaving it open. You can execute code after recieving a message on the websocket channel. Have to check for the frequencies and how long the websocket channel can stay open.

1 https://github.com/chrisguttandin/worker-timers
2 https://wicg.github.io/BackgroundSync/spec/

3 Web browser behaviour regarding the different methods

At the time of writing this thesis, the following Browser versions were current:
Chrome 74
Firefox 66
Safari 11

3.1 Desktop web browsers

3.1.1 Google Chrome

Chrome on macOS does not allow sensor readings while in background.

AmbientLightSensor has to be enabled via flags.

3.1.2 Mozilla Firefox

3.1.3 Apple Safari

3.2 Mobile web browsers

On iOS we only analyse Mobile Safari, because Apple does not allow other browser engines in the Apple AppStore. Every other browser app has to use the system-provided webview to be in accordance with § 2.5.6 from Apple Review Guidelines[3].

On Android, we can differentiate between different browser engines.

3.2.1 iOS Mobile Safari

3.2.2 Chrome for Android

3.2.3 Firefox for Android

[3] `https://developer.apple.com/app-store/review/guidelines/#software-requirements`

4 Tracing of background execution on popular websites

Using the Alexa Top 100 Website list.

4.1 Method for measuring background execution

Maybe with puppeteer[4], web developer tools or with OpenWPM[5] or with simple hooking the JavaScript functions

4.2 Evaluation of findings

[4] `https://pptr.dev/`

[5] `https://github.com/mozilla/OpenWPM`

5 Conclusion