

Maintainer steps in to prevent popular Javascript project SockJS from being abandoned

[WebSockets](#) is a protocol for two-way communication between a client (such as a web browser) and a server (such as a web application) that allows for easier asynchronous and real-time updates. Many of the most interactive web applications are built on top of WebSockets, which was standardized in 2011.

However, not all browsers at the time supported it, and other pieces of internet infrastructure such as proxies sometimes interfered. In 2011, SockJS was created as a cross-browser JavaScript API that provided WebSocket-like behavior that could be used whether the underlying system supported WebSockets or not.

SockJS contains both a client and server implementation. It's directly used by over 800 JavaScript packages, including both the [Meteor](#) and [Next.js](#) frameworks, and is downloaded over 11 million times per week. It's important that SockJS stay maintained and secure for those platforms that are built on it.

Eight years ago, Bryce Kahle took over maintainership of SockJS from the original author. He was using it in his day-to-day work, and it made sense to help out. In 2018 he signed up with Tidelift to receive income for his work.

As SockJS was a fairly mature project, Bryce noted that the majority of his work recently was updating dependencies due to security issues—critical work to keep his downstream users safe.

As time went on, though, circumstances changed. Bryce's full time job was no longer working with JavaScript, and he described his SockJS work as being mostly on autopilot. In [May of 2022](#), Bryce put out the call looking for a new maintainer to step in.

As of August, he had not found a new maintainer yet. He reached out to Tidelift noting that he was intending to stop maintenance of his SockJS packages. After some discussion, [we amplified his call for new maintainers to our community of maintainers](#) to see if someone would be interested in picking up maintenance.

[Asif Saif Uddin](#) is an open source Python developer, the maintainer of the [Celery project](#), and has partnered with Tidelift since 2018. He saw the call for a new maintainer and responded to Bryce.

When asked, Asif mentioned a few reasons why he picked it up:

- It's a popular package that needed another hand.
- It was backed by Tidelift, so he had the confidence he had support and the motivation of being paid for maintenance.
- It was a learning opportunity, and Bryce was very helpful with insights on the project.

The screenshot shows the Tidelift interface for the 'sockjs' package. At the top, there's a 'This package is lifted!' banner with a 'PAY THE MAINTAINERS' button. Below this, there are several sections: 'Maintainer commitments' with a list of security and development practices; 'Releases' showing the latest version (0.3.24) and total releases; 'Vulnerabilities' showing the latest and most severe issues; 'Dependencies' showing total, runtime, development, and other dependencies; and 'Project Usage' showing total projects using, direct use, and transitive use.

Here is the SockJS package page inside of the Tidelift application, showing the secure development practices the project is following, and also including additional useful information about releases, vulnerabilities, dependencies, and where it is being used within an example organization.

Bryce worked to hand over maintainership, and Asif is now the upstream maintainer of the SockJS project. He also has signed up to continue partnering with Tidelift for SockJS maintenance.

Now, Tidelift customers can continue to use SockJS—and other packages that rely on it—with confidence, knowing that Asif is being paid to ensure it follows a robust set of enterprise secure software development practices. What's more, SockJS is no longer simply on autopilot, thanks in part to the income Asif derives from Tidelift, ensuring it stays resilient and healthy into the future.