



# Open Source Skills Report 2026

Case Study: Rustls



## 4. Case Studies & Fireside Chats

### 4.1 Case Study: Rustls

**Joe Birr-Pixton**  
**Security and Embedded Engineer**



The story of Rustls and its creator Joe Birr-Pixton, showcases the potential of open source software as both a career path and a platform for addressing critical industry needs. Joe's story is one of rare successes in a landscape where most open source contributors work without adequate financial support, but it highlights the unique skillset developed through open source work. His experience offers valuable insight into skill development through community engagement, and the systemic barriers facing those working to keep open source projects viable and build enduring careers in this space.

#### *The Beginning*

Joe began Rustls as a software project in 2016, as a personal project to improve the landscape of TLS software and stay connected with cybersecurity and programming. At the time, he had transitioned into a less security focused role and wanted a project that would let him hone his expertise. His choice of programming language, Rust, was newly stabilised at version 1.0, and aligned with his interest in leveraging memory safe coding practices to build a Transport Layer Security (TLS) library. Rust as a language is well known from a security perspective because of its memory safety features which have been promoted by the White House.

Out of this came Rustls, a memory-safe TLS library written in Rust that focuses on simplicity, safety and performance.

Initially, Rustls was a weekend hobby project with no expectation of commercial support. Joe spent years developing and maintaining the project in his spare time, while working a full-time job. This role as an open source software "maintainer" meant he held accountability for the Rustls project to the rest of the world. As he refined the library and the use of memory safety in software security became increasingly important, Rustls' reputation grew. It eventually attracted interest from users and organisations eager to fund additional features and maintenance, as is often the case with open source projects as they gain users and adoption.

#### *Project Evolution & Skill Development*

Joe's work on Rustls demonstrates how open source projects cultivate skills far beyond technical expertise. Through maintaining the project, he deepened his security knowledge and learned to manage contributors, set development priorities, navigate governance structures, and sustain a community. These capabilities, while critical, are often undervalued in traditional career pathways, despite being central to the eventual commercialisation of Rustls.

### *A Rare Opportunity*

In 2021, Joe began receiving funding offers for Rustls. By 2023, he was able to leave his role at a large technology company and focus on the project full-time. He emphasises that this is highly unusual: most maintainers juggle their projects alongside other responsibilities, often at personal cost. The open source ecosystem still relies heavily on unpaid labour, with limited routes to full-time sustainability. While Joe attributes some of his success to timing and luck, his skill and persistence have been equally important.

### *Current Funding Landscape*

As of 2025, Rustls is primarily supported through a grant from the Sovereign Tech Agency, delivered via ISRG's Prossimo initiative, which focuses on memory-safe implementations of core internet protocols. This funding supports Joe and one co-maintainer, and the team is exploring future options with the Rust Foundation.

### *Challenges for Rustls*

Despite growing adoption, memory-safe libraries like Rustls continue to face barriers, particularly in an ecosystem dominated by long-established projects such as OpenSSL. Joe notes that replacing even 30 - 40% of OpenSSL use would be a major milestone. He must also balance the project's core technical aims with the expectations of sponsors; challenges that are common across open source maintenance and underline the need for more systemic support.

### *Recent Developments and Governance Evolution*

Since 2024, Rustls has undergone a period of formalisation and renewal. Joining the Rust Foundation's Rust Innovation Lab has provided governance support, strategic guidance and help navigating sustainability models. These types of support are often unavailable to technically led open source projects. While this does not include direct funding, the Rust Foundation's support has helped strengthen project structures and clarify Rustls' long-term direction, marking a shift from a purely technical effort to one requiring community stewardship, collaboration, and long-term planning.

### *Open Source as a Skills Platform*

Joe's journey illustrates how open source projects act as powerful incubators for talent. Through Rustls, he has developed deep expertise in memory safety and cryptographic design alongside leadership, stakeholder engagement and collaborative skills. His role now involves long-term planning and governance, including designing a stable API that can be supported "for years and years".

He also draws parallels with his civic role as a town councillor and deputy mayor, where decision-making is open, collective and accountable, mirroring many of the principles and challenges of open source governance.

For the UK, this presents a significant opportunity. Open source initiatives can cultivate the technical and leadership skills needed for future cybersecurity and software engineering challenges. However, realising this potential requires addressing the systemic barriers that make Joe's experience the exception rather than the norm.

### *Sustainability and Open Source as a Public Good*

Joe emphasises that open source software should be treated as public infrastructure requiring stable, long-term investment; something private companies rarely provide. He identifies two viable sustainability models: public good funding, through governments or grant-making bodies, or private good subsidisation, where commercial models are built around open source cores. He notes that switching between these models mid-way is rarely successful; choosing a clear path early is essential for trust and long-term viability.

Rustls now underpins an estimated 200,000 software projects and indirectly supports 2–3 billion end users through services such as WhatsApp on Android. This scale underscores the need for sustained and predictable investment.

The evolution of Rustls positions it not just as a technical achievement but as an example of how maintainers can build meaningful, sustainable careers while influencing critical global digital infrastructure. Joe's experience highlights both the transformative potential of open source work and the fragility of the systems that support it, particularly within the UK's developing digital ecosystem.

### **Joe Birr-Pixton, Creator of Rustls**

Joe hails from Cambridge, UK. He is a security engineer, formerly with Twilio, Electric Imp, BlackBerry, Good Technology, Thales, and nCipher. These days he mostly works in Rust. His professional experience is in C and C++. He is the original author of the “rustls” Rust crate, which provides TLS support to most of the Rust ecosystem.

First published by OpenUK in 2026 as part of Open Source Skills Report 2026 ©OpenUK2026





©OpenUK 2026 - Registered Office: 8 Coldbath Square, London EC1R  
5HL Company Number: 11209475 - VAT Registration: GB379697512

