

Leveraging Rusts Safety Mechanisms for a Threadsafe Java Framework

(REFERENCES)

- [1] A. D. Fekete, "Teaching Students to Develop Thread-safe Java Classes," School of Information Technologies, University of Sydney, Australia, 2008.
- [2] S. Rahman, C. Li, and A. Shi, "TSVD4J: Thread-Safety Violation Detection for Java," The University of Texas at Austin, USA, 2023.
- [3] M. Pradel and T. R. Gross, "Fully Automatic and Precise Detection of Thread Safety Violations," Department of Computer Science, ETH Zurich, 2012.
- [4] S.-E. Choi and E. C. Lewis, "A Study of Common Pitfalls in Simple Multi-Threaded Programs," Los Alamos National Laboratory and University of Washington, USA, 2000.
- [5] R. Ross, M. McEvilly, and J. C. Oren, "Systems Security Engineering: Considerations for a Multidisciplinary Approach in the Engineering of Trustworthy Secure Systems," NIST Special Publication 800-160, National Institute of Standards and Technology, Gaithersburg, MD, USA, 2016.
- [6] A. Mettler, D. Wagner, and T. Close, "Joe-E: A Security-Oriented Subset of Java," in NDSS, 2010, pp. 357-374.
- [7] A. Ugarte, "What Is Thread-Safety and How to Achieve It?", Baeldung, 2024. [Online]. Available: <https://www.baeldung.com/java-thread-safety>. [Accessed: Jun. 22, 2025].
- [8] Rust Project Developers, "The Rustonomicon," 2025. [Online]. Available: <https://doc.rust-lang.org/nomicon/>. [Accessed: Jun. 22, 2025].

[9] IEEE Computer Society, "What is Software Security and Why is it Important?", 2025. [Online]. Available: <https://www.computer.org/resources/software-security>. [Accessed: Jun. 22, 2025].

[10] R. Jung, J.-H. Jourdan, R. Krebbers, and D. Dreyer, "Safe Systems Programming in Rust: The Promise and the Challenge," *Commun. ACM*, vol. 64, no. 4, pp. 144-152, 2020.