

CS F301 : Project Proposal

Project Title:

Comparative Implementation of Command Line Secret Message Hider in Rust and C/C++ with Emphasis on Rust's Benefits for Steganography

Team Members:

(Group -24)

Prithvinandan Kunjithaya (2021A7PS3051G)

Hardav Raval (2021A7PS3041G)

Prabhanshu Tripathi (2021A7PS2065G)

Rahul Muruganandam (2021A7PS0012G)

Problem Statement:

The problem we aim to solve is to provide a secure and efficient way to hide and retrieve secret messages within PNG files, while underlining the specific advantages of Rust for steganography and comparing these to the capabilities of other programming languages.

Project Description:

We propose the development of a comparative implementation of a command-line program that allows users to hide secret messages within PNG files, with a strong emphasis on Rust's benefits for steganography. This project aims to provide a secure and user-friendly way to encode, decode, remove, and manage hidden messages within PNG images. The project will include two distinct implementations: one in Rust, serving as the primary target, and the other in C/C++ for the purpose of comparison. PNG files are chosen as the medium for secret message storage due to their widespread use and the flexibility they offer for embedding information.

Our Contribution:

Our project's primary contribution lies in emphasizing and showcasing the unique advantages of using Rust for steganography. Steganography is the practice of hiding data within other data, typically digital images in this context. What sets our project apart is the distinct focus on the following aspects:

– Rust's Memory Safety and Security: While steganography has been explored in various programming languages, we are demonstrating how Rust's memory safety features, including strict ownership and borrowing rules, can significantly enhance the security and reliability of steganography implementations. Our project will clearly articulate how this is a notable advancement in the field, ensuring that hidden messages remain secure and the implementation remains robust against common programming errors.

– Concurrency and Performance Optimization: Steganography operations can benefit from concurrency, especially when multiple threads are involved. Rust's built-in support for concurrent programming enables us to optimize performance, demonstrating that Rust can

outperform other languages. This highlights a novel approach to improving the performance of steganography while maintaining security.

– Comparative Analysis: We will conduct a comprehensive comparative analysis between the Rust and C/C++ implementations, drawing attention to Rust's advantages for steganography. This in-depth comparison will provide a unique insight into how Rust excels in solving the problem of secure message hiding within PNG files compared to C/C++, furthering the field's understanding of language selection in this context.

Conclusion:

The Comparative Implementation of Command Line Secret Message Hider in Rust and C/C++ with an Emphasis on Rust's Benefits for Steganography will make a distinct contribution to the field by emphasizing the unique advantages of Rust in steganography. Our project seeks to not only provide a solution but also elevate the conversation around language selection, security, and performance in the context of steganography, ultimately contributing to the body of knowledge in this domain.