Dante Broggi
CIS 542
December 6, 2023

Project Report

PNG Text and trailing data search

PROJECT GOALS:

The goal of this project was to create a simple command line PNG metadata reader, similar to the Strings view on <https://fotoforensics.com>, which also includes data past the official end marker in the data, that is the IEND chunk. This project was inspired by the <https://en.wikipedia.org/wiki/ACropalypse> privacy vulnerability and the Strings view in the <https://fotoforensics.com> service. Google Pixel screenshot cropping did not properly truncate cropped image files, allowing recovery of "removed" image context, like password displays.

SCOPE STATEMENT:

The project will not include a GUI, PNG renderer, or search functionality. This allows the program to be a simple Rust command line tool to examine the metadata in overwritten but not truncated files and other cases of extra large PNG files.

I am including 2 example images used through the fair for educational use expectations, from <https://www.da.vidbuchanan.co.uk/blog/hello-png.html> and <https://commons.wikimedia.org/wiki/File:Abstract_Blue_Background.png>, although additional images were used for testing, I will not providing them in the repository for personal reasons.

Additional work the program could perform includes attempting to recover image data from incomplete image data, as done for Google Pixel screenshots by <https://acropalypse.app>, or looking for hidden pixels, either out of the declared width/height or in transparent pixels, and the other forensics data presentation performed by the <https://fotoforensics.com> service. However, such analysis was considered to be out of scope for this project.

Look at the more developed <https://fotoforensics.com> / <https://lab.fotoforensics.com/?show=lab> service for such analysis, or at <https://en.wikipedia.org/wiki/ACropalypse> and <https://acropalypse.app> for more information about the vulnerability which inspired this project.

METHODOLOGY:

The PNG Text program is a command line program which reads a binary (PNG) file, and describes all PNG data in the file, regardless of whether or not the data chunks form a complete valid PNG. The program even looks for overlapping PNG chunks, thus it can appear `O(N²)` for moderate sized input, though as each chunk has a maximum length, it is `O(N)` in the limit.

For this program I used the Rust programming language, its associated Cargo build system, and its standard crates.io packages as libraries, and the VSCode IDE and GitHub Desktop tools to assist development. The program was based on the PNG specification at <https://www.w3.org/TR/png-3/> and a blog post by a David Buchanan at <https://www.da.vidbuchanan.co.uk/blog/hello-png.html>.

Specifically, the program output format is a list of valid PNG chunk chains contained in the file, starting with those which begin with a valid PNG header, in order, followed by those chunk chains which do not begin with a valid PNG header, and then a list of bytes&byte offsets which do not form valid chunk chains. Where a chunk chain is a series of PNG chunks, as described by the PNG specification, such that each chunk begins at the byte after the previous ends. The program even looks for overlapping chunk chains, thus it can appear `O(N²)` for moderate sized input, though as each chunk has a maximum length, it is `O(N)` in the limit.

PROJECT DELIVERABLES:

The project was specified as delivering:

- A command line program which produces a textual description of a PNG file, specifically of the metadata and any trailing data.
- An as-needed for the demo collection of test PNG files.
- A user manual which describes the use and output format of the program, as the README.md file in the repository.
- A 7 min presentation in class or recorded, which occurred in class on Wednesday December 6, 2023.
- And this project report document here, approximately 2 pages in length, single-spaced, 12 punto Times New Roman letters, max 1-inch margin on all sides.

All of which is to appear in a public GitHub repository by December 11, 2023 @ 11:59 PM, which shall be <https://github.com/Dante-Broggi/png-text>.

The above seems like it will be done as soon as this document here is published and the repository is marked public.