Implementation of Steganographic Techniques

TJHSST Computer Systems Lab 2006-2007
Danny Friedheim

Abstract

The purpose of this project is to design a steganographic program in C++ capable of hiding a message within a WAVE file, and then later extracting the hidden message. It should be able to work with any WAVE formatted sound file, and the message ideally will not be detectable.

Background

Steganography is the art of hiding a message or data within another set of data, such as an image, sound file, or computer program. The idea is that the presence of the message is unknown, as opposed to cryptography where the presence of the message is known, but it is unreadable. The WAVE file format is a sound file format composed of various “chunks” of data. By utilizing this “chunk” organization of the files, data can be hidden rather easily in the file without detection.

Procedures and Methods

The methods are all run by command-line arguments. That way, the same program can be run multiple times to do different things, based on which flags are used. The –i or input method inserts a message into the “fmt” chunk of the WAVE file. This keeps the data within the file completely intact, resulting in a playable and almost identical replica of the original file. The -i2 is a more complex method that hides the message better by putting it in the actual data by a method called least significant bit replacement. This method replaces the first byte of each group of 4 sound data bytes with a character from the message, and also maintains a playable WAVE files.

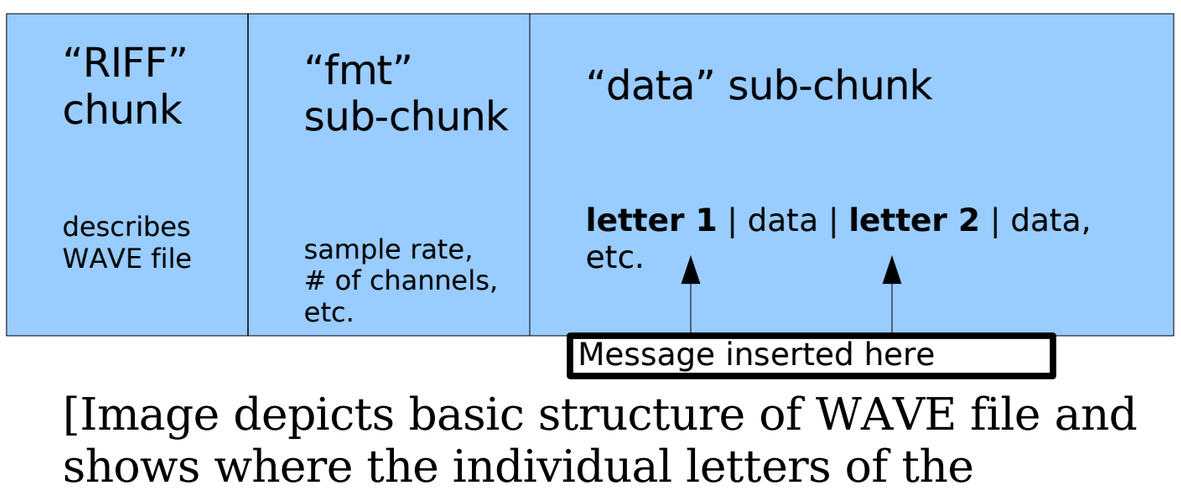
Results and Conclusions

I have a working C++ program capable of inserting hidden messages into the sound data of all WAVE files and extracting them later. The resulting files are still playable and sound virtually identical to the originals, and they also look similar down to the byte level (e.g. same formatting options within the file, etc.). This meets the goals set forth at the beginning of this project.

Sample Screenshot

```
dfriedhe@okonokwo ~/seniorresearch $ ./a.out sample2.wav -i2 MESSAGE!
sample2.wav MESSAGE!
RIFFWAVE 424100 WAVE
stereo
sample bit rate 44100
data chunk reached
datasize: 423488
data read successfully
datsize confirmed
datlen423456
message written to output3.wav
dfriedhe@okonokwo ~/seniorresearch $ ./a.out output3.wav -x2
Message found:
MESSAGE!
```

[The message, “MESSAGE!” is written to sample2.wav in the first line. It is then extracted from the resulting file (output3.wav) in the subsequent running of the program]



[Image depicts basic structure of WAVE file and shows where the individual letters of the message are being inserted (data chunk)]

Larger Purpose

This and many other steganographic programs have many uses in the worlds of intelligence and espionage. By hiding a text message within a seemingly innocent audio file, an undercover agent could transmit data over an unsecured connection without a great risk of discovery.