# Authorship Attribution of Compiled Malware

## Abstract

In the mostly anonymous world of the Internet, hackers and cyber criminals may often perpetrate crimes with little to no traces leading back to them. Often this is done using malware that the criminal has created. However, researchers can often obtain copies of this malware, and identifying information may be present within these binaries, even if only unintentionally by the authors. Within this project, I expect to analyze various malware binaries using authorship attribution techniques to try to determine if there are features that can set malicious binaries apart from bening binaries, or families of malware from other malware. We expect that there will be some identifying feature set that can be taken advantage of to show distinct authorship between files.

## Introduction

Since the advent of the Internet, humanity has found many ways to use it to increase their productivity, comfort, and enjoyment. However, with all of these benefits we've reaped, some have dedicated their efforts towards undermining others by creating and distributing malware. Anti-virus company Symantec has claimed to have detected over a million unique pieces of malware as of 2008, and the amount of malware has only increased since then [ADD CITATION].

The fact that these digital attacks can originate from any computer on the globe, with plausible deniability for the owner of that computer, being able to attribute these attacks back to the people who created them would be a useful technique for law enforcement officers and security researchers, on par with fingerprinting and handwriting analysis for attributing physical crimes to criminals. The volume of these digital attacks has steadily increased each year, and thus, automated methods for associating these tools to authors must be created, as attribution of all of these attacks by individual malware researchers is infeasible.

## Background

### Authorship Attribution

The quantitative study of authorship is known as stylometrics. This field has been used in literary fields since 1887 in analyzing the works of Shakespeare [Mendenhall, 1887]. The field gained more traction as The Federalist Papers James Madison, Alexander Hamilton, and John Jay.

### N-gram Analysis

N-gram analysis!

### Source Code Analysis

%I'm not entirely sure where I should be putting this section.

%Maybe in Authorship Attribution, near the function word section?

%Maybe down by the n-gram opcode analysis? (too late, I think)

### Feature Selection

While the addition of more test data is often useful

## Methodology

### Datasets Used

### N-gram Analysis of Malware

From a general human perspective, a compiled binary (which is how most malware is seen) is essentially ciphertext.

%maybe mention something about how it's almost a 1:1 translation, as opposed to a block cipher or something that changes with previous data.%

There is an underlying structure that the machine can read, but it is not readily apparent to the naked eye. Most code attribution studies have used uncompiled source code

%cite several studies here

as their dataset, since it is the actual text that the author created. Compiling creates a layer of obfuscation between the author's original text and the data at hand. However, unlike a cipher text, this compiled data has a roughly deterministic relationship to the source code, and thus, can be used as if it were the author's original text with only a small loss in accuracy.

This compiled data is in the form of machine code, which lacks much of the stylistic choices such as names of functions and variables (though some of this data can still be recovered, as described later), but still retains the functional choices such as program organization and algorithm implementation.

### Opcode Frequency Distribution

Malware is the same as any other program on a computer, except for its malicious activity. This malicious activity might manifest in a different distribution of operations for the CPU to perform. For example, a piece of malware might have many calls to interrupt operations in order to disrupt program flow and obtain control of a privileged process. If it is concerned with cryptographically securing information before sending it back to some sort of controller, there might be an above-average amount of shift operations and operations to send data to external devices.

These opcodes are derived from the same dataset as above by simply taking a variable length subset of the bytes. We will use only the opcodes, not the registers and other arguments passed to the opcodes. We believe that this feature selection process will provide us with a more effective disambiguation of the data into their respective classes.

For ease of programming, we use the Unix objdump utility to obtain a textual representation of all of these opcodes from the code section of the programs. All programs are compiled for the same architecture (8086) so they all draw from the same set of CPU instructions.

### N-gram Analysis of Opcodes

## Results

## Conclusions

## Future Work

Future work in this field

## Acknowledgments

## Works Cited

Binongo, J. N. G. (2003). Who wrote the 15th book of Oz? An application of multivariate analysis to authorship attribution. Chance, 16(2), 9-17.

Burrows, S., Uitdenbogerd, A. L., & Turpin, A. (2009, January). Application of information retrieval techniques for source code authorship attribution. In Database Systems for Advanced Applications (pp. 699-713). Springer Berlin Heidelberg.

Frantzeskou, G., MacDonell, S., Stamatatos, E., & Gritzalis, S. (2008). Examining the significance of high-level programming features in source code author classification. Journal of Systems and Software, 81(3), 447-460.

Frantzeskou, G., Stamatatos, E., Gritzalis, S., & Katsikas, S. (2006, May). Effective identification of source code authors using byte-level information. In Proceedings of the 28th international conference on Software engineering (pp. 893-896). ACM.

Juola, P. (2006). Authorship attribution. Foundations and Trends in information Retrieval, 1(3), 233-334.

Mendenhall, T. C. (1887). The characteristic curves of composition. Science, (214S), 237-246.

Mosteller, F., & Wallace, D. L. (1963). Inference in an authorship problem: A comparative study of discrimination methods applied to the authorship of the disputed Federalist Papers. *Journal of the American Statistical Association*, *58*(302), 275-309.

Pektaü, A., Eriü, M., & Acarman, T. (2011, August). Proposal of n-gram based algorithm for malware classification. In SECURWARE 2011, The Fifth International Conference on Emerging Security Information, Systems and Technologies (pp. 14-18).

Stamatatos, E. (2009). A survey of modern authorship attribution methods. Journal of the American Society for information Science and Technology, 60(3), 538-556.

\*Add symantec internet secruity report April 2008.