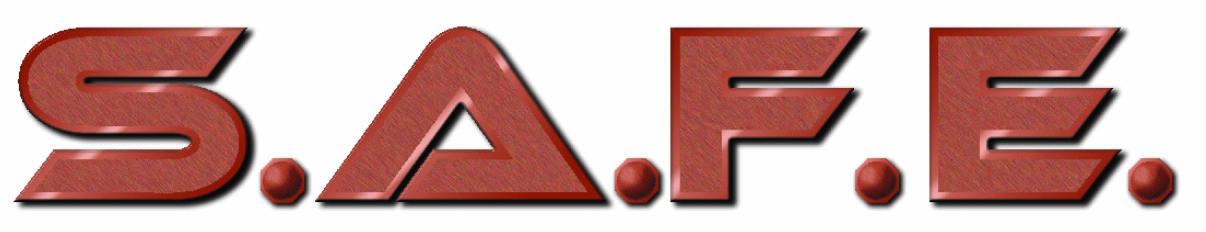
Detecting Software IP Theft Using CodeMatch



Software Analysis & Forensic Engineering Corporation
www.SAFE-corp.biz

1. Detecting Copyright Infringement

For detecting software copyright infringement, also known as software plagiarism, several tools are available from universities. CodeMatch®, part of the CodeSuite® suite of tools from SAFE Corporation for analyzing software, has been shown to have superior accuracy and superior reliability than those tools.

1.1 CodeMatch®

CodeSuite® is a tool that Bob Zeidman developed and spun off its development and sales to a company he founded called Software Analysis & Forensic Engineering Corporation. As the name implies, CodeSuite is a suite of tools for comparing and analyzing software source code, and object code, to find different kinds of intellectual property theft. For copyright infringement, there is an integrated tool called CodeMatch® that compares source code specifically to find possible copyright infringement. A fully functional copy of CodeSuite can be downloaded from the SAFE website at the URL www.SAFE-corp.biz where you will be asked to fill out a form to receive the download instructions by email. CodeMatch can be used for free on source code totaling less than 1 Mbyte, but larger amounts of code require a license to be purchased.

CodeMatch is the only available tool that measures every aspect of correlation including statement correlation, comment/string correlation, identifier correlation, and instruction sequence correlation. It also can produce detailed reports and statistics spreadsheets about the comparison results. There is also a feature to allow filtering of the results to eliminate files, folders, and code elements that are not interesting to the analysis.

1.2 Reasons for Correlation

Finding a correlation between different programs does not necessarily imply that illicit behavior occurred. There can be correlation between programs for a number of reasons as enumerated below.

- **Third-Party Source Code**. It is possible that widely available open source code is used in both programs. Also, libraries of source code can be purchased from third-party vendors. If two different programs use this same code, the programs will have correlation.
- **Code Generation Tools**. Automatic code generation tools generate software source code using similar or identical identifiers for variables, classes, methods, and properties. Also, the structure of the code generated by these tools tends to fit into a certain template with an identifiable pattern. Two different programs that were developed using the same code generation tool will have correlation.
- **Commonly Used Elements**. Certain identifier names are commonly taught in schools or commonly used by programmers in certain industries. For example, the identifier `result` is often used to hold the result of an operation. These identifiers will be found in many unrelated programs but will result in these programs having correlation. Less often but still occurring, some comments, strings, and statements are commonly used by programmers in certain industries or taught in schools. These comments, strings, and statements will be found in many unrelated programs but will result in these programs having correlation.
- **Common Algorithms**. An algorithm is a procedure or a set of instructions for accomplishing some task. In one programming language there may be an easy or well-understood way of writing a particular algorithm that most programmers use. For example there might be a way to calculate the square root of a number. Perhaps this algorithm is taught in most programming classes at universities or is found in a popular programming textbook. These commonly used algorithms will show up in many different programs, resulting in a high degree of correlation between the programs even though the programs are different and there was no direct contact between the programmers.
- **Common Author**. It is possible that one programmer, or “author,” will create two programs that have correlation simply because that programmer tends to write code in a certain way. This is the programmer’s style of coding. Thus two programs written by the same programmer can have high

correlation due to the style being similar even though there was no copying and the functionality of the programs is different.

- **Copying (Plagiarism, Copyright Infringement)**. Code was copied from one program to another causing the programs to have correlation. If the copying was unauthorized then the code was plagiarized (i.e., the copyright was infringed).

For someone attempting to find copyright infringement, there are a series of steps to go through to eliminate each of the other five reasons for correlation, as depicted in Figure 1. If the other five reasons can be eliminated, the correlation must be due to copying. If the copying was unauthorized then copyright infringement must have occurred.

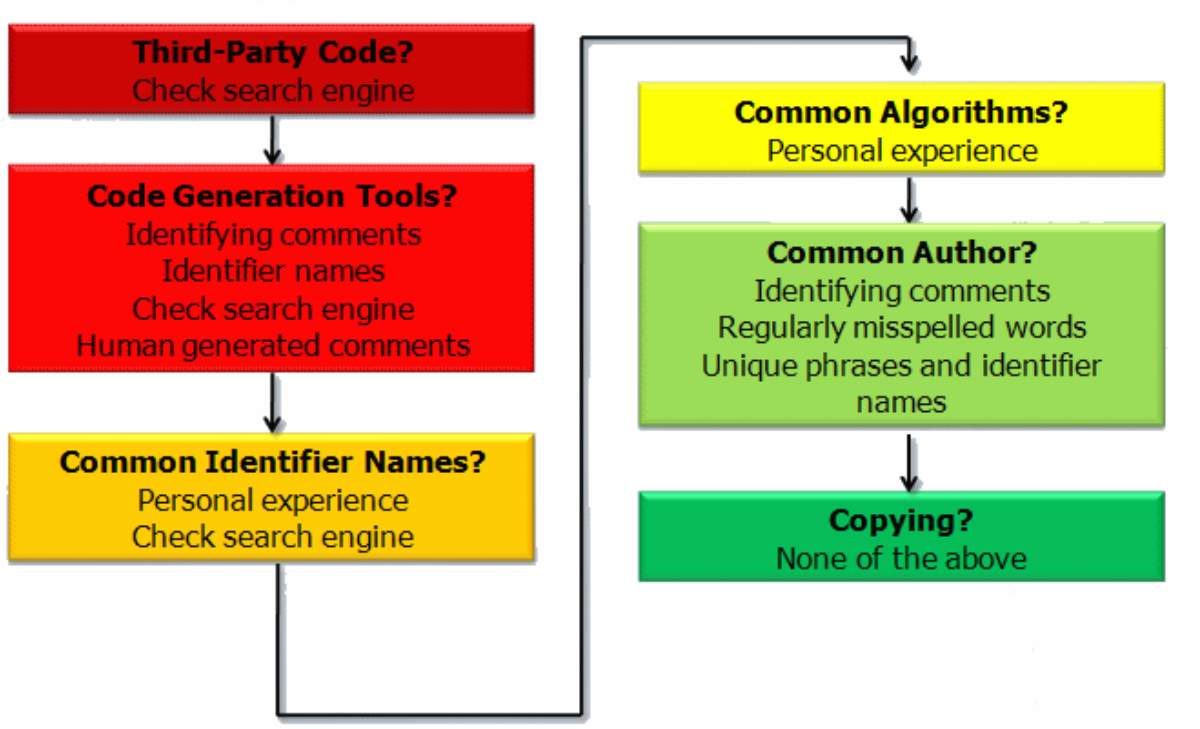


Figure 1. Steps for determining copying

2. Detecting Trade Secret Theft

There are a number of tools that are useful for finding trade secret theft. Any static analysis tool for reverse engineering software can be used to help find trade secret theft because the key task is to understand the functionality of the code in question. CodeSuite is useful for detecting certain kinds of trade secret theft if it also involved literal copying of code. In many cases when code is copied, trade secret theft charges are brought against a company because the penalties can be greater for trade secret theft rather than copyright infringement. For trade secret theft, the plaintiff needs to show the three aspects existed that are required for a trade secret – the code was not known to the public, the code was valuable to the owner, and the owner took reasonable steps to keep the code secret. The first and third aspects involve research beyond anything any software tool is capable of. The second aspect is fairly easy to show – if the code was not valuable why would anyone attempt to copy it and risk a lawsuit? The fact that the code was copied often, in itself, implies that the code was valuable. Often the first step in a software trade secret case is to determine whether any code was copied. In that case, CodeSuite is the best tool available to begin a determination of software trade secret theft.