DYNAMIC SOFTWARE WATERMARKING BY ALTERING THE NUMERIC RESULTS OF THE PROGRAM

Claudiu Chiru, Spiru Haret University,
Unirii Street nr. 32-34, Constanta, email: claudiu.chiru@seanet.ro

Abstract

Watermarking is a technique to embed a secret message into a cover message. The secret is usually a copyright message. Using this method, intellectual property can be protected against theft and can be used to prove ownership. In software watermarking, the secret is a copyright notice embedded into a program. Later, the watermark can be retrieved and allows to prove ownership. This paper presents a method of dynamic watermarking by altering the numeric results of the program.

Key words: software watermarking, set watermarking, metric

1. INTRODUCTION

Digital watermarking gained widespread popularity as a research topic in the latter half of the 1990's. Miller (Miller et al. 1999) defines digital watermarking as follows:
"A digital watermarking embeds an imperceptible signal into data such as audio, video, and images, for a variety of purposes, including captioning and copyright control".

The watermarking technique can be extended successfully over programs. In contrast with the watermarks of the multimedia documents, where the inserting of a watermark is based on the redundancy of a visualizing or auditing human system, in the case of the programs' marking, the restrictions are much greater. The programs must be equivalent from the functionality's point of view. Software watermarking (Colberg 1999) is the process of embedding a copyright message in a program.

Static watermarks (Colberg 1999) are stored in the executable program at certain locations and are constituted from octet sequences, which can be identified. Dynamic software watermarks (Colberg 1999) are stored in a program execution state rather than in the program code itself.

Starting from Radu Sion's paper (Radu Sion et al. 2001) this work presents a method of dynamic software watermarking using the numeric results of the program.

2. DISTORTING THE ELEMENTS OF A SET

Let M be the set of the numeric results of the program, $M = \{m_1, m_2, ..., m_n\}$ $M \subset \mathbf{R}$, $n \in \mathbf{N}$. Let W be the M watermarked set and $f_w : M \to W$ where f_w is a bijective function. The marking operation is done by inserting watermarking bits in subsets of M. A bit insertion corresponds to the distortion of at least one element from the set which is to be marked.

Distorting the elements of a set is made by modifying the value of an element such that the metrics are satisfied:

1. $\left| m_i - f_w(m_i) \right| < \varepsilon_1$, $m_i \in M_i$, $f_w(m_i) \in W_i, \varepsilon_1 \in \mathbf{R}$ - acceptable distortion for an element
2. $\sum_{i=1}^{|M_i|} (m_i - f_w(m_i))^2 < \varepsilon_2$, $m_i \in M_i$, $f_w(m_i) \in W_i, \varepsilon_2 \in \mathbf{R}$ - acceptable distortion for a set.

3. INSERTING THE WATERMARK

Before partitioning the set, the elements will be ordered according to a key, which is known only to the watermarking/dewatermarking process. The watermarking process depends on the value of the individual elements of the set.

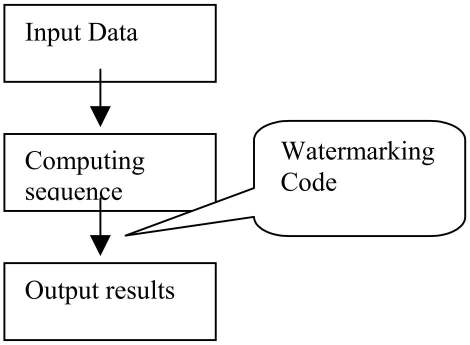

Figure 1. Inserting the watermarking sequence in a program

The ordering function is:
$index : \mathbf{Z} \times M \to \mathbf{N}$ where
$index(key, m_i) = g(m_i)^{key} \bmod r$ is a one way function.
key : a secret value (512 bits)
r : a secret value(1024 bits)

$g(m_i) = MSB_q(|10^p \cdot m_i|)$ where p is the number of decimal places and MSB_q (m) the most significant q bits of the number m. $g(m_i)$ is concatenated with a random value and 0 bits such that: $n\_bit(g(m_i)) + n\_bit(m_r) + n\_bit(0) = 512$ In the end we will have a set of pairs $(index(m_i), m_i))$. The m_i values will be sorted using the index values.

3.1. Partitioning M

After the ordering phase, the M set will be partitioned such that: $\bigcap_{i=1}^{S} M_i = \Phi$, $\bigcup_{i=1}^{S} M_i = M$, $\sum_{i=1}^{S} |M_i| = n$. S is the number of partitions, n the number of components.

3.2. Watermark Insertion

A subset is considered to be watermarked with a 0 value if $0 < \sum_{i=1}^{|M_i|} (m_i - w_i)^2 < t_1$, $t_1 \in \mathbf{R}$.

A subset is considered to be watermarked with a 1 value if $t_2 < \sum_{i=1}^{|M_i|} (m_i - w_i)^2 < \varepsilon_2$, $t_2 \in \mathbf{R}$, $0 < t_1 < t_2 < \varepsilon_2$. The subset elements will be distorted by adding small values, smaller than ε such that 1 and 2 are true.

4. EXTRACTING THE WATERMARK

The watermarking system has some constant values such as: $t_1, t_2, \varepsilon_1, \varepsilon_2, r, key, m_r$. q and p are calculated considering the set which is to be watermarked. Extracting the watermark requires the original unwatermarked resulting set. For the watermark extraction process one needs the original program without the marking block.

Steps for extracting the watermark:

1. The original program is run with a test data set. The results are:
$M = \{m_1, m_2, ..., m_n\}, M \subset \mathbf{R}$, $n \in \mathbf{N}$.
2. The program with the marking block is run. The results are:
$W = \{w_1, w_2, ..., w_n\}, W \subset \mathbf{R}, n \in \mathbf{N}$
3. Ordering both sets using $g(m_i)$ function
4. Partitioning the set *W* and *M* after ordering
5. Extracting the watermark using criteria 1 and 2

5. ATTACKS AGAINST THE WATER-MARKING SYSTEM

1. Eliminating the marking block out of the program. Eliminating the marking block out of the program is a difficult operation that requires the exact position of the marking block. Spreading the watermark block over the program can confuse attackers.

2. Distorting the results such that the watermark cannot be extracted
a) Adding or removing data. This is not a strong attack because the marking system uses the original unwatermarked data.
b) Altering the values of some elements by identifying the distorted elements and eliminating the distortions or altering all the elements. This is a very strong attack that cannot be defeated.

6. CONCLUSIONS

This paper presents a dynamic method of watermarking software, by marking of the resulting data of the program. The method proposed can be used when the user accepts a small distortion in the numeric results.

The numeric set which is to be watermarked can not be too large because the watermark extracting process uses the original unmodified data. There are some attacks against this system and the most important is “altering all the elements”.

7. REFERENCES

(Colberg 1999) Christian Collberg - Software watermarking: Models and Dynamic Embeddings POPL ’99 , Proceedings

(Miller et al. 1999) Miller et al. - Watermarking as Communications with Side Information “ Proceedings of the IEEE, 87(7), 1999

(Radu Sion et al. 2001) Radu Sion, Mikhail Atallah, Sunil Prabhakar “On Watermarking Numeric Sets”, Purdue University, 2001