Digital watermarking and cryptography

Fabien A. P. Petitcolas

Microsoft Research

fabienpe@microsoft.com

1 Introduction

As audio, video and other works become available in digital form, the ease with which perfect copies can be made, may lead to large-scale unauthorized copying which might undermine the music, film, book and software publishing industries. These concerns over protecting copyright have triggered significant research to find ways to hide copyright messages and serial numbers into digital media; the idea is that the latter can help to identify copyright violators and the former to prosecute them.

At the same time, moves by various governments to restrict the availability of encryption services have motivated people to study methods by which private messages can be embedded in seemingly innocuous cover messages.

Techniques for information hiding related to computer systems appear in many other areas, some of them of interested to cryptographers:

Covert channels have been defined by Lampson in the context of multilevel secure systems (e.g., military computer systems), as communication paths that were neither designed nor intended to transfer information at all. These channels are typically used by untrustworthy programs to leak information to their owner while performing a service for another program. These communication channels have been studied at length in the past to find ways to confine such programs and were revisited recently in the context of downgrading of images.

Anonymity is finding ways to hide the metacontent of messages, that is, the sender and the recipients of a message. Early examples include anonymous remailers.

An important sub-discipline of information hiding is *steganography*. While cryptography is about protecting the content of messages, steganography is about concealing their very existence. This modern adaptation of *steganographia* (Trithemius, 1462–1516), assumed from Greek στεαυ-ό, ςγραφ-ειυ, literally means 'covered writing' and is usually interpreted to mean hiding information in other information. Examples include sending a message to a spy by marking certain letters in a newspaper using invisible ink and adding sub-perceptible echo at certain places in an audio recording.

Watermarking, as opposed to steganography, has the additional requirement of robustness against possible attacks. In this context, the term 'robustness' is still not very clear; it mainly depends on the application, but a successful attack will simply try to make the mark undetectable.

Another fundamental difference between steganography and watermarking is that the information hidden by a watermarking system is always associated to the digital object to be protected or to its owner while steganographic systems just hide any information. This has of course some consequences on the type of attacks that can be achieved. The 'robustness' criteria are also different, since steganography is mainly concerned with detection of the hidden message while watermarking concerns potential removal by a pirate. Finally, steganographic communications are usually point-to-point (between sender and receiver) while watermarking techniques are usually one-to-many.

A number of applications of information hiding have been proposed in the context of multimedia applications. In many cases they can use techniques already developed for copyright marking directly; in others, they can use adapted schemes or shed interesting light on technical issues. They include the following:

- *Automatic monitoring* of copyrighted material on the Web: A robot searches the Web for marked material and hence identifies potential illegal usage. An alternative technique downloads content from the Internet, computes a digest of it and compares this digest with digests registered in its database.
- *Automatic audit of radio transmissions*: A computer can 'listen' to a radio station and look for marks, which indicate that a particular piece of music, or advertisement, has been broadcast.
- *Data augmentation*: Information is added for the benefit of the public. This can be details about the work, annotations, other channels, or purchasing information (nearest shop, price, producer, etc.) so that someone listening to the radio in a car could simply press a button to order the compact disc. This can also be hidden information used to index pictures or music tracks in order to provide more efficient retrieval from databases.
- *Tamper proofing*: The information hidden in a digital object may be a signed 'summary' of it, which can be used to prevent or to detect unauthorized modifications.

There are several wisdoms of the old and well researched discipline of cryptography we can borrow and try to apply to information hiding. On of the most obvious are the Kerckhoffs' principles of cryptographic engineering, in which he advises that we assume the method used to encipher data is known to the opponent, so security must lie only in the choice of key. The history of cryptology since then has repeatedly shown the folly of 'security-by-obscurity' – the assumption that the enemy will remain ignorant of the system in use.

Applying this wisdom, we obtain a tentative definition of a secure stego-system: one where an opponent who understands the system, but does not know the key, can obtain no evidence (or even grounds for suspicion) that a communication has taken place. It will remain a central principle that steganographic or watermarking processes intended for wide use should be published, just like commercial cryptographic algorithms and protocols.

2 Cryptography for information hiding

2.1 Functional analogies

Embedding and extraction of hidden information can sometimes be considered analogous to encryption and decryption. In fact this becomes obvious if one looks at them from a purely functional point of view:

In the case of symmetric encryption, a message m is encrypted under a secret key k using some encryption function E:

$m_c = E(k, m)$.

Decryption uses the secret key to reveal the original content:

$m = D(k, m_c)$.

In steganography or watermarking, we have an embedding function E that takes some original stego-objet o, a message m and a secret stego-key k as inputs and outputs a new object $\tilde{o}$ which contains the message:

$\tilde{o} = \mathrm{E}(o, k, m)$.

The extraction of the hidden message is done from $\tilde{o}$ or $\tilde{o}'$, a possibly slightly distorted version of $\tilde{o}$:

$m = \Delta(\tilde{o}', k)$.

With such analogy is not unreasonable to believe that some steganographic or watermarking problems can be solved using appropriate cryptographic techniques. But we will see that there are problems specific to steganography and watermarking.

2.2 Categories of attacks

In this section we assume that the attacker has detailed knowledge about the hiding algorithm, unless specified otherwise. Attacks and analysis on hidden information may take several forms and any cover can be manipulated with the intent of detecting, extracting, counterfeiting, overwriting or disabling hidden information. So the granularity of the attacks is much finer in the case of watermarking or steganography than in cryptography where the goal of the attacker is to get the plaintext.

Detection of the existence of a hidden message in some content, extraction of this message or modification and removal of this message, all constitute successful attacks of a steganographic system. In the case of watermarking the attacks are relatively simpler because the attacker usual know whether the content has been watermarked or not. Thus, he just need to fiddle with the content such that the mark become unreadle or undetectable. Alternatively he can also try to insert a new mark such that both a detectable – this is usually referred to as the 'dead lock' attack. He could also try to completely remove the mark from the stego-object.

As cryptanalysis has different level of attacks such as ciphertext-only, known plaintext, chosen plaintext or chosen ciphertext, steganalysis comes with its own, but very similar, range of attacks:

- *Stego-only attack* – Only the stego-object is available for analysis. When such an attack becomes possible the embedding scheme does not provide anymore guaranties and should not be used.
- *Known cover attack* – The 'original' cover-object and stego-object are both available.
- *Known message attack* – At some point, the hidden message may become known to the attacker. Analyzing the stego-object for patterns that correspond to the hidden message may be beneficial for future attacks against that system. Even with the message, this may be very difficult and may even be considered equivalent to the stego-only attack.
- *Chosen stego attack* – The steganography tool (algorithm) and stego-object are known.
- *Chosen message attack* –The steganalyst generates a stego-object from some steganography tool or algorithm from a chosen message. The goal in this attack is to determine corresponding patterns in the stego-object that may point to the use of specific steganography tools or algorithms.
- *Known stego attack* – The steganography algorithm (tool) is known and both the original and stego-objects are available.

One possible characterisation of a secure steganography system is that the attacker cannot gain any information about m or E by comparing both stego- and cover-objects so the mutual information is 0:

$\mathrm{H}(m) - \mathrm{H}(m \mid o, \tilde{o}) = 0$.

Consequently steganography and watermarking to a certain extent, bring some amount of confidentiality. In practice however, due to the constraints of imperceptibility the key space offered by the scheme is not very large and is prone to brute force attack. So one simple way to tackle the problem or unauthorised extraction of the watermark is to rely on existing cryptographic algorithms: m is first encrypted and the hidden:

$\tilde{o} = \mathrm{E}(o, k_s, E(k_c, m))$ and

$m = D(k_c, \Delta(\tilde{o}', k_s))$.

3 Example of ‘cryptographic’ attacks

3.1 Brute force attack

Using the model described above one can easily draw a simple brute force attack on the schemes: given the embedding and extraction function of the scheme, try all possible values of k until the message is detected.

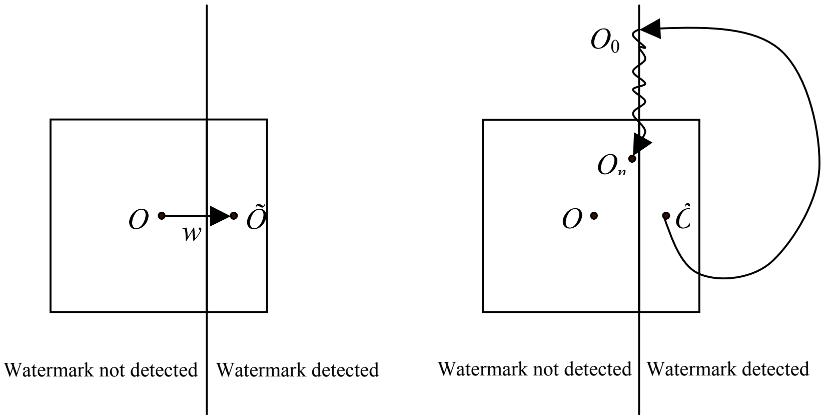

Figure 1 – The *oracle attack* is an example of brute force attack on watermarking schemes whose public detector is available to the attacker. Typically the detector could be embedded in a consumer electronics device. Each time the detector is queried the attacker gets some information which can be used to reconstruct an un-marked object.

Another brute-force attack relies on the fact that in most applications the attacker has access to a detector. This detector can be a piece of software shipped with a major image processing package or an electronic circuit embedded into consumer electronics such as DVD. Even if the attacker does not know much about the watermark embedding method, he can still use the information returned by the detector to remove the watermark by applying small changes to the image until the decoder cannot find it anymore.

The attacker starts by constructing an image that is very close to the decision threshold of the detector: modifying this image very slightly should make the detector switch from ‘watermark present’ to ‘watermark absent’ with probability close to 0.5. Note that the constructed image does not need to resemble to the original. This can be achieved by slightly blurring repeatedly the image until the detector fails to find a watermark or by replacing progressively pixels by grey.

The second step analyzes the sensitivity of the detector to modification of each pixel. The luminance of some given pixels is increased or decreased until the detector changes its output. This is repeated several times (e.g., 10^4 in practice). From this analysis the attacker can devise a combination of pixels and modifications such that the distortions in the image are minimized and the effect of the modifications on the detector are maximized, that is that the watermark is not detected.

Unless a breakthrough is made (e.g., implementation of a reliable asymmetric scheme), applications that require the public verifiability of a mark (such as DVD) appear doomed to operate within the constraints of the available tamper-resistance technology, or to use a central ‘mark reading’ service.

3.2 Dead lock attack

Many private watermarks survive the insertion of a second mark, since the mark is stored in a manner or a location which is kept secret, with enough ‘room’ for marks that an attacker must inflict an unreasonable amount of damage to the content in order to lay waste to every possible hiding place within. One may be tempted to ask then, if one can simply add a second watermark and claim ownership of marked content. The reason this fails is that the original content creator has a piece of truly original content, hopefully hidden away from attackers. This original contains no watermarks, whereas the attacker's supposed original contains the first watermark and not the second, clearly establishing an order of insertion.

In general, a dispute over multiple watermarks could be resolved by each party searching for his watermark within the other's original. It would certainly be very strange if each original was a watermarked version of the other original! If appropriately symmetric in terms of watermark strength, this would prevent either party from establishing ownership, obliterating the effectiveness of the watermark. It turns out that this very situation can be engineered for some watermarking methods by a clever attacker.

Ideally, the original content creator would add a mark w to an object $\tilde{o}$, yielding a marked object $\tilde{o} = o + w$. This object $\tilde{o}$ is distributed to customers and when a suspect object $\tilde{o}'$ is found, the difference $\tilde{o}' - o$ is computed, which should equal w if $\tilde{o}'$ and $\tilde{o}$ are the same and which would be very close to w if $\tilde{o}'$ was derived from $\tilde{o}$ and the scheme robust. We assume that a correlation function $\mathrm{c}(w, x)$ is used to determine the similarity between the original watermark w and the extracted datum x.

Mallory, hoping to steal the object for himself *subtracts*, rather than adds, a second watermark x to get an object $o' = \tilde{o} - x = \mathrm{o} + w - x$. Now, Mallory claims o' rather than $\tilde{o}$ to be his original image and hauls Alice into court for violating *his* copyright. When originals are compared, Alice will find that her mark w is present in Mallory's object o':

$o' - o = w - x, \qquad \mathrm{c}(w - x, w) = 1$

Since two marks can survive in the same image, the subtraction of x should not greatly hurt the presence of w. So, Mallory has not successfully removed Alice's mark. However, Mallory can show:

$\mathrm{o} - o' = x - w, \qquad \mathrm{c}(x - w, x) = 1$

In other words, Mallory's mark is present in Alice's original image, despite the fact that Alice has kept it locked away. Empirical data collected by Craver et al. for the watermarking scheme described by Cox shows that this attack works and that the relative strengths of the two watermarks are virtually identical. There is no real evidence that either party was the image's originator.

This attack works by subtracting a mark rather than adding one and so relies on the invertibility of a watermarking scheme. A good way to fix this is to make the watermark insertion method a one-way function H of the original image. In these noninvertible schemes, it is practically impossible for Mallory to subtract his mark x, for $x = h(o')$ could not be computed until o' is known and o' could not be computed from $\tilde{o}$ until x is known. As long as h is difficult to invert, o' is difficult to compute.

3.3 Collusion attacks on watermarking

As in cryptography the re-use of material can sometime lead to devastating attacks (e.g., reuse of a one-time pad), reuse of the cover-signal or of the watermark leads to collusion attacks. Use of the same watermark in different content allows the attacker to estimate the watermark and hence remove it more easily. Marking the same content with different watermarks (as it is the case for fingerprinting) can be used to estimate a un-watermarked content or generate a new content with a watermark that was not used before.

This has severe implications because most multimedia content has a lot of redundancy within it. For instance, musical recordings often contain repetitions, which, if not taken into account properly, could be used by an attacker to defeat the watermark detector. This can be achieved by swapping blocks within the signal or indeed between different sound tracks. This has the same effect as the collusion version of the mosaïc attack, where by fingerprinted images can be shopped into sub-images and a new image can be reconstructed using sub-images of different images.

4 Concluding remarks

In this manuscript we have tried to emphasise certain links – some obvious, other less – between cryptography and steganography and watermarking. Existing cryptographic techniques are rarely directly usable in information hiding but cryptography can clearly be used in complement of watermarking. In other words many security requirements can be met by incorporating cryptographic tools to ensure the integrity of the hidden message. However, preventing unauthorised detection of watermarks without decoding the mark cannot yet be achieved.

About the author

Fabien A. P. Petitcolas received his Ph.D. on information hiding and its application to copyright protection from the University of Cambridge, England. He is currently with Microsoft Research and his research interests include computer security, robustness, testing and evaluation of digital watermarking and steganography algorithm. He is the editor of the first book on information hiding and digital watermarking (http://www.cl.cam.ac.uk/~fapp2/publications/book99-ih/) and is involved in several conferences on the topic.