Computational Security Models for Digital Watermarks (Extended Abstract)

Stefan Katzenbeisser

Institut für Informatik
Technische Universität München
D–85748 Garching, Germany
skatzenbeisser@acm.org

ABSTRACT

In this paper, we review common information-theoretic security models for watermarking schemes and argue that they have some deficits. In particular, they do not consider the limited computing power of any practical attacker. To resolve this issue, we propose to adapt computational security definitions, as commonly used in cryptography, to the watermarking scenario. In this paper, we give two different computational security notions for digital watermarks—one for attacks that aim at estimating the watermarking key and one for oracle attacks.

1. INTRODUCTION

During the last few years, watermarking has become an accepted technology in various application areas—from Digital Rights Management systems over digital identification technologies to content labelling. Despite the evidential practical success of watermarking, few theoretical papers exist that lay the foundation for watermarking technology. In particular, there is still no consensus in the scientific community what can be considered a *secure* watermarking system. Up to now there is no clear distinction between *robustness* and *security*; both words were commonly used interchangeably. Recent works started to make the boundary line between these terms clear; for example, Kalker [3] defined security as the "inability by unauthorized users to have access to the raw watermarking channel", where access refers to "remove, detect, estimate, write and modify the raw watermark bits". In such a definition, security clearly is a more general term than robustness (and particularly captures *malicious* attacks).

Starting with [4], information-theoretic models for the security of a watermarking scheme became popular. In an information-theoretic approach, the covers, watermarks, keys and watermarked objects are seen as random variables (see Figure 1). Security and Robustness are quantified by the mutual information between these variables; in the particular approach of [4], security is measured by $I(\mathbf{M}, \mathbf{Y})$, i.e., the by information that can be obtained on $\mathbf{M}$ when observing $\mathbf{Y}$. If $I(\mathbf{M}, \mathbf{Y}) = 0$, we speak of a watermarking scheme that achieves *perfect secrecy*. Robustness may be modeled by the mutual information $I(\mathbf{M}, \mathbf{Z} \,|\, \mathbf{K})$, where $\mathbf{Z}$ denotes a random variable representing attacked objects. Other approaches (such as [1] and [5]) used other measures of information, namely the Fisher Information Matrix or a translation of Shannon's perfect secrecy [6] to continuous variables.

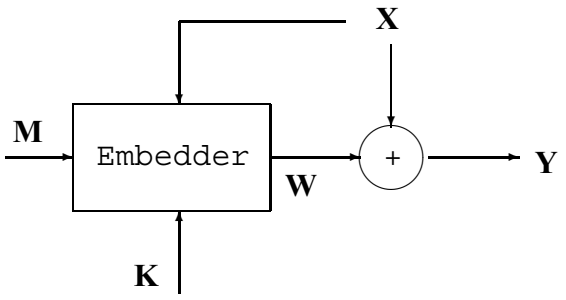

Figure 1: *Watermark embedding model: the embedder takes the raw watermark message **M**, a key **K** and possibly the cover **X** and produces a watermark **W** which is eventually embedded into **X** to obtain the watermarked object **Y**.*

Although it was possible to quantify the security of several practically relevant watermarking schemes using information-theoretical models under approximations of the channel distribution, they have some apparent drawbacks:

- First, these definitions do not consider the attacker's computing power, but rather give an upper bound for the amount of information any successful attacker may get from observing (possibly

many instances of) watermarked objects. However, even if this amount of information is large there is still the possibility that computing it will be exceptionally hard and thus prohibitive for any practical attacker. In some sense, information-theoretic models capture the "worst case" for the watermark designer.

- Perfectly secure watermarking schemes (in the information-theoretic sense) may be difficult to construct. In cryptography, we know that perfect secrecy (in the sense defined by Shannon) is almost impossible to achieve, as this would require a secret key that is at least as long as the secret message that needs to be transmitted. The reason for this result is that the attacker may consume an arbitrary amount of time in Shannon's model. In the meantime, cryptography switched to a more liberal security model which takes into account the limited computational resources any practical attacker can use. They define security in terms of success probabilities for a (randomized) polynomial attacker.

- Another serious issue arises with information-theoretic security models. Say, we are able to prove that for a given watermarking scheme we have $I(\mathbf{M}, \mathbf{Y}) < \varepsilon$ for some function ε. How can we evaluate whether this specific watermarking scheme is "secure enough" for a particular application; in other words, which security bounds do we have to achieve so that we can argue that the watermarking scheme is *practically* secure?

- In addition, it may be difficult to judge the security of a compound scheme that uses both watermarks and cryptographic primitives. As the security of the latter is normally defined only in a computational framework, it is not evident how the computational security guarantees of the cryptographic primitives integrate into an information-theoretic framework.

In this paper, we review possible alternative security definitions that do not rely on information-theoretic models, but rather follow a computational approach that is commonly adapted in cryptology.

2. COMPUTATIONAL SECURITY MODELS

In cryptography, it is common to evaluate the security of cryptographic primitives in a computational model. An attacker A is modeled by a probabilistic polynomial-time Turing machine that tries to break the primitive in question. In particular, the machine can make random decisions during its operation; these random decisions can be viewed as being triggered by random coin flips. The sequence of coin flips during one possible sequence of operations will be encoded as random string $r \in \{0,1\}^*$; the set of all possible coin flips will be denoted by $R \subset \{0,1\}^*$. Note that R is necessarily finite, as we only consider machines whose computations halt after at most a polynomial number of steps (polynomial in the length of the input).

The success of one such attacker A can be measured by the success probability s of the machine, that is the probability that the machine A correctly breaks a random instance of the cryptographic primitive in question by a (random) computation sequence. The probability is taken over all problem instances and possible random computations (all possible coin flip sequences); s can be written as

$$s = \sum_{r \in R} \mathbf{P}[A \text{ succeeds} \,|\, r]\, \mathbf{P}[r].$$

Note that s will normally not be zero, as the attacker A may be able to "guess" the correct secrets of the cryptographic primitives (such as symmetric keys). However, this is a rare event (for naive guessing of the key secrets, the probability of a successful run on the machine A would decrease exponentially if the key length is increased). In fact, the probability s can be used as a measure for the security of a cryptographic primitive: if for all possible attacker machines A, s converges exponentially to zero by increasing the key length, there are no other attack possibilities than a naive guess. If, however, s converges slower, then efficient attacks are possible.

In the computational setting, a cryptographic scheme is said to be secure, if *all* such attackers A only have a success probability s that rapidly converges to zero as the length of the secret keys increases. Formally, we require that s is *negligible* in the sense that s can be bounded from above by the quotient of *every* polynomial $p()$:

$$s \leq \frac{1}{p(|K|)},$$

where $|K|$ denotes the length of the secret keys involved. In other words, *each* possible attacker A should only be able to break the primitive under very unfortunate conditions. Note that this notion is stable under polynomial repetition of the attacker machine; if an attack A with negligible probability is performed polynomially often, the success probability of the iterated attacker is still negligible. In the security setting, this means that the attacker cannot considerably enhance his chances by repeating the attack independently and polynomially often.

In the following sections we sketch ways how to apply computational security models to digital watermarking schemes.

3. WATERMARK SECURITY

In this section, we propose a computational model for an attacker that wants to obtain information about the watermarking key that was used to watermark an object. In an information-theoretical setting, we would quantify this information by the expression $I(\mathbf{K}, \mathbf{Y})$. Here, we model the goal of "obtaining information about the key" by the ability to distinguish whether a given watermarked object was more likely watermarked with one out of two keys.

Formally, we define watermarking security through the following two-part game between an attacker and a (trusted) judge. The first part of the game consists of information gathering operations, whereas the second part has the form of a challenge:

1. A judge generates two watermarking keys $K_1, K_2 \in \mathbf{K}$ of length n and gives the attacker access to two oracles $\mathcal{O}_{K_1}$ and $\mathcal{O}_{K_2}$, which implement the watermark embedder with keys K_1 and K_2 respectively. These oracles can be used by the attacker to watermark objects of his own choice. (Note that we cannot make the keys K_1 and K_2 directly available to the attacker, as we assume a symmetric watermarking scheme).

2. In the first part, the attacker (adaptively) produces test documents $X_1, X_2, \ldots \in \mathbf{X}$ and obtains watermarked versions $Y_1, Y_2, \ldots \in \mathbf{Y}$ by using the oracles $\mathcal{O}_{K_1}$ and $\mathcal{O}_{K_2}$. The attacker is free to perform probabilistic polynomial time computations with the objects $Y_1, Y_2, \ldots$ obtained.

3. When the attacker has finished the information gathering process, the judge flips a coin $b \in \{0, 1\}$ and produces a watermarked object Y_c with key K_b. The judge hands over Y_c to the attacker.

4. During the second part, the attacker (who has no oracle access any more) has to determine whether Y_c was produced with key K_0 or key K_1, i.e., the attacker has to guess the bit b chosen randomly by the judge. Again, the attacker is allowed to perform probabilistic polynomial operations.

The *advantage* for the attacker to win the game can be used to assess the security of the watermarking scheme. We define the advantage for the attacker as the probability of a correct guess in step 4 minus $1/2$. This advantage measures the systematic chance of an attacker to distinguish whether a watermarked object Y_c was watermarked with key K_1 or K_2 (note that an attacker can always make a completely random choice and succeed in winning the game with probability $1/2$). A watermarking scheme that perfectly hides every information about the key would have an attacker advantage close to zero; ideally, the advantage should be negligible in the length of the watermarking key.

Note that the attacker cannot choose the document Y_c sent to him during the challenge. We get a slightly stronger notion of security, if we allow chosen plaintext attacks. In this setup, the attacker can choose the test document; for this definition, we replace the step 3 above with the following version:

3′. When the attacker has finished the information gathering process, the attacker computes a test document X (different from all previous oracle queries $X_1, X_2, \ldots$) and hands X over to the judge. The judge flips a coin $b \in \{0, 1\}$ and watermarks the object X with key K_b to obtain the watermarked object Y_c. The judge hands over Y_c to the attacker.

Note that in this setup it is necessary that the attacker does *not* have oracle access during the challenge. Otherwise, an attacker could trivially win the challenge for any deterministic watermarking scheme by feeding the document X into both oracles $\mathcal{O}_{K_1}$ and $\mathcal{O}_{K_2}$ and comparing their answers with the test document Y_c.

Both security definitions closely resemble the accepted security notions for symmetric cryptography [2].

4. ORACLE ATTACKS

Note that in the above security definition, the attacker has access to an oracle that implements the watermark embedding algorithms. In contrast, attacks that utilize the presence of a watermark detector are called oracle attacks. During the attack, the attacker is able to query the detection oracle on intentionally modified test documents in order to obtain knowledge about the secret key in use. We model such knowledge by a polynomially computable predicate $P : \{0, 1\}^* \to \{0, 1\}$ that maps a watermarking key to a bit. That is, we model "knowledge" by the ability to tell whether a watermarking key has a special (binary) property or not; for example, properties such as "the least significant bit of the key equals 1" or "the value of the key, interpreted as a binary number, exceeds the value of 2^{20}" can easily be modeled by appropriate predicates.

This time, the security definition is given in terms of two games. In the "real world game" the attacker has

to compute the property $P(K)$ of a watermarking key. For this purpose, he has access to a watermark detection oracle containing key K; he can use the oracle to detect whether a watermark is present in a test document under the key K. In the "ideal world", the attacker does not get any information on the key other than the key is chosen randomly from the key space. Again, the goal of the attacker is to compute $P(K)$.

Real World Game:

1. A judge generates a watermarking key K of length n and a watermark W. Finally, the judge gives the attacker access to a watermark detection oracle $\mathcal{O}_K$; this oracle returns 1 if and only if W is detectable in an input object.
2. The attacker repeatedly generates test documents $Y_1, Y_2, \ldots$; he can use the oracle to test the presence of K in these objects. At the end of his (polynomially bounded) computation, the attacker outputs a bit b. The attacker wins the game, if $b = P(K)$, i.e., if the attacker correctly guesses the property $P(K)$ of the watermarking key in use.

In contrast, in the ideal world game, the attacker guesses the value of $P(K)$ directly without access to K (this is necessary to account for properties that do not occur exactly with a probability of $1/2$ for a randomly chosen watermark key).

Ideal World Game:

1. A judge generates a watermarking key K of length n, but keeps the key secret and asks the attacker to guess $P(K)$.
2. The attacker does probabilistic polynomial computations and outputs a bit b. The attacker wins if $b = P(K)$.

We say that the watermarking scheme is computationally secure against oracle attacks, if the success probability for an attacker in the real world game is almost equal to the success probability in the real world game for every polynomially computable predicate P. Under this condition, the watermark detection oracle gives the attacker no additional information on the key other than that he can readily compute out of inspection of the keyspace alone.

5. CONCLUSIONS

In this paper, we have argued that information-theoretic security definitions for watermarking schemes have some deficits: among others, they do not consider the computing power of an attacker. These problems could be avoided if one adapts computationally security notions from cryptography. In this setup, we have provided computational security definitions for attacks against the secrecy of the watermarking keys and for oracle attacks.

Although it can be quite difficult to measure the security of an existing watermarking scheme with respect to the models proposed in this paper, the models can be useful in the design of new watermarking schemes. First, new designs for watermarks should be evaluated with respect to the attack models described in this paper. Even if this process gives no formal security guarantee, it can be used to exclude important classes of attacks. Second, the models can yield to provably secure watermarking schemes. We leave this for future research.

Acknowledgement. The work described in this paper has been supported in part by the European Commission through the IST Programme under Contract IST-2002-507932 ECRYPT. The information in this document reflects only the author's views, is provided as is and no guarantee or warranty is given that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability.

6. REFERENCES

[1] F. Cayre, C. Fontaine, and T. Furon. Watermarking security: theory and practice. *IEEE Transactions on Signal Processing*, 2005. to appear.

[2] Oded Goldreich. *Foundations of Cryptography, Volume II Basic Applications*. Cambridge University Press, 2004.

[3] T. Kalker. Considerations on watermark security. In *IEEE International Workshop on Multimedia Signal Processing (MMSP'01)*, pages 201–206, 2001.

[4] T. Mittelholzer. An information-theoretic approach to steganography and watermarking. In *3rd Workshop on Information Hiding (IHW'98)*, volume 1768 of *Lecture Notes in Computer Science*, pages 1–17. Springer, 1998.

[5] ECRYPT Network of Excellence in Cryptology. First summary report on fundamentals. Technical report, report D.WVL.1, 2005.

[6] C. Shannon. Communication theory of secrecy systems. *Bell Systems Technical Journal*, 28:656–715, 1949.