PRACTICAL SECURITY OF NON-INVERTIBLE WATERMARKING SCHEMES

Qiming Li and Nasir Memon

Department of Computer and Information Science
Polytechnic University, Brooklyn, NY 11201
qiming.li@ieee.org memon@poly.edu

ABSTRACT

Designing secure digital watermarking schemes resistant to invertibility attacks (or more generally, ambiguity attacks) has been challenging. In a recent work, Li and Chang (IHW'04) give the first stand-alone provably secure non-invertible spread-spectrum watermarking scheme based on cryptographically secure pseudo-random generators. Despite its provable security, there are certain constraints on the security parameters that require further analysis in practice, where it is more important to analyze the exact security instead of theoretical asymptotic bounds. In this paper, we consider a security notion that is slightly weaker theoretically but still reasonable in practice, and show that with this alternative security notion, the exact requirements on the parameters can be analyzed, and such analysis can be used to guide flexible implementations of similar schemes in practice.

***Index Terms*—** Digital watermarking, security, non-invertible watermarking, ambiguity attacks

1. INTRODUCTION

Digital watermarking schemes have been proposed to solve ownership disputes, where the owner (Alice) of a digital work $\widetilde{I}$ proves the ownership by producing an original unmarked work I_A and a watermark W_A that can be detected in $\widetilde{I}$. In this scenario, an important requirement on the watermarking scheme is that it has to be resistant to invertibility (or ambiguity) attacks. That is, it needs to be *non-invertible*. Under invertibility attacks (as first studied by Craver et al. [1]), an attacker (Bob) creates an ambiguity about the ownership by finding a fake watermark W_B that can also be detectable in the same digital work $\widetilde{I}$, and a fake original I_B. A more general type of attacks, namely *ambiguity attacks*, is discussed later in [2, 3], where the attacker is not required to generate a fake original. In either case, the key to prevent the attacks is to make it hard to find a forged watermark that can be detected in the distributed work $\widetilde{I}$.

There have been some heuristic based approaches to tackle ambiguity attacks, such as [1, 4, 5]. However, they are later shown to be insecure (e.g., see [6, 2, 3]). While it is not difficult to see that constructing a provably secure non-invertible watermarking scheme is possible if a trusted third party is available (as shown in [3]), such a requirement of a trusted party makes it difficult to implement the scheme in many practical application scenarios. In a stand-alone setting, it is shown that if the false-alarm of the underlying watermarking scheme is high, the scheme cannot be non-invertible ([2, 3]).

When the false-alarm is very small, stand-alone non-invertible watermarking schemes are known to be possible, and a general construction of non-invertible watermarking scheme is given for spread-spectrum watermarking schemes in [7]. The main idea is to generate *valid* watermarks from a secret seed K and a cryptographically secure pseudo-random generator G, such that for a randomly chosen watermark W, the probability that W is valid (i.e., there exists a K such that $W = G(K)$) is negligible. In this way, it is proved in [7] that the existence of an ambiguity attacker that succeeds with a probability that is not negligible would imply the existence of a polynomial algorithm that can distinguish the output of the pseudo-random generator G and a truly random sequence with a probability that is not negligible, which contradicts with the assumption that G is cryptographically secure.

Although the security proof in [7] is sound, the security notion used in the proof is somewhat unnecessarily strict. In particular, an ambiguity attacker B is considered as *successful* if for any watermarked work $\widetilde{I}$, B is able to find a pair (W, K) such that W is present in $\widetilde{I}$ and $W = G(K)$ with a probability that is not negligible. It is worth to note that a successful attacker by this definition may be infeasible in practice. To see this, first let us consider an attacker who can find a pair (W, K) with high probability only in the following cases: (1) $n = |W|$ is a multiple of 1000, (2) $n > 10^9$, or (3) it works for all n but the probability is a small constant 2^{-80}. In all cases, the success probability of the attacker is not negligible with respect to n (asymptotically), but is not effective in practical sense. At the same time, attackers that are not successful by this definition can be a real threat in practice. For example, an attacker that finds a pair (W, K) with probability 1 when $n < 10^9$ and probability 0 otherwise is considered not successful by such a definition, but is really effective in many practical scenarios.

Therefore, to design a non-invertible watermarking scheme that is secure in practical sense, we need to be very careful about the security that we can achieve with practical parameters. In this paper, we propose to relax the definition of security by requiring that attackers are successful only when they can find a pair (W, K) with a *noticeable* probability. A noticeable quantity is one that can be bounded from below by $1/q(n)$, where $q(\cdot)$ is a fixed polynomial. This essentially guarantees that the attackers can always find a pair (W, K) with an expected effort of no more than $q(n)$. Furthermore, we can even require $q(n)$ to be asymptotically "small" (e.g., $q(n) = 2^{80}$, which is a constant).

We found that the security can be proved with much fewer constraints on the system parameters under the new security notion, which allows system designers to choose them more freely to meet other requirements of the system. We further analyze the security of the system in more details for some typical parameters.

We will discuss related work (Section 2) and give details of the watermarking model and the security notions (3). We analyze the exact security under the modified security notion for typical parameters, and discuss how to choose the parameters to achieve desired security in Section 4. We conclude in Section 5.

1-4244-1437-7/07/$20.00 ©2007 IEEE

2. RELATED WORK

Most of the work in digital watermarking literature focuses on the robustness, capacity, and perceptual distortion. Many security issues related to digital watermarking are not well-understood, and they are rarely treated in a way that is sufficiently rigorous.

Craver et al. [1] first study invertibility attacks that aim to find a fake watermark and a corresponding fake original from a watermarked work, so as to falsely claim the ownership of the work. Under such attacks, the attacker would be able to provide an evidence of ownership that is no weaker than the real owner. They give an attacker for spread-spectrum image watermarking schemes, and propose a counter measure, where watermarks are generated by applying a one-way hash function on the original work. The intuition is that an attacker would have to break the underlying one-way hash function to launch an invertibility attack. Qiao et al. [4, 5] study invertibility attacks on audio and video objects and give schemes that they claim to be non-invertible.

The weaknesses of these results are discussed in some subsequent papers [6, 2, 3]. Ramkumar et al. [6] give an algorithm to break the scheme proposed by Craver et al. [1], as well as an improved scheme. A formal definition of ambiguity attacks (which is a generalized version of invertibility attacks) is discussed in [2, 3], where it is pointed out that a scheme cannot be non-invertible if the false-alarm of the underlying watermarking scheme is high. A provably secure non-invertible scheme with the help of a trusted third party is proposed in [3], where valid watermarks are generated and issued by the trusted party. The first stand-alone provably secure non-invertible scheme is proposed in [7], where valid watermarks are generated by a cryptographically secure pseudo-random generator, and the underlying watermarking scheme is spread-spectrum based. A zero-knowledge proof for the detection algorithm in [7] is given in [8].

Observing that reducing the false-positive in the underlying watermarking scheme is important to the security of such non-invertible schemes, Sencar and Memon [9] proposed to embed not one but multiple watermarks into a work, such that the robustness and perceptual quality of the work is not affected, but the false-positive is reduced. In this way, by generating all watermarks using a secure one-way function and requiring all watermarks to be present for the ownership proof, the security of the resulting scheme can be improved.

3. NOTATIONS AND MODELS

3.1. Spread-Spectrum Watermarking

Here we consider a work $I = (x_1, \ldots, x_n)$ to be a sequence of n coefficients, which represents some feature space of a multimedia object (e.g., the DCT coefficients of images). The coefficients are independently and identically distributed standard Gaussian variables, with zero mean and unit variance. That is, x_i is drawn independently from $\mathcal{N}(0,1)$ for all $1 \leq i \leq n$. We follow the watermarking model in [7] and consider the following watermark embedder $\mathcal{E}$ and detector $\mathcal{D}$ with a watermark generator $\mathcal{G}$.

The watermark generator $\mathcal{G}(\cdot)$ takes a m-bit secret key K as input and outputs a watermark $W \in \{-1,1\}^n$. That is, $W = \mathcal{G}(K)$ is a sequence of -1's and 1's of length n. We say that a watermark W is *valid* if there exists a K such that $W = \mathcal{G}(K)$. Let the set of all valid watermarks be $\mathcal{W}$.

The embedder $\mathcal{E}(\cdot,\cdot)$ takes a work I and a watermark W as inputs and outputs a watermarked work $\widetilde{I}$. We consider a simple additive embedding algorithm. That is,

$$\widetilde{I} = \mathcal{E}(I, W) = I + \alpha W$$

for some $0 < \alpha < 1$, which is used to control the distortion to the work.

The detector $\mathcal{D}(\cdot,\cdot)$ takes a watermark W and a work I', which may or may not be watermarked by a valid watermark, and outputs 1 if W is detectable in I', and 0 otherwise. In particular, for some threshold T,

$$\mathcal{D}(I', W) = \begin{cases} 1, & \text{if } I' \cdot W > T \\ 0, & \text{otherwise.} \end{cases}$$

The parameters α and T should be chosen according to the requirements on robustness and distortion of the application. It is suggested in [7] that $\alpha = 0.01$ and in another setting proposed in [9], it is equivalent to $\alpha = 0.06$. For the convenience of analysis, we assume that $\alpha \leq 0.1$. Our analysis can be adapted easily for other values of α. Furthermore, we will use a threshold $T = \alpha n/2$, and similar analysis can be done for other choices. Although these values are plausible, it should be noted that they are somewhat arbitrary and should be modified to suit the needs of the actual application.

3.2. Attacker Models

Similar to the definition of ambiguity attacks in [7], we consider an attacker to be successful if, given a watermarked work $\widetilde{I}$, the attacker can invert the scheme with certain probability p. The difference is that, instead of only requiring p to be not negligible, we require p to be noticeable.

DEFINITION 3.1 (NEGLIGIBLE FUNCTION) *A function $f(n)$ is negligible with respect to n if for all positive polynomial $q(\cdot)$ and for sufficiently large n, it holds that $f(n) < 1/q(n)$.*

DEFINITION 3.2 (NOTICABLE FUNCTION) *A function $f(n)$ is noticeable with respect to n if there exists a positive polynomial $q(\cdot)$ such that for sufficiently large n, it holds that $f(n) > 1/q(n)$.*

An example of negligible function is $f(n) = 2^{-n}$, and functions such as $f(n) = 1/n$ and $f(n) = 2^{-80}$ are noticeable. It is worth to note that there are functions that are neither negligible nor noticeable, hence being *not negligible* does not imply being noticeable. Now we define a successful ambiguity attacker.

DEFINITION 3.3 (SUCCESSFUL ATTACKER) *A successful ambiguity attacker B is a probabilistic polynomial-time algorithm such that, given a watermarked work $\widetilde{I} = \mathcal{E}(I, W)$ for some work I and valid watermark $W \in \mathcal{W}$, B finds a pair (W', K') with a noticeable probability $p(n)$ so that $\mathcal{D}(\widetilde{I}, W') = 1$ and $W' = \mathcal{G}(K')$.*

In many practical scenarios, we need to be more specific about the success probability for all typical values of n. Hence, we also define a stronger ambiguity attacker as below.

DEFINITION 3.4 (STRONG ATTACKER) *An ℓ-strong attacker S is an attacker with a success probability $p(n) > 2^{-\ell}$ for all n.*

Note that ℓ here is an arbitrary constant. The larger the ℓ, the more secure the scheme is if it is proved to withstand such an ℓ-strong attacker.

3.3. Pseudo-Random Generator

A cryptographic pseudo-random generator (PRG) G is an expanding function such that given an m-bit seed K it outputs an n-bit string Y for $n > m$ and $n = poly(m)$ for some positive polynomial $poly$. As noted in [7], it is straightforward to construct a watermark generator $\mathcal{G}$ from a PRG G: We just need to map every output of 0 bit into a -1, and leave 1 bits unchanged.

By convention, a PRG G is secure if no efficient algorithm can distinguish its output from that of a truly random source.

DEFINITION 3.5 *A PRG G is secure if for all probabilistic polynomial-time algorithm A, for any polynomial q and sufficiently large m, it holds that* $|\Pr[A(G(\mathcal{U}_m)) = 1] - \Pr[A(\mathcal{U}_n) = 1]| < 1/q(m)$.

Here $\mathcal{U}_k$ denotes a uniformly distributed binary string of length k.

Furthermore, there are a number of PRG's that are secure under this definition. The Blum-Blum-Shub PRG [10] was suggested in [7]. Nevertheless, any secure PRG would suffice.

4. SECURITY ANALYSIS

4.1. A Previous Proof ([7])

The non-invertibility of the watermarking scheme described in Section 3 is proved in [7] with respect to a weaker attacker definition where the success probability is only required to be not negligible. For completeness, we briefly describe the proof given in [7].

The main idea is to show that if a successful attacker B exists, we can use B as an oracle to construct another algorithm $\mathcal{T}$ that distinguishes $\mathcal{G}$ and a truly random source with a probability that is not negligible, which contradicts with the assumption that $\mathcal{G}$ is constructed from a secure PRG G, hence such attacker does not exists.

The algorithm $\mathcal{T}$, given input string W, does the following.

1. Randomly choose a work I.
2. Embed W into I and obtain $\widetilde{I}$.
3. Send $\widetilde{I}$ to B and obtain its output.
4. If B finds a pair (W', K') such that $\mathcal{D}(\widetilde{I}, W') = 1$ and $W' = \mathcal{G}(K')$, output 1. Otherwise output 0.

Clearly $\mathcal{T}$ runs in polynomial time. The claim that $\mathcal{T}$ is an effective distinguisher is established by further considering the following two cases.

In the first case, W is a valid watermark. That is, $W = \mathcal{G}(K)$ for some K. In this case, B correctly outputs a pair (W', K') with a probability $p(n)$ that is not negligible by our hypothesis. The expected output of $\mathcal{T}(W)$ will be $p_0(n)$, which is also not negligible.

In the second case, W is uniformly random. In this case, we consider the probability $V(n)$ that some valid watermark happens to be detectable in $\widetilde{I}$. According to some asymptotic analysis in [7], it is shown that $V(n)$ is negligible for $m = \sqrt{n}$. In this case the expected output of $\mathcal{T}(W)$ will be no more than $V(n)$, which is negligible.

Therefore, the difference in $\mathcal{T}(W)$ between these two cases cannot be negligible, which means that $\mathcal{T}(W)$ distinguishes the PRG G and a truly random source with a probability that is not negligible.

4.2. Practical Security Analysis

Now let us consider stronger definitions of attackers as in definitions 3.3 and 3.4. We note that the previous proof as briefly described in Section 4.1 can be adapted to analyze the security with regard to our definitions of attackers. The construction of $\mathcal{T}$ is the same as before, and we also consider the same two cases. The difference is the argument that follows.

In particular, in the first case where W is a valid watermark, now the attacker can successfully find a pair (W', K') with probability $p_1(n)$. Hence, the expected output of $\mathcal{T}(W)$ is $p_1(n)$. In the second case where W is uniformly random, we similarly consider the probability $V(n)$ that a valid watermark happens to be detectable in $\widetilde{I}$. Let $p_2(n)$ be the probability that an attacker can find a pair (W', K') given that a randomly watermarked work $\widetilde{I}$ contains a valid watermark. The expected output of $\mathcal{T}(W)$ in this case will be $V(n)p_2(n)$.

The key quantity here is the difference $D(n) = p_1(n) - V(n)p_2(n)$, which must be negligible, otherwise $\mathcal{T}$ can distinguish the output from G and that of a truly random source with a probability that is not negligible. The proof in [7] essentially states that if $V(n)$ is negligible, then so is $V(n)p_2(n)$, and hence so is $p_1(n)$.

4.2.1. Practical Security

We make an important observation that when $V(n)$ is not negligible (but still less than 1), the difference between $p_1(n)$ and $p_2(n)$ should be negligible. Therefore, what remains to be shown is that when $p_1(n)$ is noticeable, $D(n)$ is also not negligible, hence arrive at the same contradiction. We can safely assume that $p_2(n) < 2p_1(n)$, since otherwise the difference would be $p_1(n)$, which is not negligible. In this case, $D(n) > p_1(n)(1 - 2V(n))$. For this quantitity to be negligible, $1 - 2V(n)$ has to be negligible. Hence, to create a contradiction, it suffices to move $V(n)$ away from $1/2$, which is our focus here.

CLAIM 4.1 *For large enough n, $0 < \alpha \leq 0.1$ and threshold $T = \alpha n/2$, we have*

$$V(n) < \frac{\sqrt{2(1+\alpha^2)}}{\alpha\sqrt{\pi n}} e^{-\frac{\alpha^2 n}{8(1+\alpha^2)}} 2^m < 2^{-(0.18\alpha^2 n - m)} \quad (1)$$

To see this, let us first consider the probability $v_i(n)$ that a particular valid watermark $W_i \in \mathcal{W}$ can be detected in a randomly watermarked work $\widetilde{I}$. Recall that a randomly watermarked work $\widetilde{I} = I + \alpha W_r$ with randomly selected I and W_r (where W_r may or may not be valid). Also, recall that each coefficient x_i in $\widetilde{I}$ follows a standard normal distribution, and each w_i in W_r is uniformly chosen from $\{-1, 1\}$. Let $Y = \widetilde{I} \cdot W_i = I \cdot W_i + \alpha(W_r \cdot W_i)$ be the inner product that is to be compared with the threshold T. The random variable $I \cdot W_i$ is a normal distribution $\mathcal{N}(0, n)$, and when n is large enough, $\alpha(W_r \cdot W_r)$ can be approximated by a normal distribution $\mathcal{N}(0, \alpha^2 n)$. Hence Y approximately follows a normal distribution $\mathcal{N}(0, (1+\alpha^2)n)$, where the standard deviation $\sigma = \sqrt{(1+\alpha^2)n}$. Let $z = T/\sigma$, we have

$$v_i(n) = \Pr[Y > T] < \frac{1}{\sqrt{2\pi}z} e^{-z^2/2}. \quad (2)$$

Considering that $0 < \alpha \leq 0.1$ and $z = T/\sigma = \frac{\alpha\sqrt{n}}{2\sqrt{1+\alpha^2}}$, which is larger than 1 for large enough n, we have

$$v_i(n) < \frac{1}{\sqrt{2\pi}} e^{-\frac{\alpha^2 n}{8(1+\alpha^2)}} < 2^{-(0.18\alpha^2 n)} \quad (3)$$

Finally, note that $V(n) \leq \sum_{i=1}^{|\mathcal{W}|} v_i(n) \leq 2^m v_i(n)$, hence Inequality (1) holds as claimed.

Now, since $D(n)$ represents the advantage any efficient attacker could have to attack the underlying PRG, we can link the values of $p_1(n)$, m and $V(n)$ with the security of the PRG. For many existing PRG's, for any given value of m, it is possible to make $D(n) < 2^{-m+1}$. Hence, to make sure that we have ℓ bit security, we only need to have $m \geq \ell + 1$. Therefore, to make sure that $V(n)$ is less than $1/2$, it suffices to choose

$$0.18\alpha^2 n \geq m + 1. \tag{4}$$

4.2.2. *On Choosing the Parameters*

As noted in [7], the choice of the security parameter m is crucial for the actual security. Using the analysis in Section 4.2.1, we know that m is related to the actual security through the parameters of the underlying PRG (e.g., $m = \ell + 1$), and also related to n according to (4).

One one hand, we can choose a desired security level ℓ first, and then choose the parameters for the underlying PRG, which determines the minimum value of m, which in turn determines the minimum value for n with a chosen α. One the other hand, we can also examine the underlying watermarking scheme first, determine the maximum n and the value of α we need, and then determine the maximum value m using (4), which determines the exact level of security ℓ together with the underlying PRG.

For example, using Blum-Blum-Shub PRG with appropriate parameters and if $\alpha = 0.1$, we need to have $1.8 \times 10^{-3} n \geq \ell + 2$ to achieve ℓ bits of security. Hence, if we need 80 bits of security, we can choose $m = 81$, and for $\alpha = 0.1$, $n = 4.56 \times 10^4$ would be sufficient. Smaller values of α would require larger n. For example, for $\alpha = 0.05$, we would require $n = 1.83 \times 10^5$. If we only require a weaker security of 60 bits, we would require $m = 61$ and $n \geq 3.45 \times 10^4$ and $n \geq 1.38 \times 10^5$ for $\alpha = 0.1$ and $\alpha = 0.05$ respectively. We can also use the same analysis to find out what is the required values of α if both m and n are fixed. For example, when $m = 81$ and $n \geq 1.83 \times 10^5$, we can choose an α from anywhere between 0.05 and 1, as long as the robustness and distortion requirements are satisfied.

4.2.3. *Other Watermarking Variants*

One straightforward way to improve the security of the scheme is to employ the multiple-watermark variant of the spread-spectrum watermarking scheme as given in [9]. Such techniques would allow us to further reduce the false-positive rate without affecting the robustness and distortion of the scheme. It is also possible to employ other watermarking techniques that gives much lower false-positives. In these cases, the quantity $V(n)$ can be further reduced with the same parameters m, n and α, hence making it easier to choose the parameters in practice.

5. CONCLUSIONS

In this paper we study a recently proposed provably secure non-invertible watermarking scheme built upon spread-spectrum watermarking [7]. We note that although their security proof is sound, the results are mainly based on theoretical asymptotic arguments, which need to be examined more carefully in actual applications to achieve desired security.

We observe that the security notion in [7] may be unnecessarily strict in practical scenarios, and propose to look at some weaker security notions that are still reasonable in practice. In particular, we consider an attacker to be successful in launching ambiguity attacks only when the success rate is the reciprocal of a fixed polynomial, or a constant (which is a special case of a polynomial). In other words, we require that all successful attackers must be able to break the system with *reasonable* expected amount of efforts by today's computation standard.

We further use the new security notion to re-examine the security proof given in [7], and investigate the exact requirements on the parameters of the watermarking scheme such that the security is maintained at a reasonable level.

We note that the suggested parameters from our analysis may not be practical in all scenarios. However, we believe such analysis is important for the following reasons. First, it may serve as negative results in some application scenarios. Secondly, if the underlying watermarking scheme can be improved to achieve a much better performance, especially false-positives, we can employ similar analysis to work out the required parameters. Lastly, when a system designer needs to consider the feasibility of achieving certain performance and security requirements, such analysis serves as an example of how to trade-off among several parameters.

6. REFERENCES

[1] S. Craver, N. Memon, B.L. Yeo, and M.M. Yeung, "Resolving rightful ownerships with invisible watermarking techniques: Limitations, attacks, and implications," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 4, pp. 573–586, 1998.

[2] A. Adelsbach, S. Katzenbeisser, and A.-R. Sadeghi, "On the insecurity of non-invertible watermarking schemes for dispute resolving," *International Workshop on Digital Watermarking (IWDW)*, pp. 374–388, 2003.

[3] A. Adelsbach, S. Katzenbeisser, and H. Veith, "Watermarking schemes provably secure against copy and ambiguity attacks," *DRM*, pp. 111–119, 2003.

[4] L. Qiao and K. Nahrstedt, "Watermarking methods for mpeg encoded video: Towards resolving rightful ownership," in *Proc. of ICMCS*, 1998, pp. 276–285.

[5] L. Qiao and K. Nahrstedt, "Non-invertible watermarking methods for MPEG encoded audio," in *SPIE 3675, Security and Watermarking of Multimedia Contents*, 1999, pp. 194–202.

[6] M. Ramkumar and A. Akansu, "Image watermarks and counterfeit attacks: Some problems and solutions," in *Symposium on Content Security and Data Hiding in Digital Media*, 1999, pp. 102–112.

[7] Qiming Li and Ee-Chien Chang, "On the possibility of non-invertible watermarking schemes," in *Information Hiding Workshop*, 2004, vol. 3200 of *LNCS*, pp. 13–24.

[8] Qiming Li and Ee-Chien Chang, "Zero-knowledge watermark detection resistant to ambiguity attacks," in *ACM Multimedia Security Workshop*, 2006.

[9] H.T. Sencar and Nasir Memon, "Combatting ambiguity attacks via selective detection of embedded watermarks," *IEEE Transactions on Information Forensics and Security*, Accepted in 2007.

[10] L. Blum, M. Blum, and M. Shub, "A simple secure unpredictable pseudo-random number generator," *SIAM Journal on Computing*, vol. 15, pp. 364–383, 1986.