Cryptanalysis of Hwang-Chang's a Time-Stamp Protocol for Digital Watermarking

*Jue-Sam Chou [1], Yalin Chen [2], Chung-Ju Chan [3]

[1] Department of Information Management, Nanhua University Chiayi 622 Taiwan, R.O.C

*: corresponding author

jschou@mail.nhu.edu.tw

Tel: 886+ (0)5+272-1001 ext.56226

[2] Institute of information systems and applications, National Tsing Hua University

d949702@oz.nthu.edu.tw

Tel: 886+(0)3-5738997

[3] Department of Information Management, Nanhua University Chiayi 622 Taiwan, R.O.C

chanchungju@gmail.com

Tel: 886+ (0)5-2721001 ext.2017

Abstract

In 2005, Hwang et al. [17] proposed a time-stamping protocol for digit watermarking. They claimed that their scheme is secure against attacks. However, in this article, we will show that their scheme is not secure enough for that when the owner of the image sends both the encrypted session key and image to the TSS, the attacker can intercept these transmitted data. Then, he can launch an off-line attack to analyze these intercepted data. We will describe the attacker's action in this article. After that, we propose an improved scheme to prevent this off-line attack.

***Keywords*:** Digital watermarks; Copyright protection; Time-stamping

1. Introduction

Since the concept of watermarking was put forward in the early 1990s, many researchers have paid much attention to it, such as [1-18]. Each scheme intends to become an effective approach in this area. Generally speaking, a well constructed watermark can be embedded into the original data under the requirement that the original data quality degradation is imperceptible. Also, it should be able to be detected when the watermarked data had been tampered. Besides, a good watermarking scheme should possess the following properties. (1) Robustness: since watermarking is designed as a

way to authenticate the copyright of the input data, it must be able to resist against different kinds of intentional attacks. (2)Imperceptibility: watermarked data should be accepted with imperceptible quality degradation. (3)Security: in watermark processing, only the authorized user can execute the corresponding security measurements.

There are two main categories in the study of watermarking: the first is the wave watermarking and the second is the spatial domain watermarking. Although in 1998, wave watermarking technique concentrating on spatial domain [9] claimed that their scheme is robust against the general signal processing attack. However, in 1999 and 2000, Pereira et al. found that their scheme can be broken by two different attacking algorithms as listed in [1] and [10] by using geometric attack. Besides, in [1] and [10], they also proposed two improvements, respectively. In their two improvements, each incorporates a signal analysis scheme to strengthen the robustness against the geometric attack. Nevertheless, in 2001, Herrigel et al. [11] found the same weakness as stated in [9] in their two improvements, [1] and [10]. As for the second kind of watermarking technique, the spatial domain watermarking, there has been several papers proposed in this area, such as [12, 13, 14, 15, 16, 18].

However, in 2005, Hwang et al. [17] pointed out that all the above-mentioned watermarking schemes are not secure enough, because there still exists some breaking methods to the original or embedded image, such as JPEG compression, filtering, noise addition, scaling, rotation and geometric distortions, and so on. Thus, they proposed a time-stamping protocol intending to prevent malicious attacks. They claimed that their scheme is secure. However, after our analysis, we find that their scheme suffers from the off-line attack.

The organization of this article is as follows: in Section 2, we briefly review Hwang et al.s' scheme. In Section 3, we describe the reason why their scheme is not secure enough. In Section 4, we show the improvement of their scheme. Finally, a conclusion is given in Section 5.

2. Review of the Hwang-Chang's scheme

In this section, we first rewrite the notations used in Hwang et al.s' protocol [17] in Section 2.1 to make their scheme more readable, then in Section 2.2, we briefly introduce

their protocol.

2.1 Rewriting the notations and definitions used in Hwang et al.s' protocol

The following notations are the replacements of the ones used in Hwang et al.s' scheme.

- X, Y: Denote two strings.
- Sign_K (X): Denotes the signature on X using key K.
- Ver_K (Y): Denotes the verification of Y using key K.
- [X, Y]: Denotes a string consisted of string X concatenated with string Y.
- H(X): Denotes the hash value of string X.
- E_K(X): Denotes the encryption of string X using key K.
- D_K(Y): Denotes the decryption of string Y using key K.
- t: Denotes the timestamp.

2.2 Hwang-Chang's protocol

Hwang- Chang's time stamping watermarking scheme mainly consists of three phases: (1) the setup phase (2) time-stamp signing phase and (3) time-stamp verifying phase. In the time stamp verifying phase, everyone can use the TSS public key to verify the embedded watermark at any time. In this section, we only delineate (1) the setup phase and (2) the time-stamp signing phase as follows:

(1) Setup phase: The system is initialized with each participant having a public key P_u and a private key S_u. All participants must be authenticated by a trusted certificated authority in advance and there is a secure one-way hash function known to all participants. The owner of the original host media X chooses a random session key *r* in the protocol.

(2) Signing phase: The details for their watermarking time-stamping protocol in the signing phase are shown in figure 1 and described as follows:

Step1. Owner *i* randomly chooses a number *r* and encrypts it using

TSS's public key to obtain the value *c*. Meanwhile, he also encrypts *X* using r as the session key to obtain *CX*. Then he sends both *c* and *CX* to TSS.

Step2. After receiving *c* and *CX*, TSS decrypts *c* using his private key S_{TSS} to obtain *r*. Then, using *r* to decrypt *CX*, he can deduce the value *X*. Then, TSS uses secure one-way hash function *H* to compute the hash value of the received *CX*, obtaining *d*. Then, he concatenates a time stamp *t* to *d* and signs on this concatenation using his private key S_{TSS}, obtaining s_1. Finally, TSS uses *r* to encrypts the concatenation string [*t*, s_1], obtaining *T*. That is, TTS computes $\boldsymbol{s_1 = \mathbf{Signs}_{TSS}([t, H(CX)])}$ and $\boldsymbol{T = E_r([t, s_1])}$, where *t* is the timestamp. After that, TSS sends *T* to owner *i*.

Step3. After receiving *T*, owner *i* uses the session key *r* to decrypt it and obtains *t* and the TSS's signature s_1. Then, before embedding the watermark, owner *i* checks to see if s_1 is valid by the following equation.

$$Ver_{P_{TSS}}(s_1) = [t, H(CX)] \qquad (1).$$

If it is valid, owner *i* then embeds the watermark *m* to the original media *X*, getting [*X*, *m*]. After that, he first encrypts s_1 using session key *r*, obtaining *ES1*, then using *ES1* as the encryption key to encrypt [*X*, *m*], obtaining X_W. That is, he computes $X_W = E_{ES1}([X, m])$. Finally, he sends X_W to TSS.

Step4. After receiving X_W, TSS also computes *ES1* and uses *ES1* as the decryption key to decrypt, obtaining [*X*, *m*]. He then extracts *X* and compares it with the earlier computed *X* obtained in step2 to see if these two values are equal. If it

so, TSS signs on [s_1, H (X_W)] to get s_2(=$Signs_{TSS}$([s_1, $H(X_W)$])). He then encrypts s_2 using session key r and sends the result $E_r(s_2)$ to owner i. Finally, TSS destroys X, r, s_1, s_2 and X_W.

Step5. After receiving $E_r(s_2)$, owner i first decrypts it and then checks to see if s_2 is signed by TSS by the following equation.

$$Ver_{P_{TSS}}(s_2) = [s_1, H(X_W)] \qquad (2).$$

When all of the above steps were done, the time stamp watermark signing phase is completed. The time-stamped signatures s_1, s_2, and the session key r all must be kept secret by the owner. After that, when a notary requests owner i to verify the time stamp t of a watermarked medium X_W, owner i must send (t, s_1, s_2, CX) to the notary, the notary then can verify the time-stamped signatures s_1 and s_2 using Eqs. (1) and (2).

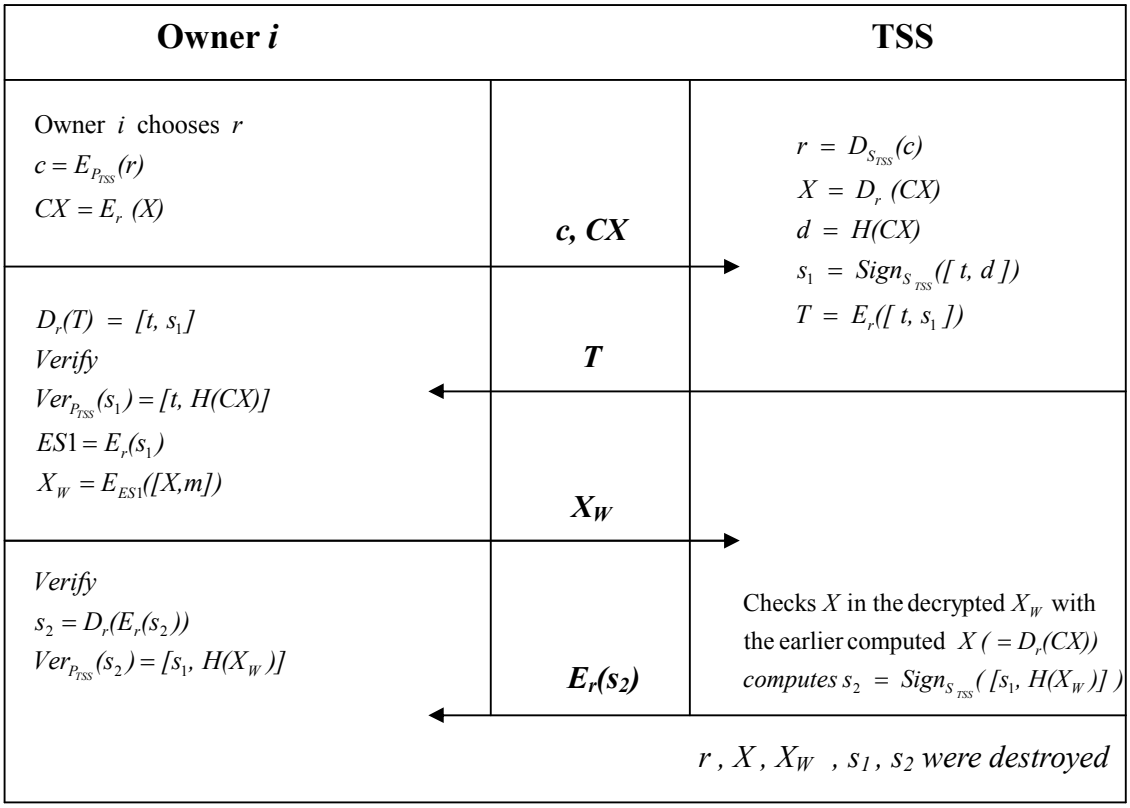

Fig.1. Hwang-Chang's time stamp protocol for watermarking.

3. Why their scheme is not secure enough

In this section, we point out why their scheme is not secure enough. For, in their scheme, the owner sends all the messages, CX, T, $E_r(s_2)$ encrypted using the same key r, except for the values c which is encrypted using TSS's public key. This would be a weakness since the attacker can launch an off-line attack due to the ability of today's computer computing cooperation through network, i.e., the collision finding of hash function MD5 is under such a computation cooperation [19]. Especially, the ciphertext T is the encryption of the concatenation of timestamp t and s_1 using key r. For t is usually an easily recognizable parameter, the attacker has the right r, once he has seen the recognizable timestamp t in the decryption result. And after this, the attacker has obtained the session key r, he can get s_2 and therefore s_1. From s_1 and r, he can compute $ES1$ and henceforth obtains $[X, m]$ from X_W. Also, he can get X from CX.

Besides, except for the above-mentioned weakness, in their scheme, TTS destroys the parameters, r, X, X_W, s_1 and s_2 finally. This would incur the situation where owner i claims that the idea X belongs to him but the attacker can prove to the third party that he owns the idea X as well for he has s_1 and s_2 after he has the session key r. The notary can not identify which one is the real owner of the copyright X. This second weakness is from the fact that ID is not bound to the idea X of the real owner. Hence, we propose an improvement to enhance its security.

4. Improvement of their scheme

In this section, we present an improvement on both the security and efficiency in their scheme. We describe our method in Section 4.1, and in Section 4.2, the signing phase and the verification phase, respectively. In our improvement, we denote ID_i as owner i's ID, T_p as the TSS's public key and T_s as the TSS's secret key. Our scheme just runs in two passes which is far more efficient and secure than Hwang-Chang's scheme.

4.1 The signing phase

The procedure for our watermarking time-stamping protocol in signing phase are shown in figure 2 and also described as follows:

Step1. Owner *i* randomly chooses a number *r* as the session key. He encrypts the original image *X*, the watermark *m*, his ID_i and *r* using TSS's public key T_p to obtain the value $c(= E_{T_p}(X, m, ID_i, r))$ and then sends to TSS.

Step2. TSS decrypts *c* using his private key T_s to obtain *X*, *m*, ID_i and *r*. Then, TSS uses the one-way hash function *H* to compute the hash value of *X*, *m*, ID_i and a timestamp *t*, obtaining *d*. After that, he signs on *d* using his private key, obtaining s_1. Finally, TSS uses session key *r* to encrypt s_1, obtaining T_1 and at last he sends the timestamp t and T_1 to owner *i*.

Step3. After receiving t and T_1, owner *i* uses *r* to decrypt T_1 and obtains the TSS's signature s_1. Then, owner *i* verify to see if s_1 is valid and checks to see whether the result of equals to H(*X*, *m*, ID_i, *t*) by the following equation.

$$\mathrm{Ver}_{\mathrm{Tp}}(s_1) = \mathrm{d} = H(X, m, ID_i, t). \quad (3).$$

Owner *i*		**TSS**
Owner i chooise r $c = E_{T_p}(X, m, ID_i, r)$	***c*** →	$(X, m, ID_i, r) = D_{T_s}(c)$ $d = H(X, m, ID_i, t)$ $s_1 = Signs_{T_s}(d)$ $T_1 = E_r(s_1)$
$(s_1) = D_r(T_1)$ $ver_{T_p}(s_1) = d$ *checks d ? = H(X, m, ID_i, t)*	← ***t, T_1***	
		r , X , m , s_1 were destroyed

Fig.2. the signing phase of our proposed time-stamping watermarking.

4.2 The verification phase

The time-stamping verification phase is described as follows:

Step1. The notary requests owner *i* to verify the time-stamp of a medium *X* and the corresponding watermark *m*.

Step2. The owner i sends (X, m, ID_i, t, s_1) to the notary.

Step3. The notary verifies s_1, obtaining d and compares d to the computed H(X, m, ID_i, t) in Eq (3).

5. Security analysis

The improvement we proposed is that we embed the owner's ID, a timestamp t, and the watermark m to the copyright X to be hashed by TSS. Our method can get rid of the situation where the notary can not identify which one, the real owner i or the succeeded off-line attacker is the real owner of the copyright of X as mentioned in Section 3.

For, using our attack, if an attacker intercepts the value c, he can not decrypt it because he does not know the TSS's secret key. Besides, our method also avoid the situation where the easily recognizable timestamp t is directly encrypted by r. If T1 is broken, the attacker still can not obtain X and m for they are hashed by a secure one way function. This makes our improvement more robust than Hwang-Chang's scheme.

6. Conclusion

This article shows that Hwang et al.s' scheme is not secure enough. For, in their scheme, most transmitted data are encrypted using the same key, thus making an attacker can launch an off-line attack to analyze the encrypted data. We have proposed an improved scheme to enhance their security and after our analysis in Section 5, we conclude that our scheme is more robust than theirs.

References

[1] S. Pereira, J.J.K.O. Ruanaidh, F. Deguillaume, G. Csurka, T. Pun, "Template based recovery of fourier-based watermarks using log–polar and log–log maps, "Proc. IEEE Int. Conf. Computing and Systems, vol. 1, 1999, pp. 870–874.

[2] C.-W. Tang, H.-M. Hang, "A feature-based robust digital image watermarking scheme, "IEEE Trans. Signal Process. 51 (4) (2003) 950–959.

[3] M. Kutter, S.K. Bhattacharjee, T. Ebrahimi, "Towards second generation watermarking schemes, " Proc. IEEE Int. Conf. Image Process., vol. 1, 1999, pp. 320–323.

[4] S. Bhattacharjee, M. Kutter, “Compression tolerant image authentication, ”Proc. IEEE Int. Conf. Image Process., vol. 1, 1998, pp.435–439.

[5] B.-S. Kim, J.-G. Choi, K.-H. Park, “Image normalization using invariant centroid for rst invariant digital image watermarking, ”IWDW 2002 2613 (2003) 202–211.

[6] J.J.K. O_Ruanaidh, W.J. Dowling, F.M. Boland, “Phase watermarking of digital image, ”Proc. IEEE Int. Conf. Image Process., vol. 3, 1996, pp. 239–242.

[7] M. Kutter, F. Jordan, F. Bossen, “Digital signature of color images using amplitude modulation, ”J. Electron. Imaging 7 (2) (1998) 326–332.

[8] I. Cox, J. Kilian, F.T. Leighton, T. Shamoon, “Secure spread spectrum watermarking for multimedia, ”IEEE Trans. Image Process. 6 (12) (1997) 1673–1687.

[9] H. Inoue, A. Miyazaki, A. Yamamoto, T. Katsura, “Digital watermark based on the wavelet transform and its robustness on image compression, ”Proc. IEEE Int. Conf. Image Process., vol. 2, 1998, pp. 391–395.

[10] S. Pereira, T. Pun, “Robust template matching for affine resistant image watermarks, ”IEEE Trans. Image Process. 9 (6) (2000) 1123–1129.

[11] A. Herrigel, S. Voloshynovskiy, Y. Rytsar, “The watermark template attack, ”Proc. SPIE, vol. 4314, 2001, pp. 394–400.

[12] J. O’ Ruanaidh, T. Pun, Rotation, “scale and translation invariant spread spectrum digital image watermarking, ”Signal Process 66 (3) (1998) 303–317.

[13] D. Zheng, J. Zhao, A.E. Saddik, “Rst-invariant digital image watermarking based on log–polar mapping and phase correlation, ”IEEE Trans. Circuits Syst. Video Technol. 13 (8) (2003) 753–765.

[14] V. Solachidis, I. Pitas, “Circularly symmetric watermark embedding in 2-D DFT domain, ”IEEE Trans. Image Process. 10 (11) (2001) 1741–1753.

[15] M. Kutter, “Watermarking resisting to translation, rotation and scaling, ”Proc. SPIE, vol. 3528, pp. 423–431.

[16] F. Sebe’, J. Domingo-Ferrer, J. Herrera, “Spatial-domain image watermarking robust against compression, filtering, cropping, and scaling, ”ISW 2000, Berlin, LNCS, vol. 1975, Springer-Verlag, Berlin, 2000, pp. 44–53.

[17] M.S. Hwang, K.F. Hwang, C.C. Chang, “A time-stamping protocol for digital watermarking, ”Applied Mathematics and Computation 169, 2005, 1276–1284.

[18] W. Lu, H. Lu, F.L. Chung, “Feature based watermarking using watermark template match, ”Applied Mathematics and Computation 2005.

[19] Eric Thompson, “MD5 collisions and the impact on computer forensics, “Digital investigation 2005, pp. 36-40.