World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:1, No:7, 2007

Watermark-based Counter for Restricting Digital Audio Consumption

Mikko Löytynoja, Nedeljko Cvejic, and Tapio Seppänen

***Abstract*—**In this paper we introduce three watermarking methods that can be used to count the number of times that a user has played some content. The proposed methods are tested with audio content in our experimental system using the most common signal processing attacks. The test results show that the watermarking methods used enable the watermark to be extracted under the most common attacks with a low bit error rate.

***Keywords*—**Digital rights management, restricted usage, content protection, spread spectrum, audio watermarking.

I. Introduction

THE recent popularity of broadband Internet had made downloading and distribution of multimedia content more easier for a large number of users. As music, video and other multimedia content is distributed in digital form, it not only increases the quality of the content but also eases copying it, and the risk of piracy grows. Since digital content can be copied without degrading its quality and content providers increasingly lose revenues to content piracy, digital watermarking is seen as one of the possible solutions. Encryption alone is not a solution to solve the problem of the content piracy, because it only protects the content delivery. After the content has been decrypted, it is no longer protected. A straightforward approach to prevent piracy is to prevent the user's access to the decrypted content. However, the user has to be able to play the content, which requires the content to be decrypted. Watermarking can be used to provide additional protection as it is embedded directly into the content and cannot be removed without decreasing the quality of the content.

Digital watermarking enables us to embed additional information to digital content (e.g. images, audio, video), so that the user is unable to perceive the embedded information from the cover media. Another describable property is that the embedded watermark is very hard to remove from the cover media. Watermarking has been used in applications like broadcast monitoring, owner identification, proof of ownership, authentication, transaction tracking, copy control, digital rights management and covert communication [1].

One copy protection application where watermarking is being considered to be used is DVD videos, where it would enhance the protection provided with encryption. Even if DVD content were decrypted and copied, the compliant players would refuse to play it, if the content contains a copy protection watermark and the content is on illegally copied media [2]. The compliant players can be required, by patent license, to check for the watermark.

Digital rights management (DRM) can be used to describe what the user can do with the content and what are the requirements and limits for that kind of content usage. One of the restrictions enabled by DRM is to limit the number of times that the content can be played. This method could be used for example in an Internet-based content rental store use case, in which the user downloads the content and is allowed to play it for a certain number of times. The counter scheme could also be used to advertise content, where the user is allowed to play the content for a few times before a payment is required.

Tracking the number of instances the content has been played is a very complex task, if the counter is to be stored on the user's terminal. The reason is the fact that user must be considered a hostile party in this kind of a use case. If the content is played with a terminal that has online connection the counter can reside on a server, which makes the attacking harder; the attacker must concentrate on modifying the player software or remove the protection from the content in order to crack the protection. In order to make the protection more secure, a hardware-based solution must be used, since the user is able to modify the data stored on the local terminal. Frequent patching can help to close holes in the system, but much better results could be achieved with a trusted computing platform [3].

In this paper we propose a novel watermark-based counter scheme that can be used to count the number of remaining usages of audio content. In Section 2 we present three alternatives for a watermark-based counter. Section 3 presents the watermarking algorithms used. In Section 4 we discuss the results of attacking experiments against the used watermarks. Section 5 introduces attacks that are not targeted to watermark removal. Section 6 concludes the paper.

Manuscript received July 15, 2005. This work was supported by the National Technology Agency of Finland (TEKES), Graduate School in Electronics, Telecommunications and Automation (GETA) and Infotech Oulu Graduate School of the University of Oulu. Financial support provided by the partners in the Stego and Zirion projects consortiums is gratefully acknowledged.

Authors are with the MediaTeam Oulu Research Group at the Department of Electrical and Information Engineering, University of Oulu, Finland (http://www.mediateam.oulu.fi), address: MediaTeam Oulu, P.O. Box 4500, 4STOINF, 90014 University of Oulu, Finland (email: {lode,cvejic,tapio}@ee.oulu.fi).

Open Science Index, Computer and Information Engineering Vol:1, No:7, 2007 publications.waset.org/2896.pdf

II. Watermark-based Counter Methods

Using digital watermarking to restrict content consumption can be done in several ways. In this section we present a few alternative ways to implement the counter and discuss their advantages and disadvantages. The watermarking algorithm used in the counter application does not need to have a very large embedding capacity, but it should be as robust as possible while still being imperceptible.

A. *Method 1: Interval Watermarking*

The first method embeds a watermark into audio content in specific intervals. The interval length is called watermarking interval. The lengths of the content segments within the counter intervals that have not been watermarked are related to the number of usages left. After each usage, a new segment is watermarked within each watermarking interval, thus decrementing the counter value.

This counter scheme can utilize either fixed length watermarks or varying length watermarks. In the first alternative, after each usage of the content, it is watermarked by a fixed length segment within the watermarking interval. This implies that, as a large value for the granted usage number is used, the watermarking interval must be increased as well. However, an advantage is that knowledge of the number of granted usages is not needed for calculating the current counter value at any time instant.

The second alternative uses fixed length watermarking intervals to embed the counter data. Therefore, the more usages the user has been granted, the shorter the watermark segment is. This scheme has the drawback that knowledge of the number of granted usages is needed in order to prevent ambiguous counter values, since the length of the watermark segment varies.

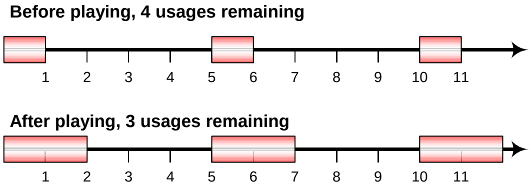
Before playing, 4 usages remaining
1
2
3
4
5
6
7
8
9
10
11
After playing, 3 usages remaining
1
2
3
4
5
6
7
8
9
10
11

Fig. 1 Counter based on watermarking intervals

B. *Method 2: Single Watermark*

In the second method the information about the number of times the content can be played is embedded into one watermark to embed the information in the content. In the simplest version, a watermark can be embedded in a fixed location in the content. To increase security, repetition can be applied to prevent the user from clipping off the part of the content where the watermark is located. In this method the embedded watermark must be updated during content consumption to update the counter. The player must be able either to remove the old watermark and replace it with a new one or to update the embedded watermark.

The advantages as compared to Method 1 are that a smaller amount of time needs to be used for audio watermarking, and, due to the small amount of corrupted signal, the user can easily renew the license without downloading the audio file.

A disadvantage is that, if the player can remove the watermark while updating it, the malicious attacker is more likely able to remove the watermark also. Another disadvantage is that if the embedded watermark cannot be fully removed, the remaining parts of the watermarks add noise to the watermarked content, thus limiting the number of times that the watermark can be updated.

C. *Method 3: Multiple Watermarks in Single Location*

The main principle of the third method is to embed multiple watermarks in the same location. The number of watermarks in that location indicates how many times the content has been played. The maximum number of times that the user is allowed to play the content must be stored in the watermark or in an external license file.

Multiple watermarks can be embedded to the same signal location if they have a low cross-correlation. The m-sequence is one example of a set of code vectors that exhibit low cross-correlation. [4]

The disadvantage of this method is that if many watermarks are embedded into the same location there is a risk that the quality of the content suffers as it is harder to embed many watermarks imperceptibly. The advantage is that the remaining number of usages can be extracted from one location and it is easier to embed multiple watermarks to the same location than to modify one, which was necessary in the second method.

III. Watermarking Method

The implemented watermarking embedding scheme watermarks the original audio signal, which is represented as a 16-bit sample sequence sampled at 44100 Hz, mono. The pseudo noise (PN) sequence is obtained from a pseudorandom number generator and represented in the bipolar form {-1,1}.

Prior to further processing, the PN sequence is filtered in order to adjust it to masking thresholds of the human auditory system (HAS) in the frequency domain. The main goal is to adapt the watermark to such a form that the energy of the watermark is maximized under the restriction of keeping auditory distortions to a minimum. The frequency characteristic of the filter is the approximation of the threshold in the quiet curve of the HAS.

Despite the simplicity of the shaping process of the PN sequence in frequency domain, the result is an inaudible watermark as the largest amount of the shaped watermark’s power is concentrated in the frequency sub-bands with lower HAS sensitivity. In addition, these frequency sub-bands (frequencies below 500 Hz and above 11 kHz) are an essential part of the watermarked audio and cannot be removed from its spectrum without causing a serious loss of the perceptual quality. Although standard frequency analyses have more accurate data about the audio spectrum, simulation tests done with selected audio clips showed a high level of similarity with the frequency masking thresholds derived from the masking model defined in ISO-MPEG Audio Psychoacoustic Model [5].

Host audio sequence is also analysed in the time domain, where a minimum or a maximum is determined in the block of audio signal that has the length of 5.6 ms. The goal of the temporal analysis is to place the watermark inside the raw audio without making any perceptual distortion to the host signal by using temporal masking characteristics of the HAS. The algorithm equally uses both pre- and post-masking properties, therefore making the most significant error if the maximum of the host audio is situated at the end of the analysed block. However, the impact of sub-maximums and the maskers from the contiguous blocks is not negligible and it helps the current masker in the masking process. As the result of this analysis, the samples of the watermark sequence are weighted, in order for them to be adjusted to psycho-acoustic perceptual thresholds.

Fig. 2 depicts the embedding of a watermark and Fig. 3 shows the extraction part of the algorithm. The resulting watermarked signal can be written as:

$$y(n)=x(n)+w(n)\cdot a(n)$$

Weighting coefficient a(n) is the output of the temporal analysis block. Furthermore, for spread factor c each bit $d(n)\in\{-1,1\}$ that is a part of watermark stream is being spread as:

$$s(n) = d(k),\ kc\leq n\leq(k+1)c$$

and the product

$$w(n)=s(n)\cdot f(n)$$

is formed, where f(n) is the PN sequence, filtered in order to be adjusted to the masking thresholds of the human auditory system (HAS) in the frequency domain.

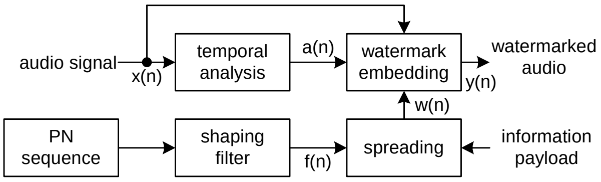

Fig. 2 Watermark embedding scheme

Before the watermarked signal is segmented into blocks in order to measure the cross-correlation with the PN sequence, the detection algorithm filters it with the whitening filter. Generally, in a correlation detector scheme it is often assumed that the communication channel is white Gaussian. Nevertheless, statistics for real audio signals show that audio samples are highly correlated. Applying a whitening procedure should considerably reduce any correlation in the audio and thus achieve optimum detection. In order to decrease correlation between samples of the audio signal, the algorithm uses least squares (Savitzky-Golay) smoothing filters [6], and forwards the residual signal (with increased signal-to-noise-ratio (SNR) value) to the correlator [7]. Savitzky-Golay filters with fourth order polynomial and 21 samples long time windowing were used during experiments.

The proposed detection procedure does not require access to the original signal to detect the embedded watermark. The cornerstone of the detection process is the mean removed cross-correlation between the watermarked audio signal and the PN sequence. Prior to calculating correlation, the PN sequence is shaped in frequency domain and whitened in order to achieve the optimal correlation values. The shaped and whitened PN sequence is also used in method 2.2 to partly remove the watermark from the watermarked audio. The correlator calculates mean removed correlation $c_{my}(m)$ between the residual signal **y*** and whitened PN sequence **m**:

$$c_{my}(m)=\begin{cases}\sum\limits_{n=0}^{N-|m|-1}\left(m(n)-\dfrac{1}{N}\sum\limits_{i=0}^{N-1}m_i\right)\left(y^*_{n+m}-\dfrac{1}{N}\sum\limits_{i=0}^{N-1}y^*_i\right), m\geq 0\\ c^*_{ym}(-m), m<0\end{cases}$$

The correlation method and the watermark extraction algorithm in general are reliable only if correlation frames are aligned with those used in watermark embedding. Therefore, one of the malicious attacks can be de-synchronization of the cross-correlation procedure by time-scale modifications. The method, which is robust against time scaling attacks, was chosen for this watermark extraction scheme; it uses redundancy in the watermark chip pattern, similar to the one described in [8]. The basic idea is to spread each chip of the shaped PN sequence onto R consecutive samples of watermarked audio. It has been proved [8] that using such an embedding and detection scheme, the correlation is correctly calculated even if a linear shift of floor (R/2) samples across the temporal or frequency domain is induced. However, there is a trade-off between the robustness of the algorithm and computational complexity, which is significantly increased by performing multiple correlation tests.

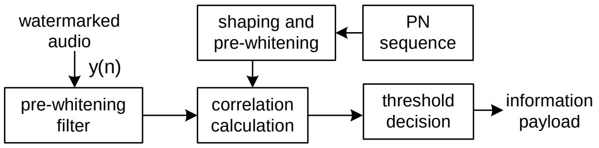

Fig. 3 Watermark extraction algorithm

IV. Watermark Attacking Experiments

In this section we present the results of the common attacks against the watermark that were obtained with our experimental solution.

A total number of 12 audio pieces were used as tests signals. Duration of the audio pieces ranged from 10 to 15 seconds, and a total of 5263 watermark bits have been embedded into them. The audio excerpts were selected so that they represent a broad range of music genres.

Subjective quality evaluation of the watermarking methods has been carried out by listening tests involving eight persons. In the first part of the test, participants listened to the original and the watermarked audio sequences and were asked to report dissimilarities between the two signals, using a 5-point impairment scale: (5: imperceptible, 4: perceptible but not annoying, 3: slightly annoying, 2: annoying 1: very annoying). The lowest impairment scale value recorded during the experiments was 3 and the average mean opinion score (MOS) for the tested audio excerpts was 4.57. In the second part of the experiments, participants were repeatedly presented with unwatermarked and watermarked audio clips in random order and they were asked to determine which one the watermarked clip is (blind audio watermarking test). A discrimination value near 50% indicates that the two audio clips (original audio sequence and watermarked audio signal) cannot be discriminated. The discrimination value obtained during the tests with the chosen 12 audio clips ranged from 43% to 56%.

The detection performance of the algorithm was also tested against common signal processing modifications [8]:

1. MPEG compression, bit rate 48 kbps mono, maximum bandwidth 10546 Hz.
2. Low-pass filtering using a second order Butterworth filter with cut-off frequency of 6 kHz.
3. Resampling consisting of subsequent down and up sampling to 22.05 kHz and 44.10 kHz, respectively.
4. Amplitude compression (8.91:1 for A>-29dB, 1.73:1 for –46dB<A<-29dB and 1:1.61 for A<-46dB).
5. Echo addition with a delay 100ms and decay 50%, respectively.
6. All-pass filtering using system function: H(z)=(0.81z2 - 1.64z + 1) / (z2 - 1.64z + 0.81).
7. Equalization (6-band equalizer, signal suppressed or amplified by 6 dB in each band).
8. Noise addition (with uniform white noise and maximum noise magnitude of 200 quantization steps).
9. Time scale modification between –3% and 3% of the total audio excerpt length.
10. D/A–A/D conversion using a commercial analogue tape recorder.

Detection results for the various attacks described above are shown in Tables 1, 2, and 3, which show the bit error rates (BER) for watermark extraction for given attack type.

TABLE I
THE METHOD 1 – WATERMARKING INTERVAL (BER)

Attack type / watermark bit rate (bps)	**2.70**	**5.38**	**10.75**
MPEG compression	$7.0\cdot10^{-3}$	$1.5\cdot10^{-2}$	$2.4\cdot10^{-2}$
Low pass filtering	$4.1\cdot10^{-4}$	$2.8\cdot10^{-3}$	$1.8\cdot10^{-2}$
Resampling	$6.0\cdot10^{-3}$	$1.2\cdot10^{-2}$	$2.1\cdot10^{-2}$
Amplitude compression	0	0	0
Echo addition	0	0	0
All-pass filtering	0	0	0
Equalization	0	0	$3.8\cdot10^{-4}$
Noise addition	0	0	$1.9\cdot10^{-4}$
Time scale modification	$1.5\cdot10^{-2}$	$1.8\cdot10^{-2}$	$1.8\cdot10^{-2}$
D/A–A/D conversion	$3.8\cdot10^{-4}$	$3.8\cdot10^{-4}$	$3.8\cdot10^{-4}$

The watermark was embedded only in one fourth of the given window; therefore the watermark bit rate is four times smaller than in the other two methods.

The second watermarking scheme is not able to completely remove the embedded watermark, because it is not able to estimate the phase of the embedded watermark. Therefore a part of the watermark remains in the content after the watermark removal process. These remains of watermarks become audible after embedding a maximum of 20-25 watermarks. The exact figure depends on the particular audio clip to be watermarked and can be exactly determined by subjective listening tests. This non-ideal watermark removal sets the maximum number of times that the watermark can be updated.

TABLE II
THE METHOD 2- SINGLE WATERMARK (BER)

attack type / bit rate (bps)	**10.80**	**21.50**	**43.00**
MPEG compression	$8.1\cdot10^{-3}$	$2.2\cdot10^{-2}$	$4.1\cdot10^{-2}$
Low pass filtering	$5.7\cdot10^{-4}$	$4.1\cdot10^{-3}$	$3.9\cdot10^{-2}$
Resampling	$7.4\cdot10^{-3}$	$1.9\cdot10^{-2}$	$2.9\cdot10^{-2}$
Amplitude compression	0	0	$1.9\cdot10^{-4}$
Echo addition	0	0	0
All-pass filtering	0	0	0
Equalization	0	0	$3.8\cdot10^{-4}$
Noise addition	0	0	$3.8\cdot10^{-4}$
Time scale modification	$1.6\cdot10^{-2}$	$1.9\cdot10^{-2}$	$1.9\cdot10^{-2}$
D/A–A/D conversion	$3.8\cdot10^{-4}$	$3.8\cdot10^{-4}$	$7.6\cdot10^{-4}$

In the third method it was presumed that 5 times is the maximum number in the counter. The maximum power is divided to those five watermarks and they are subsequently added to the host audio, since the power of added noise increases linearly. The results shown are only for the first watermark embedded, and because the watermark was embedded with five times less power than in the second method, the detection reliability is decreased. The maximum number of watermarks than can be embedded to the same location depends on the bit rate used, as it controls the processing gain achieved in spread spectrum scheme.

TABLE III
THE METHOD 3 - MULTIPLE WATERMARKS IN SINGLE LOCATION (BER)

attack type / bit rate (bps)	**10.80**	**21.50**	**43.00**
MPEG compression	$2.4\cdot10^{-2}$	$5.4\cdot10^{-2}$	$1.2\cdot10^{-1}$
Low pass filtering	$1.9\cdot10^{-2}$	$6.9\cdot10^{-2}$	$1.9\cdot10^{-1}$
Resampling	$1.8\cdot10^{-2}$	$3.4\cdot10^{-2}$	$9.4\cdot10^{-2}$
Amplitude compression	0	0	0
Echo addition	0	0	0
All-pass filtering	0	0	0
Equalization	0	0	$7.6\cdot10^{-4}$
Noise addition	0	0	$7.6\cdot10^{-4}$
Time scale modification	$1.8\cdot10^{-2}$	$1.8\cdot10^{-2}$	$1.7\cdot10^{-2}$
D/A–A/D conversion	$1.1\cdot10^{-3}$	$1.5\cdot10^{-3}$	$5.8\cdot10^{-3}$

V. OTHER ATTACKS

This section will discuss attacks that are not intended for removing the watermarks from the content but aim to bypass counter methods in other ways.

A. *Replay Attack*

The easiest way to attack the proposed counter application is by a method that we call replay attack. In this replay attack the user makes backup copies of the content before playing them. After the backup, the user can play the content freely as many times as allowed. When the number of usages has been exhausted, the user simply restores the backup copy of the content and the counter is reset.

Since the counter information is embedded in the content and the user is able to access the data freely, it is impossible to prevent the user from attacking this way. In fact it is not possible to securely store any data on the user's terminal, if the user is able to access it freely. The hardware-based solution could be used to make things much more secure. The trusted computing platform could have a secure storage area, where the player could store the counter data.

When storing the counter data embedded in the content, the user needs more storage space to be able to make backup copies. This would make the replay attack more impractical especially with mobile terminals that usually have limited storage space available.

B. Player Conformance

The watermark-based methods always require that a conforming player be used. Since the watermark alone does not prevent the user from playing the content as many times as he/she wants to, the player software must refuse playing it if there are no more usages left. This offers one kind of attack to a malicious user, where the user prevents the player from extracting the watermark and always enabling playback.

Frequent patching and updating can help close some of the vulnerabilities in the player. It is also important to design and implement the player so that cracking the software is harder. To make cracking harder the software should always run the whole code and not stop as soon as the cracking attempt is noticed, as this makes finding the security checks in the code harder.

Also here the trusted computing platform could be used to prevent user modifications to the player, if the operating system were to check that the player has not beet modified before allowing it to access securely stored data.

VI. Conclusion

This paper presents a novel watermarking application to restrict the number of times that the user is allowed to play content. Three different methods to implement this counter scheme were presented and some experiments were done with audio files against common attacks using spread spectrum scheme to embed watermarks.

The experiments suggest that the first method is the most robust against common attacks to watermarks, but it has the drawback that less data can be embedded than with the other two, since only a part of the whole content is used in watermarking. Also, if the length of the watermark segment is fixed, the length of the watermarking interval depends on the number of usages granted to the user. This method renders the content unusable after the maximum number of usage has reached, i.e. the user must download the content again to renew the license.

In the second method the number of usages can be found from a fixed location, but the player must be able to remove the watermark in order to update the counter. This can also help the attacker to remove the watermark. Removing watermarks leaves residual energy in the content and this restricts the number of times that counter can be updated. Experimental results showed this to be around 20-25 updates. This method has the advantage that the user can renew the license until the cumulative number of counter values becomes 20-25. In other words, if the user first buys 5 usages, he/she can buy 5 more without downloading the whole content again.

The third method has a bigger bit error ratio than the other two methods, since the watermarks must be embedded with lower power to keep them imperceptible. The advantages of this method are that more data can be embedded and watermarks do not need to be removed.

Finally we discussed attacks against the counter scheme, in which the user can make backup copies of the content and later restore them and thus reset the counter to the previous state. It is not possible to prevent this attack if the user is able to freely access the data stored in the terminal and no online connection is used. This attack can be impractical with terminals that have limited storage space available. The use of watermarks to control whether or not the user is allowed to play the content requires that the player enforce that the rules are followed.

References

[1] I.J. Cox, M.L. Miller and Bloom J.A. "Watermarking applications and their properties" Proc. International Conference on Information Technology: Coding and Computing, Las Vegas, NV, 2000, pp. 6–10.

[2] J.A. Bloom, I.J. Cox, T. Kalker, J.P. Linnartz, M.L. Miller and C.B.S. Traw "Copy protection for DVD video", *Proceedings of the IEEE*, Vol. 87, No. 7, pp. 1267–1276.

[3] E. Becker, W. Buhse, D. Günnewig and N. Rump "Digital Rights Management: Technological, Economic, Legal and Political Aspects", *Springer Lecture Notes on Computer Science*, Vol. 2770, p. 805.

[4] I.J. Cox, J.A. Bloom and M.L. Miller "Digital Watermarking: Principles & Practice", Morgan Kauffman Publishers, p. 542.

[5] ISO/IEC IS 11172, "Information technology – coding of moving pictures and associated audio for digital storage up to about 1.5 Mbits/s"

[6] Cvejic N. and Seppänen T "Audio prewhitening based on polynomial filtering for optimal watermark detection", Proc. European Signal Processing Conference, Toulouse, France, 2002, pp. 69-72.

[7] S. J. Orfanidis, "Introduction to Signal Processing", Prentice-Hall, Englewood Cliffs, NJ, 1996.

[8] Kirovski D. and Malvar H., "Spread-spectrum watermarking of audio signals", *IEEE Transactions on Signal Processing*, Vol. 51, No. 4, pp. 1020–1033.

[9] Haitsma J., Van Der Veen M., Kalker T. and Bruekers F., "Audio watermarking for monitoring and copy protection", Proc. ACM Multimedia Workshop, Los Angeles, CA, 2000, pp. 119–122.