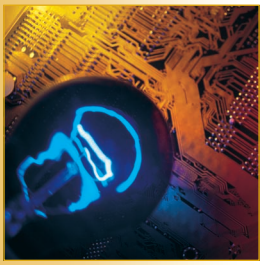

Managing Copyright in Open Networks

Integrating cryptography with watermarking technologies can provide intellectual property rights protection in an open network environment such as the Internet.

Alessandro Piva and Franco Bartolini
University of Florence

Mauro Barni
University of Siena

Despite the ease with which digital data owners can now transfer multimedia documents across the Internet, current technology does not let them protect their rights to the works. In fact, although the Internet permits widespread dissemination of interactive services such as remote database access, archival browsing, and electronic commerce, the easy-to-copy nature of digital data limits data owners' willingness to distribute their documents electronically. Thus, the need for an electronic copyright management system (ECMS) that protects intellectual property rights (IPR) in open-network environments continues to grow.

Network security issues are classically handled through cryptography;[1] however, cryptography ensures confidentiality, authenticity, and integrity only when a message is transmitted through a public channel, such as an open network. It does not protect against unauthorized copying after the message has been successfully transmitted.

Digital watermarking is an effective way to protect copyright of multimedia data even after its transmission.[2,3] A watermark, embedded in the data, can uniquely identify the document's owner or authorized user. The main problem with using watermark technology for IPR protection, however, is its *reversibility*. Anyone who can read or detect the watermark can also remove it by inverting the watermark process. Our open-network ECMS combines watermarking with cryptography to achieve reliable copyright protection while satisfying two contrasting requirements:

- Actors in ECMS transactions must be able to verify that the watermark granting their rights is truly embedded in the multimedia document.
- Actors (other than the author) must not be able to remove the watermark.

In this article, we discuss digital watermarking and describe our integrated ECMS

Electronic Copyright Management Systems

Electronic copyright management systems automatically manage issues related to trading multimedia documents through open communication networks. An ECMS can be considered an ensemble of services, connected through a network environment, cooperating to allow intellectual property rights (IPR) protection of multimedia data.

Several projects are under way to develop ECMSs. The most recent MPEG standardization effort (MPEG-21), for example, aims to establish rules and protocols for permitting the legal and reliable exchange of IPR-sensible multimedia documents.

We distinguish two approaches to designing effective ECMSs:

- preventing copyright violations (IBM's Cryptolope, www-3.ibm.com/software/security/cryptolope, for example)
- tracking copyright violations (the EC-funded Imprimatur, www.imprimatur.net, for example).

Both approaches require authoring tools to properly prepare multimedia documents before distributing them.

Cryptography-based ECMSs

In a cryptography-based ECMS, the author wraps the digital object in an encrypted system and integrates it with an application (the reader). Because users cannot access the content without the proper application, the owner can control how the document is used—for example, a user can display the images but not print them, or play the audio files but not save them.

The main disadvantage of this approach is the difficulty of establishing a standard for embedded applications. Moreover, when a multimedia document finally reaches the end user (for example, it appears on a PC screen or is played by a digital recorder), it can still be captured and copied without constraint. Liquid Audio (www.liquidaudio.com) is an example commercial system.

Watermark-based ECMS

A watermark-based ECMS tightly and robustly embeds IPR-related information into purchased digital objects (the hidden data can be the name of the copyright owner or a unique code identifying the document). Watermarking can also be used to hide the identification of the authorized distributor or buyer (the more correct term for this is *fingerprinting*) inside the document. It is thus always possible to check the document's legal status, and to track the path IPR-infringing material follows through the network.

A main limitation of current watermarking technologies is their reversibility; that is, anyone who can read or detect a watermark can remove it. Only the effective development of asymmetric watermarking methods, which still seem far off, will overcome this intrinsic limitation. On the other hand, watermark-based IPR management does not require users to adopt a particular format for the watermarked multimedia content, because IPR data are directly injected into the content itself.

approach. We also introduce our prototype system, available at http://lorenzo.det.unifi.it, to show the approach's viability. The sidebar, "Electronic Copyright Management Systems," discusses current technologies for IPR management over open networks.

Digital Watermarking

In digital watermarking, a digital code, or *watermark*, is embedded into a document so that a given piece of information, such as the owner's or authorized consumer's identity, is indissolubly tied to the data. This information can later prove ownership, identify a misappropriating person, trace the marked document's dissemination through the network, or simply inform users about the rights-holder or the permitted use of the data.

Watermarking does not solve all IPR problems, however,[4,5] and most researchers agree that the technology is less mature than cryptography. Still, its potential to provide reliable protection is already attracting copyright holders.

Watermarking Algorithms

Several watermarking schemes have been introduced, and a great deal of research has sought to develop data-labeling techniques that are robust against the most common attacks and multimedia processing manipulations. Little attention has been given to protocol-level analysis, however. The sidebar, "Related Copy-Deterrence Protocols," on page 20, discusses some work in this area.

Because how a watermarking algorithm recovers the watermark from the data determines which technique will be used in a given situation, we classify digital watermarking techniques by their decoding processes.

- *Blind versus not blind.* A watermarking algorithm is *blind* if it does not need to compare the marked and unmarked documents to recover the watermark. Conversely, a watermarking algorithm is *not blind* if it needs the original data to extract the information from the watermark. Blind techniques are sometimes referred to as oblivious or private.
- *Private versus public.* A watermark is *private* if only authorized readers can detect it. Not-blind techniques are private because only authorized users can access the original data needed for watermark reading. We extend the concept of

Related Copy-Deterrence Protocols

Various copy-deterrence protocols combining watermarking and cryptography have been proposed. Lintian Qiao and Klara Nahrstedt propose an owner-customer watermarking protocol, in which a customer sends the owner an encrypted version of a predetermined code.[1] After receiving the code, the owner embeds the encrypted sequence into a copy of the image as a watermark and transmits the copy to the buyer. Because no one else knows the decryption key, the buyer can prove legitimate ownership of the copy. The protocol does not link the customer to the purchased copy, however, so unauthorized copies cannot be traced. In fact, a counterfeiter can claim that an unauthorized copy was created by the seller or caused by a security leak in the system.

Nasir Memon and Ping Wah Wong propose a buyer-seller protocol in which the seller does not know the buyer's watermark, and so, cannot create copies of the image containing it.[2] The watermarking protocol is based on public key cryptography and requires a watermark certification authority. This model does not let the buyer verify that a watermark proving ownership is truly embedded in the copy.

These models do not allow each actor to check that the data exchange was carried out correctly and, at the same time, verify that the current holder is using the data legally. This is the main novelty of our proposed approach.

References

1. L. Qiao and K. Nahrstedt, "Watermarking Schemes and Protocols for Protecting Rightful Ownership and Customer's Rights," *J. Visual Comm. and Image Representation*, vol. 9, no. 23, Sept. 1998, pp. 194-210.
2. N. Memon and P.W. Wong, "A Buyer-Seller Watermarking Protocol," *IEEE Trans. Image Processing*, vol. 10, no. 4, Apr. 2001, pp. 643-649.

privateness to techniques using any mechanism to prevent unauthorized personnel from extracting the watermark. If anyone can read the watermark, we call it *public*.

- *Readable versus detectable.* We also distinguish between algorithms that embed a code users can *read* without knowing the content in advance, and those that insert a mark that can only be *detected* — that is, a user can only verify that a given code is in the document. Watermarks that are encrypted before they are embedded are even harder to detect. Detectable watermarking is sometimes referred to as 1-bit watermarking because the detector output is just "yes" or "no."

Not-blind methods are more robust to attacks than blind methods, because the original content can be used in detection to estimate possible modifications introduced by an attacker to remove the watermark or make it unreadable. Very often, however, the original document is not available, making not-blind algorithms unsuitable for many practical applications. Moreover, private mechanisms tend to be significantly more robust than public ones: an attacker can easily remove or make unreadable a known watermark. Because detectable watermarks are intrinsically private, it follows that blind, detectable systems are more robust than other schemes.

Reversibility

A watermark is reversible if, once read or detected, it can be removed from the document, or at least made unreadable or undetectable. Virtually all existing techniques are potentially reversible. Indeed, because watermarks must be invisible, the modification introduced by the watermarking process is very small and thus linearizable and consequently invertible. Therefore, anyone who can read or detect the watermark can also remove it.

This conflicts with our requirement that a legal buyer have the right to check that his or her name is truly embedded in the multimedia document. Watermark reversibility allows a buyer who can check for watermark presence to also remove it, and possibly reuse the document illegally by embedding a forged watermark.

An asymmetric watermarking algorithm might overcome reversibility issues.[6] In asymmetric watermarking, watermark detection and decoding reveals only part of the secret used to embed the watermark (the public key); the private key remains hidden. Requiring the private key for watermark removal prevents reversibility problems. Asymmetric watermarking is a very immature field, however, and researchers are still not sure whether it can be used for secure public watermark detection. Moreover, asymmetric schemes embed a very small amount of information into a document and thus are not suited for complex ECMS applications. Rather, we expect they will be used to manage document copies, where a lower capacity is required.

The ECMS presented in this article is explicitly designed to overcome the problems deriving from watermark reversibility. We assume the use of a detectable watermarking scheme because such techniques are more robust and reliable than readable schemes.

An Integrated Approach to IPR Protection

We have developed a watermark-based ECMS that integrates cryptography to compensate for the weaknesses of watermarking schemes and to achieve reliable copyright protection.

Trading multimedia documents in an open-network environment involves many actors — the document author or authors, an editor, a media distributor, buyers, and so on. It also involves electronic payment issues, such as information security and customer privacy. To simplify our presentation, we limit the number of actors and do not address payment or privacy issues here.

Transaction Model

Figure 1 shows a simplified trading model. Annie, the author of a multimedia document, registers her document and deposits a copy of it with a collecting society. She then contacts a media distributor, McDarrel, who makes her document available on the network, where Peter accesses and buys it. For simplicity, we assume the CS is a trusted third party that will ensure that the protected documents are traded correctly. Note that the transaction between the buyer and the media distributor also involves an exchange of data with the CS.

In our approach, the document is self-contained. At any given instant it contains all the information needed to verify whether the current holder is using the data legally. No attempt is made to trace the document history, however, either by watermarking the document each time the owner changes, or by recording transaction details in a register. We take particular care to allow each actor to check that the data exchange was carried out correctly.

The basic principle underlying our ECMS strategy is that the data holder's name must be watermarked into the data to prove legal ownership. To ensure that a document is being used legally, any authorized person can check the watermark field the holder's name is written in. We also envision a protocol-level mechanism that addresses the reversibility problem by preventing data holders or counterfeiters from benefiting from watermark removal: at no step of the transaction can a counterfeiter insert a fake watermark, so a counterfeiter cannot prove document ownership. To keep misappropriating persons from writing their names into the data, the ECMS assumes that the seller (or the author when a media distributor sells the document) embeds the watermark.

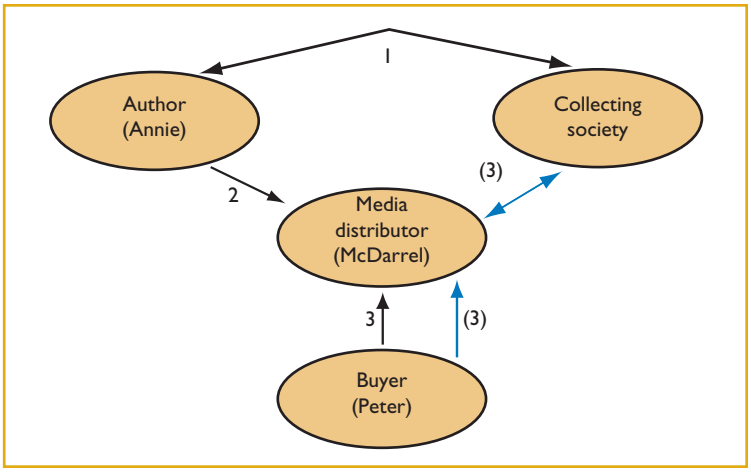

Figure 1. A simplified transaction model. (1) An author registers a new document with a collecting society. (2) The author sends a copy of the document to a media distributor for dissemination. (3) A buyer contacts the media distributor and purchases a digital copy of the document.

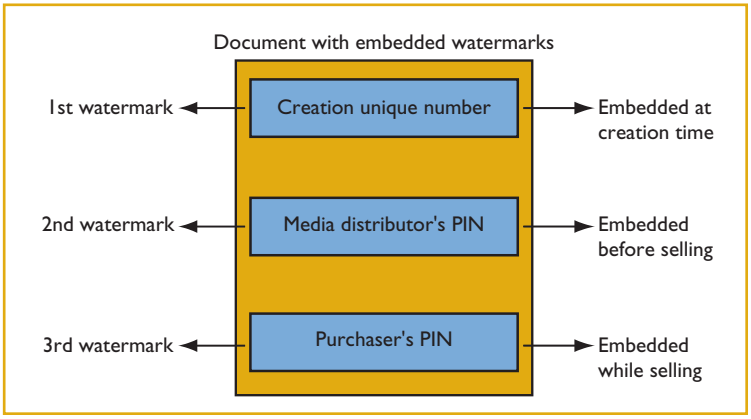

Figure 2. A document with embedded watermarks. Our ECMS uses three watermarks: the first refers to the creation identity; the second contains the media distributor's personal identification number (PIN); and the third identifies the buyer.

Verifying Ownership Rights

As Figure 2 shows, the document contains three watermarks embedded into the data at different times. We use blind, detectable watermarking and reversible watermarks. Although similar watermarking algorithms could be used to implement the proposed ECMS, it is beyond the scope of this article to investigate them. A companion article in *IC Online* (www.computer.org/internet/v6n3/ecms.htm) details the watermarking method used to implement our prototype ECMS. Figure 3 (next page) illustrates the transactions involved in selling a multimedia document.

Author identifier. When Annie registers a document in the CS, she also embeds into the data a cre-

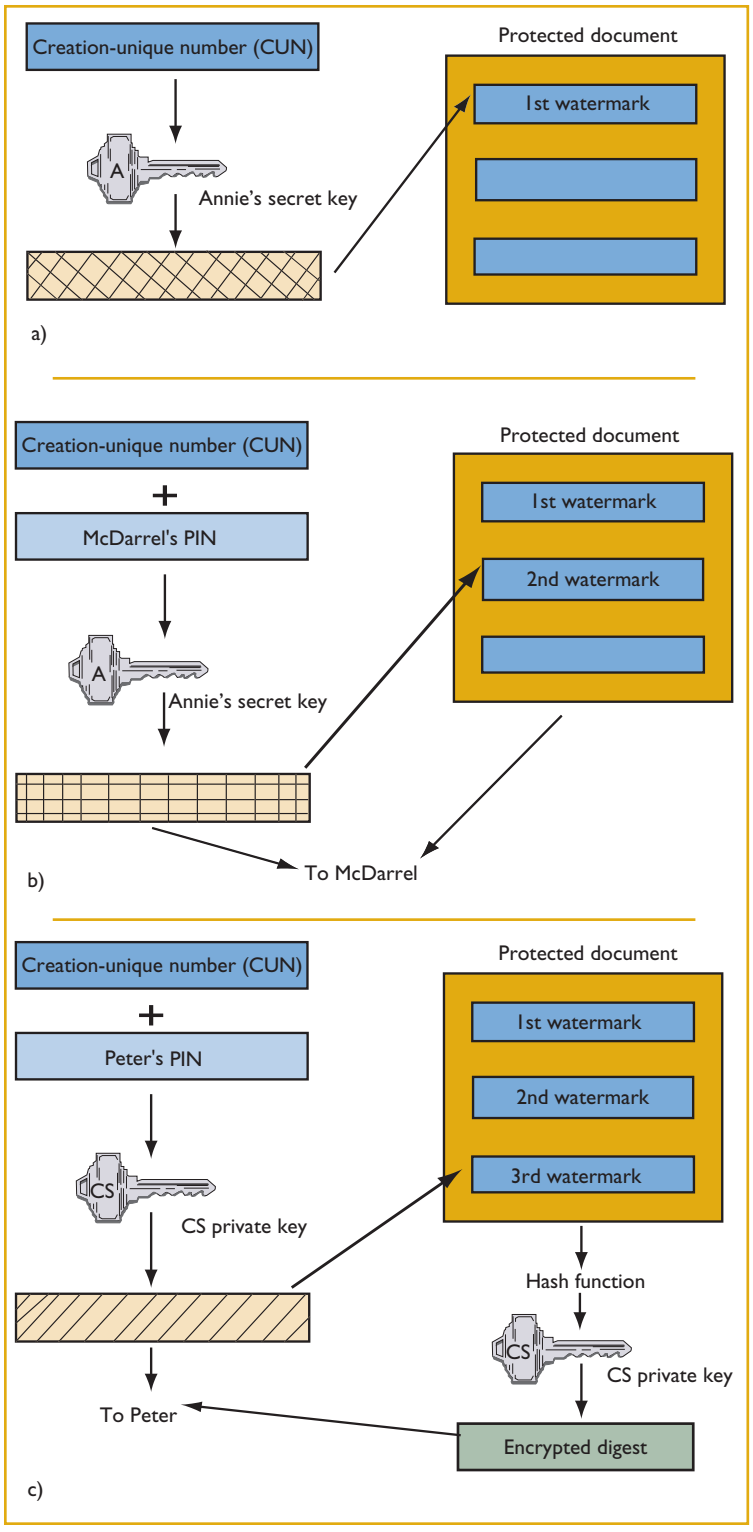

Figure 3. Transactions involved in selling a multimedia document. (a) The document author, Annie, embeds the first watermark, containing a creation-unique number encrypted with her secret key. (b) Annie embeds the second watermark, which contains the CUN and the media distributor's personal identifier encrypted with her private key. (c) The media distributor inserts the third watermark, which contains the document CUN and the buyer's PIN encrypted with the collecting society's private key.

ation-unique number (CUN), which unambiguously identifies her document. To prevent anyone from reading the watermark with the CUN and exploiting watermark reversibility to remove it, Annie encrypts the CUN before casting. We use symmetric key encryption, but we could also use an asymmetric scheme (for example, we could use the same private key used for the second watermark) at this stage. Annie then deposits a copy of the watermarked document into the CS archive. Figure 3a shows the steps involved in this transaction.

The first watermark will allow a trusted control authority to verify the original owner of a multimedia document. We assume that the document can be identified as belonging to Annie in some other way (by visual inspection, for example), given that a detectable watermark only allows the control authority (CA) to check for the CUN, not to guess it.

Distributor personal identifier. If Annie wants to sell copies of her document through a media distributor, she embeds a second watermark into the document. This watermark contains a personal identification number (PIN) identifying the media distributor, McDarrel, and the document's CUN. Annie encrypts the watermark string with her private key and a copy of the encrypted string, which McDarrel can use to verify that Annie really inserted his name into the document. McDarrel can use Annie's public key to read the encrypted string, and watermark detection software to verify it. (Unlike with the first watermark, only an asymmetric cryptography scheme can be used here.) Figure 3b illustrates this transaction. Note that because McDarrel knows the watermark content, he can use detectable watermarking.

Watermark reversibility is not a problem here: if McDarrel erases the watermark from the document, he cannot prove his right to sell it. In addition, because Annie encrypted McDarrel's name with her private key, no one can counterfeit the second watermark. Moreover, inserting the CUN into the second watermark prevents McDarrel from embedding the encrypted string into other documents of Annie's he does not have permission to sell. To prove his right to sell the document, McDarrel must demonstrate that the CUN contained in the second watermark matches the CUN in the first.

Of course McDarrel could embed another CUN on behalf of a fake author into the document. To get the new CUN, he must deposit a copy of the newly watermarked document at the CS. Because this new CUN would be issued after the original one, time ordering would allow Annie to prove

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.