Chapter II
Digital Rights Management for Streaming Media

Deepali Brahmbhatt
San Jose State University, USA

Mark Stamp
San Jose State University, USA

ABSTRACT

This chapter presents a digital rights management (DRM) system designed for streaming media. A brief, general introduction to DRM is also provided, along with a discussion of the some specific issues that arise in the context of streaming media. The DRM system proposed here has been implemented and some details related to the implementation are discussed.

INTRODUCTION

For the purposes of this paper, digital rights management (DRM) can be viewed as an attempt to provide “remote control” over digital content. That is, DRM is supposed to make it possible to securely deliver digital content and to restrict the actions of the recipient after the data has been delivered.

Consider, for example, a digital book. Certainly, the publisher would like to deliver such a book over the Internet, since there is an enormous potential market and the costs of reproduction and delivery are negligible.

However, if an attacker can redistribute a perfect digital copy of the book, then the rate of piracy would almost certainly be intolerable—if not virtually 100%. An ideal DRM system (from the perspective of the copyright holder) would prevent the redistribution of the book in an unprotected form, and perhaps also enforce other restrictions, such as “no printing.” Such an ideal DRM system is impossible, as noted in the indispensable paper by Biddle, et. al. (2002) and
Digital Rights Management for Streaming Media

in Stamp (2002). The interesting question then is, what is the best practical DRM system? What is “best” depends on many factors, not all of which are strictly technical in nature. For example, the value of the content being protected, the technical sophistication of typical users, the overall business model, and the credibility of threats of legal reprisals greatly effect the utility of any DRM system; see Stamp (2003b) for a discussion of these and other non-technical DRM issues.

It is important to note a few salient issues concerning DRM. First, the fundamental requirement of a DRM system is that restrictions must be enforced after the content is delivered to the intended recipient. Since these restrictions must stay with the data wherever it goes, the buzzword for this DRM requirement is “persistent protection.” This is in contrast to the usual cryptographic scenario, where the goal is simply to securely deliver the bits.

Second, the legitimate recipient is a potential attacker. This is also in stark contrast to the usual cryptographic situation, where the intended recipient (i.e., the entity that holds the corresponding decryption key) is a “good guy”, not an attacker.

Third, cryptography is necessary, but far from sufficient for useful DRM. It is necessary to encrypt digital content in order to securely deliver the bits, and this is a part of any DRM system. However, securely delivering the bits is the easy part of DRM.

Cryptography alone is insufficient for effective DRM—to render the content, the recipient must access to the key, and the recipient is a potential attacker. Cryptography was not designed to solve this problem, where, in effect, we must give the attacker the key. The essence of DRM security can therefore be reduced to playing “hide and seek” with cryptographic keys, although this fundamental fact is not always clear from the descriptions of fielded or proposed DRM systems.

It has been shown that it is not trivial to effectively hide a key in data (Shamir and van Someren 1999) and that software obfuscation techniques are not sufficient to hide a key in software (Jacob, Boneh, and Felten 2003). In any event, this game of hide and seek is at the core of any DRM system. This topic is explored further in the next section.

Finally, it is important to note that there is an absolute limit on the utility of any DRM system. The digital content must ultimately be rendered, and at that point it is subject to capture in analog form. Even if a perfect DRM system were available, digital music, for example, could be recorded using a microphone when it is rendered. In our digital book example, an attacker could take digital snapshots of the pages of the book when it is displayed on a computer screen. This is the so-called “analog hole” (Doctorow 2002), which is obviously present in any DRM system. These sorts of analog attacks are beyond the scope of the DRM system. But since such attacks likely result in a loss of fidelity as compared to the original digital content, they are, perhaps, not as serious of a concern as a successful attack on the DRM system itself. Consequently, the goal of a DRM system is to prevent the attacker from obtaining an unprotected and high quality digital copy of the original.

In the next section we give a brief description of some of the techniques commonly employed in DRM systems. Our focus is on software-based systems, but we do mention some of the inherent advantages and disadvantages of hardware-based protection. Then we turn our attention to our proposed software-based DRM system for streaming media. A prototype of this proposed system has been implemented and we briefly discuss some details of the implementation. We then consider likely attacks on our proposed system as well as open problems in the field of DRM.

DRM TECHNIQUES

As mentioned in the previous section, the fundamental DRM problem is the protection of a
Related Content

Location-Aware Caching for Semantic-Based Image Queries in Mobile AD HOC Networks
[www.igi-global.com/article/location-aware-caching-semantic-based/64629?camid=4v1a](www.igi-global.com/article/location-aware-caching-semantic-based/64629?camid=4v1a)

Expert-Novice Differences and Adaptive Multimedia
[www.igi-global.com/chapter/expert-novice-differences-adaptive-multimedia/8429?camid=4v1a](www.igi-global.com/chapter/expert-novice-differences-adaptive-multimedia/8429?camid=4v1a)

Critical Gameplay: Design Techniques and Case Studies
[www.igi-global.com/chapter/critical-gameplay-design-techniques-case/50736?camid=4v1a](www.igi-global.com/chapter/critical-gameplay-design-techniques-case/50736?camid=4v1a)

ISEQL, an Interval-based Surveillance Event Query Language
Sven Helmer and Fabio Persia (2016). *International Journal of Multimedia Data Engineering and Management* (pp. 1-21).
[www.igi-global.com/article/iseql-an-interval-based-surveillance-event-query-language/170569?camid=4v1a](www.igi-global.com/article/iseql-an-interval-based-surveillance-event-query-language/170569?camid=4v1a)