12

Digital Rights Management

Paul Koster[1] and Willem Jonker[1,2]

[1] Philips Research
[2] University of Twente

12.1 Introduction

Digital Rights Management, or DRM for short, is a much-discussed topic nowadays. The main reason for this is that DRM technology is often mentioned in the context of protection of digital audio and video content, for example to avoid large scale copying of CDs and DVDs via peer-to-peer networks in the Internet. However, DRM technology is much more than a simple copy protection technology. It is one of the enabling technologies that open the way to secure distribution and exchange of digital content over open digital infrastructures such as the Internet.

In order to show how DRM addresses this challenge, we will discuss what DRM technology actually is. There are two main lines of DRM technology based on two different approaches to the problem. The first approach is preventive, while the second approach is reactive.

12.1.1 Preventive DRM Technology

Preventive DRM technology aims at preventing behavior that violates the regulations. The technology is based on encryption of the content. The encrypted content can only be accessed through an encryption key. The use of this key is regulated by so called usage rights. A typical electronic distribution system consists of a client-server system. At the server side the content is encrypted and sent to the client. The client needs to be in possession of both the key and the usage right to access the content. The DRM software that runs on the client checks that this is the case. The key and usage right together are typically contained in a data object that we call license. More details and examples can be found in the section on DRM architecture and the case.

12.1.2 Reactive DRM Technology

Reactive DRM technology aims at tracing of behavior that violates the regulations. The approach is also called forensic tracking. The technique that is commonly used is that of embedding information in the content itself that

allows tracing the origin of the content. The main technology that is exploited in this context is that of watermarking. Watermarking allows inserting information in music or movies in such a way that consumers do not perceive any difference from the original. It is very difficult to remove or detect a watermark when the characteristics of the watermark are not known. A typical reactive DRM system consists of a server that inserts the watermark containing information on the client at the moment a client downloads content. Violations can be detected by using a watermark detector. Such a detector may, for example, be used to monitor content distribution in the network. If, for example, a usage rule does not allow a client to redistribute the content and the content is nevertheless spotted in the distribution network, the watermark can be used to trace the client that originally downloaded the content.

12.1.3 Relation to Other Chapters

This chapter discusses the protection of multimedia data using DRM. It also positions DRM in the Multimedia Information Retrieval System architecture presented in Chapter 1. In this extended architecture both the content server and the client are extended with DRM functionality. DRM introduces the concept of licenses, which may be regarded as metadata. The concept of metadata is introduced in Chapter 2, which provides an overarching framework to ensure interoperability of digital multimedia objects, including protection and management of rights.

12.1.4 Outline

In the remainder of this paper we concentrate on preventive DRM systems. The next section discusses the context in which DRM operates such as the legal framework and the applications areas for DRM. Section 12.3 describes the general DRM architectural principles. Section 12.4 discusses a case to highlight a number of technical aspects relating to DRM. As an example the Personal Entertainment Domain (PED) DRM concept is chosen. We focus on the person-based and domain-based aspects of PED-DRM. We conclude with further reading and a summary.

12.2 DRM Context and Application Areas

12.2.1 DRM and the Legal Framework

It is important to note that DRM is more than technology alone. DRM technology functions in the context of a legal framework that outlines the regulations that DRM technology supports to enforce. Examples of such legal frameworks are copyright laws, privacy laws and antitrust laws.

Copyright law differs per jurisdiction although mostly the same principles are present. The background of these principles can be found in an international treaty called the Berne Convention for the Protection of Literary and

Artistic Works. In daily life most relevant are the Digital Millennium Copyright Act (DMCA) in the United States, and the European Union Copyright Directive that is used as a basis for copyright law in the EU countries. New provisions in the DMCA and EUCD also address DRM technology by outlawing circumvention technology.

12.2.2 DRM for Secure Audio and Video Content Management

The secure management of audio and video content is an important application area for DRM. The fact that digital audio and video content can be easily transported over electronic networks opens the way for electronic delivery of music and movies. Both consumers and content owners are interested in exploiting this new way of content distribution. For example a networked version of a video rental store would be advantageous to both consumers (that do not need to drive to the rental store) and content owners (that will rent more videos due to a lower threshold). However, in this example there is one issue: how to make sure that the consumer does not watch the video any longer after the rental period is over? Of course this problem also exists with physical distribution, since the consumer can make a copy of the video at home before returning the original to the shop. However, due to the ease of digital content distribution, the impact of such behavior is much larger in a digital world, something that was clearly demonstrated by the peer-to-peer networks already mentioned before. As a result the development of electronic music and video distribution services is taking up slowly.

12.2.3 Standardization and Products

There are several activities going on around the standardization of DRM technology. Important activities are taking place in DVB (Digital Video Broadcasting) for the secure delivery of digital TV and for the secure sharing of this in home networks [32], in OMA (Open Mobile Alliance) for the secure delivery of music and video to devices including mobile phones [22], in Marlin JDA (Joint Development Association) [21] focusing on efficient implementation of DRM in consumer electronics devices, in the Coral Consortium focusing on DRM interoperability (i.e., solving the problem of content exchange between different DRM systems) [5], and in MPEG-21 focusing more broadly on the secure exchange of digital items [16].

Next to standardization a number of proprietary DRM systems exist, the best known currently is FairPlay that comes with Apple iTunes, but also Microsoft with its Windows Media DRM (WM DRM) technology is offering DRM functionality as well as Sony with its Open Magic Gate system and RealNetworks with its Helix DRM.

12.2.4 DRM in Other Areas

Although the current application focus of DRM is on secure delivery of music and movies, DRM can be used in a much wider range of applications. It can be

used to protect any digital document, and as such it can be used to implement secure document management or workflow systems for example.

Enterprise DRM is one of such applications. The focus lies on protecting company documents such that only authorized people have access to their contents. Important players are Microsoft with its extensible Windows Rights Management Services, which is also used by other companies as a technology platform, and Adobe with its Adobe Live Cycle Policy Server product.

Another interesting application domain is healthcare. Healthcare has strict regulations with respect to privacy of medical data, e.g., the Health Insurance Portability and Accountability Act (HIPAA) in the US. At the moment we see a starting digitization in healthcare. Increasingly, medical information is becoming available in digital form. Already inside hospitals medical information is managed by departmental information systems, and hospital information systems emerge. The next step will be the exchange of medical information between hospitals and all kinds of parties involved in the healthcare processes, leading to the creation of electronic health records containing a lot of privacy sensitive information. DRM technology has the potential of becoming a key technology for the secure exchange of all kinds of medical information. Research in this field is emerging [27] and some DRM vendors start to address this. For example Microsoft presents their Windows Rights Management Services as a solution for protecting electronic content in Healthcare, and Sealed-Media offers a similar proposition with its solutions targeted at healthcare applications. Both solutions advertise their audit facilities next to preventive DRM methods.

Different DRM applications share the basic technical principles, although aspects may differ. For example DRM for audio/video content is often device oriented meaning that certain devices are authorized to access content, while enterprise DRM is often more identity or user oriented, and medical DRM typically has special measures to support emergency cases.

12.3 DRM Architecture and Technology

Figure 12.1 depicts the generic DRM system architecture. The essential information exchanged between components are content and licenses. The policies that control content use are defined by the rules of the DRM system itself and by the licenses.

Separation between content and licenses is a core characteristic of DRM. This characteristic is present in all major commercial systems like WM DRM and OMA DRM. That said, many systems support embedding of licenses in the content container to make content use more convenient. The main benefit of separating licenses and content is that it allows for a wide variety of distribution and business models, while still having an efficient system with respect to bandwidth, storage and processing requirements at servers and networks.

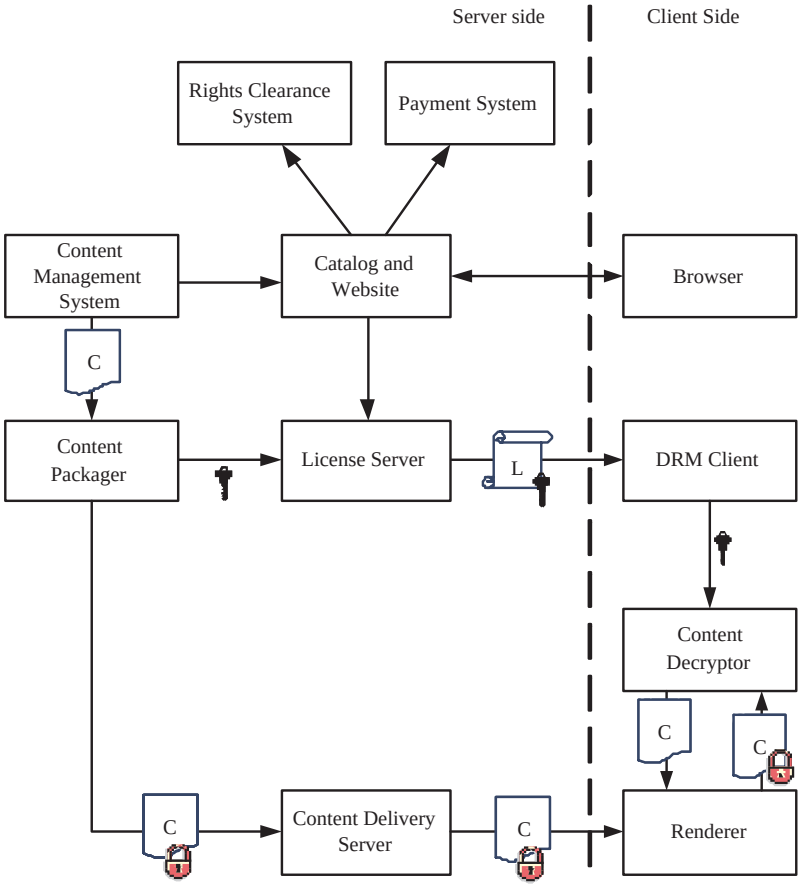


Fig. 12.1. General DRM architecture.

Content packager, license server, DRM client and content decryptor are the main DRM components. Content packager is responsible for protecting the content and fitting it in a DRM format. License server issues licenses after content is bought. DRM client interprets a license and makes the content key available to the content decryptor to decrypt the content for the renderer.

Next to the core DRM components, front-end components play a role such as Web browsers and Web shops with a catalog and ordering system. Also backend systems play a role such as rights or royalty clearing systems that facilitate payment of the copyright holders, and payment services that serve as intermediaries for payments by end-users.

In reality the client side is more complex than illustrated in the figure. Instead of just one DRM client people have multiple devices with heterogeneous capabilities. For example, some devices such as a PC can buy licenses, while other devices such as portable music players cannot. Naturally, users want to use their content on all their devices. This implies that content and licenses must be distributed to these devices and use of the content on these devices must be authorized. To address these issues concepts are introduced like domains, tethered devices, and person-based DRM. Section 12.4 elaborates on a number of these aspects.

From now on we mainly focus on the DRM functionality at the client and server side. Commercial front-ends as shops and payment are not considered further, while backend DRM functionality like content packaging only get minimal attention.

Figure 12.2 depicts the DRM functions and the relation between content and licenses. This approach achieves enforcement of the intended policy set by the content shop. From a data management perspective content has the role of data, while all supporting information such as licenses, keys and identifiers are metadata. The metadata facilitates data management and policy enforcement on the data. The following sections describes in more detail how content management and license management achieves content security and enforcement of the intended policy.

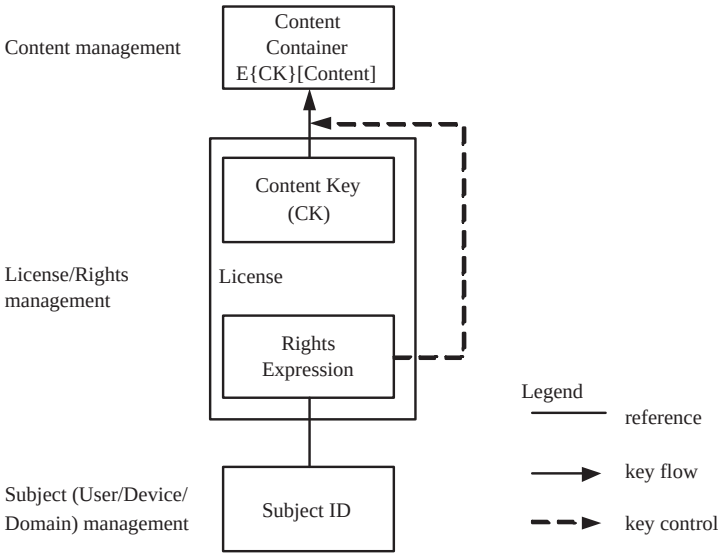


Fig. 12.2. Functional and informational DRM architecture.

12.3.1 Content Management

Content management in the context of DRM comprises the protection of content, the handling of content through the system, e.g., moving, copying and accessing audio/video/text assets, and the insertion of new content in the system.

Content protection has as a goal to prevent unauthorized access to content. One method is content protection during storage or transmission. The content is embedded in a secure content container by a process called content packaging. The main purpose of the content container is to offer confidentiality by means of encryption of the content using a content encryption key. Many content container definitions exist, typically one for each proprietary DRM system, but also standards exist such as ISMACryp [14], MPEG-2 Transport Streams (TS) [6] and OMA DCF (DRM Content Format) [23]. ISMACryp is intended for streaming content and makes use of the secure RTP (Real-Time Protocol) on top of IP for the content, and RTSP (Real-Time Streaming Protocol) and SDP (Session Description Protocol) for control and key management. Encryption is applied on packets with MPEG content, while authentication is done at transport level. MPEG-2 TS as defined by DVB is typically used for conditional access pay TV and also encrypts at the content level. DCF is a content file format that contains the encrypted content, some content metadata such as identifiers, and metadata related to DRM such as licenses. DCF has a profile for discrete media such as pictures which can be used for any type of content, and a profile for continuous or streaming content such as audio and video content. DCF also offers integrity protection to the content. Integrity and authentication of protected content may serve the end-user who is assured that he gets what he paid for, but also the content and service providers who have a means to control who can insert content into the DRM system infrastructure.

The handling of protected content is not much different than the handling of unprotected content, because licenses and content are separated. The same protocols to move, copy and stream content are typically used with sometimes small extensions to improve user convenience. For example UPnP AV [31] may be used to move content around in a home network. DRM metadata extensions indicate to the user and the receiving system that it is DRM protected content.

Insertion of new content happens frequently, for example when music labels release new music albums. Insertion of new content involves content protection by the content packager component. The content key is distributed to the license server, and the availability of new content is signaled to the catalog and Website to make it available for sale.

12.3.2 Rights/License Management

In a classic DRM system a license defines the rights that are issued for the content. A license is a signed statement by a content provider that indicates under what conditions it is allowed to use the content encryption key and access a piece of content. A typical license has a structure like:

License = { ContentID, ContentKey, Subject, RightsExpression,
SignatureContentProvider }.

The content identifier forms the link between the license and the content to find the right license for some content and vice versa. For this purpose it is necessary that the content can be uniquely identified. If the content identifier is also used for other purposes such as rights clearing then often standards are used like DOI (Digital Object Identifier) [13] and MPEG-21 DII [15].

The rights expression indicates how the subject may use the content. Typical examples for rights expressions include unlimited play (a user "owns" the content, see Figure 12.3), play for one month (a user has a subscription), play three times, copy once and write to CD. The rights expression format can be ranging from copy control bits to XML rights expression languages (REL) such as ODRL [12], XrML [4] and MPEG21-REL [17]. Most full-fledged DRM systems nowadays use a REL, although still systems exist that use implicit rights such as FairPlay in which the rights are defined by the system. All main RELs are based on a model that relates assets (content), subjects or principles, permissions, constraints on permissions, and conditions, although the exact terminology differs per standard.

Security of licenses relates to three main aspects, namely integrity of the rights expression, confidentiality of the content key and integrity of the state for so-called stateful licenses, e.g., ensuring that a play-three-times license is not played four times. Integrity of the rights expression is typically addressed by a signature of the content provider. Confidentiality of the content key is realized by protecting the content key using some other key, e.g., encrypting the license with the public key of the target device or by a domain key. The management for these keys is system specific and therefore we will give one example in Section 12.4. Integrity of license state is the responsibility of the DRM client, which will typically maintain the state in some secure license storage.

12.3.3 User, Device and Domain Management

Granting access to content based on licenses is key to DRM. Access may be granted because a certain device is used. Access could also be granted because a certain person has authenticated and requests access. Alternatively, content access is granted if a device is used that belongs to a certain domain, i.e., a group of devices. Figure 12.2 conveniently summarizes these three cases with the term subject. A license server binds a license to a subject as part of license acquisition after the content is bought.

The above principle requires identification and authentication of devices and users. For this purpose devices get an identity and are certified. Certification has the further advantage that it allows to make distinction between trustworthy compliant devices that follow the rules of the DRM system and devices that do not. Only compliant devices may have access to DRM secrets and keys.

```
<o-ex:rights>
 <o-ex:context>
  <o-dd:version>2.0</o-dd:version><o-dd:uid>RightsObjectID</o-dd:uid>
 </o-ex:context>
 <o-ex:agreement>
  <o-ex:asset>
   <o-ex:context><o-dd:uid>ContentID</o-dd:uid></o-ex:context>
   <o-ex:digest>
    <ds:DigestMethod ds:Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
    <ds:DigestValue>DCFHash</ds:DigestValue>
   </o-ex:digest>
   <ds:KeyInfo>
    <xenc:EncryptedKey>
    <xenc:EncryptionMethod
     xenc:Algorithm="http://www.w3.org/2001/04/xmlenc#kw-aes128"/>
     <xenc:CipherData>
      <xenc:CipherValue>EncryptedCEK</xenc:CipherValue>
     </xenc:CipherData>
    </xenc:EncryptedKey>
    <ds:RetrievalMethod ds:URI="REKReference"/>
   </ds:KeyInfo>
  </o-ex:asset>
  <o-ex:permission><o-dd:play/></o-ex:permission>
 </o-ex:agreement>
</o-ex:rights>
```

Fig. 12.3. OMA DRM 2.0 license (simplified) for unlimited play right using ODRL.

12.4 Case: Content Management in the Personal Entertainment Domain

The previous section presented the general DRM architecture. We continue with a more detailed discussion for a specific case. We have selected the Personal Entertainment Domain (PED) concept. We sketch the PED-DRM concept and a realization. PED-DRM builds upon two hot topics in DRM, namely domain-based and person-based DRM.

The objective of this case is to give an impression what aspects and considerations play a role if we want to access content on a number of devices and based on user presence. The solutions and mechanisms presented highlight certain aspects rather than that they present a blueprint of a DRM system. The architecture deviates on some aspects from existing approaches such as OMA DRM or WM DRM. This follows mainly from other assumptions and requirements. For this case we assume that DRM functions should be performed on devices instead of servers where possible, and that devices should be able to operate while they are not online.

Content management in PED-DRM for commercial audio/video content is a special case of data management, because of its distribution model and required security. The distribution model of commercial audio/video content

is typically characterized by a download from a server to some device over a public channel, followed by small scale distribution in a local network. Securitywise, content in this case requires usage control next to access control.

12.4.1 Personal Entertainment Domain concept

Digital Rights Management (DRM) with support for domains [11, 32] needs to fulfill the requirements of both the content owners and the users, which often appear to be conflicting. The general idea is that content can flow freely between the devices that belong to the domain, while content transactions between domains are restricted.

Companies [10, 26, 29] and standardization bodies such as DVB (Digital Video Broadcasting) [32] and OMA (Open Mobile Alliance) are investigating and developing the concept of domains [9, 18]. Traditionally people have taken a device-oriented approach [11], where a domain groups a set of devices that belong to a certain household.

Many of the device-based domain concepts suffer from technological or user convenience problems, e.g., with respect to enabling the user to access content anywhere, at any time and on any device. The PED-DRM concept [20] does not have many of the disadvantages of device-based domains. More important, it starts with a comprehensive concept that users can understand, i.e., a limited number of rules and no differentiation between device classes. This is different compared with current DRM systems like FairPlay and WM DRM that do make this distinction, e.g., between PCs and portables.

PED-DRM is characterized by its structure, i.e., the relationship between various entities such as content, devices and persons, and by its policy, i.e., the rules that govern content access and proliferation. Key characteristics of the PED-DRM structure are that one single person is the member/owner of the domain, that content is bound to that person and that a number of devices is bound to the user (see Figure 12.4). Key characteristics of the PED-DRM policy are that content can be accessed on the domain devices and on all other compliant devices after user authentication. Content access on the set of permanent domain devices without user authentication allows for convenient content usage at home, including the sharing of content among family members. The only thing people must do is to register their device to their domain once. Temporary content access on all other compliant devices after user authentication enables people to access their content anywhere and at any time. Devices may be a member of multiple domains, both permanent and temporary.

Two small scenarios form the foundation of the PED-DRM concept as they illustrate the expected user experience and interaction, namely use of family content at home, and personal content use at another remote place. We assume that a user has a user identity device, such as a smartcard or mobile phone, with which he can authenticate conveniently to other devices. Access to family content at home is typically done on a central device in the living room such as a media center or PVR connected to the TV. A user can operate

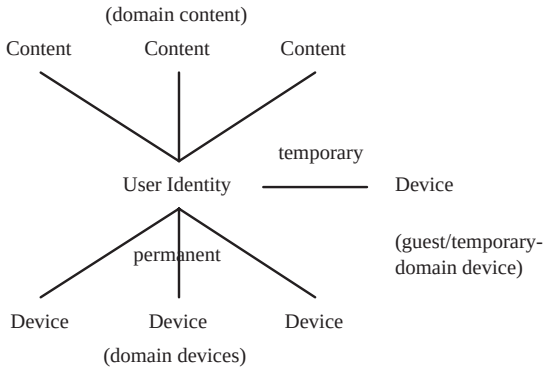


Fig. 12.4. PED-DRM concept.

his media center and select a movie bought by another family member using the remote control. The movie starts rendering after he presses play. Access to personal content at a remote location, conveniently called guest access, is typically done in a hotel or at a friend's house. The user decides he wants to render some content stored on his media center at home and he authenticates to the hotel TV using for example his mobile phone. The TV lists his available content and after the user selects some content the TV renders the content, which is streamed from his home over the Internet.

12.4.2 Functional PED-DRM Architecture and Design

Figure 12.5 shows a functional and data view of PED-DRM. The typical domain aspects of PED-DRM build upon the user, device and domain management functions (Figure 12.5, right). Domain management concerns the management of the set of permanent devices in the domain. It has a loose coupling to the rest to limit the effect on the traditional DRM functions (Figure 12.5, left). The relation between rights management and domain management is typically realized by means of a user identifier embedded in the license. This relation illustrates that a user owns a piece of content.

User and Device Management

User and device management in PED-DRM is not different than normal DRM. Users get provisioned with a certificate and corresponding public/private key pair.

Devices in PED-DRM are given a DeviceID certificate and key pair that they can use to prove their compliance. Devices are also given explicit authorization to fulfill certain functions. This limits the effects of a security breach by preventing the certificate and keys of a hacked device from being misused for other functions, e.g., keys from a rendering device cannot be used to register devices to the domain.

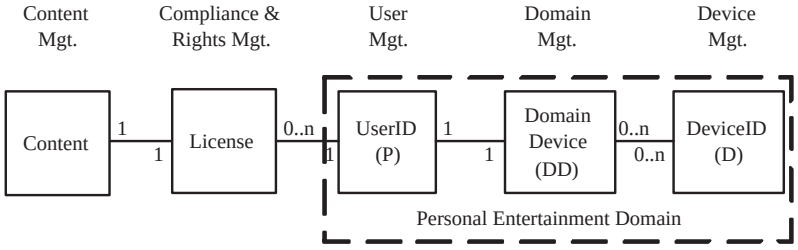


Fig. 12.5. PED-DRM functions and data overview.

Domain Management

Domain management in PED-DRM concerns the relation between users and a number of devices, as depicted in Figure 12.5 where a UserID and a number of DeviceIDs are brought together by a DomainDevices (DD) data object. Here, we present an approach in which DD is a certificate containing a reference to the user of the domain, references to a number of devices, a version number and the signature of the domain manager (DomainManager in Figure 12.6):

$$\text{DD} = \{\ \text{DomainID, Version, UserID, DeviceID1, } \ldots\text{, DeviceIDn,} \\ \text{SignDomainManager }\}.$$

The first advantage of making DD a certificate is that it shows who issued it. The second advantage of putting all domain members in one certificate is that this allows a simple but secure signaling mechanism to show which devices are in the domain. The third advantage of the DD certificate is the ability to report domain information to the user on any domain device at any time. Alternatively, a device gets a domain membership certificate that only lists itself. This is an option, but it lacks amongst others the latter two advantages.

To make optimal use of the DD certificates devices should exchange each other's DD certificate as part of common DRM operations such as license exchange of licenses belonging to the domain. When a domain device receives a valid DD certificate with a higher version number than its stored DD certificate, it replaces the stored DD with the new DD, provided that it is still contained in the new DD certificate, otherwise it removes its DD completely.

It is typically a security requirement that in case of a hacked device that only content is compromised that was available to the device, i.e., the domain content. To address this requirement domain-based DRM systems often base their security on domain key(s), e.g., SmartRight [29], xCP [26], PERM [10] and OMA DRM 2.0 [22]. In these systems the content key is typically encrypted with the domain key. We address this requirement differently by limiting license distribution to permanent and temporary domain devices.

System Components and their Interaction

Figure 12.6 presents the main client side DRM components – DomainManager, DRMClient and UserIdentity – that group PED-DRM functionality, and the interaction between them. These components interact with Terminal for interfacing with the end-user, and License Server to acquire licenses for content.

The typical connectivity means that enable interaction between the components are also indicated: combined on the same device (local), connected through a network (IP) or via wired/wireless connection with a limitation on the distance (near-field).

The DomainManager, DRMClient and UserIdentity must run on a compliant device which has a DeviceID certificate because they manage domain or content-related sensitive data. In a typical deployment of components over devices UserIdentity and DomainManager components are combined on one device, e.g., on a smartcard or mobile phone. Alternatively, DomainManager runs as a service on the Internet, an approach similar to OMA DRM 2.0 and Apple's FairPlay. Ideally, the DRMClient and Terminal are combined on one device, allowing straightforward domain management operations using the user interface of the device for interaction with the user. Typical devices include media centers and connected renderers (TVs).

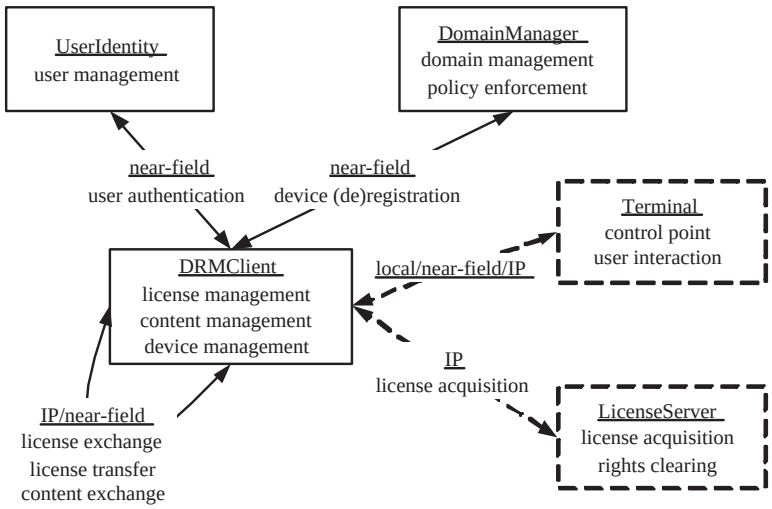


Fig. 12.6. PED-DRM client side components and their interaction.

Domain Policy and Domain Management

The domain policy specifies under which conditions entities are entitled to be part of the domain and thereby largely defines the scale of content proliferation in a domain-based DRM system. It is evident that end-users prefer a policy with a relaxed regime, while copyright holders prefer more tight regimes. As in most domain-based DRM systems, PED-DRM has a domain policy that is fixed for the system. OMA DRM 2.0 takes a slightly different approach by making solely the license issuer responsible for domain policy enforcement. The drawback of the latter is that a user has to redefine his domain for each shop he wants to buy content from.

We propose a simple and straightforward basic domain policy enforced by the DomainManager. The policy is based on a maximum number of devices per domain. Furthermore, a DomainManager only registers DRMClients that are in direct proximity. This limits the domain size and content proliferation to places that the user visits. Devices may be a member of multiple domains to support sharing of content between people who share devices.

Technically, enforcement of the domain policy by DomainManager is the main part of domain management together with the creation and management of DD certificates. Other aspects are secure domain registration and deregistration protocols. In a successful run of the registration protocol the device is authenticated as a compliant device, the request is evaluated against the domain policy, and the device gets an updated DD certificate with its own identity listed.

Content and License Management

The working of a DRM system is largely defined by the protocols and processes for content and license management. This section discusses the PED-DRM protocols for the leading example presented in Figure 12.7.

The leading example starts with some content bought by the user (U1). Figure 12.7 shows that the license server stores the encrypted content (contentB) and the related content key (keyB). The first action is the acquisition of a license for this content by a device containing a DRMClient (A). After that the DRMClients belonging to the domain (D1) exchange the content, and the receiving DRMClient (B) renders the content. Subsequently the content is exchanged with another DRMClient (C) which renders it. The exchange and rendering are both based on authentication of the user (U1). Finally, the user transfers the content ownership to another user (U2). This user has its own domain (D2) that includes his device with DRMClient (C).

The figure also shows that all devices have compliance certificates (certX) and related public/private key pairs (pubKeyX/privKeyX). The DRMClients are member of a domain for which they store a DD certificate. As defined before, DD consists of the domain ID, the version number of the certificate, the domain user, the domain devices and a signature by the domain manager.

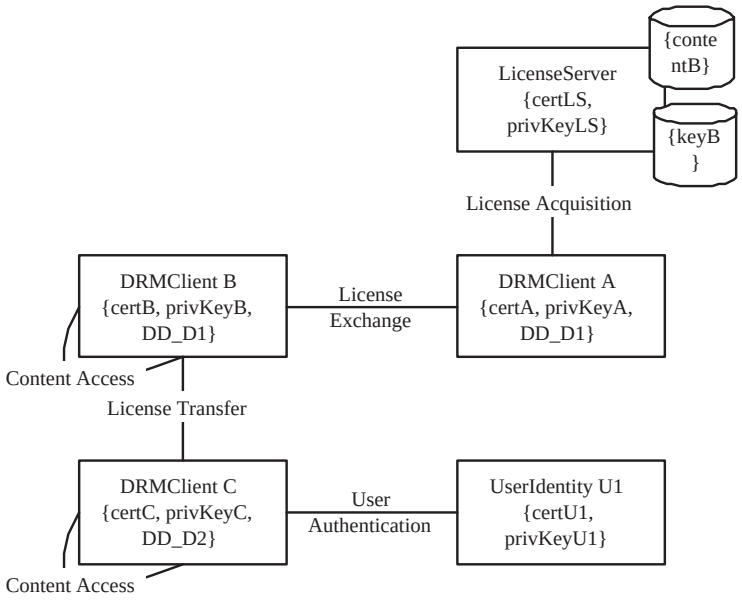


Fig. 12.7. Leading example content and license management.

The protocols follow the general principle that content keys are only distributed to devices that can access the content. This affects the license acquisition and license exchange protocols, because the content key is part of the license. For example, devices distribute the license encrypted by the public key of the target device so that only the target device can decrypt it. Furthermore, devices keep the licenses in their secure storage database. Devices are responsible for sufficiently protecting their secure storage database.

Content and License Acquisition

Content acquisition in a DRM context involves buying content, acquiring a license and downloading the content. We assume that the user already bought and paid for the content. Figure 12.8 depicts the starting point and subsequent steps for license acquisition for our example.

The essential part of license acquisition is the binding of the license to the user identity and his domain. Therefore, the license acquisition request contains the license ID and the DD certificate.

The license server must be assured that it delivers the license and content key to a compliant trustworthy DRMClient (A). Therefore, the license server

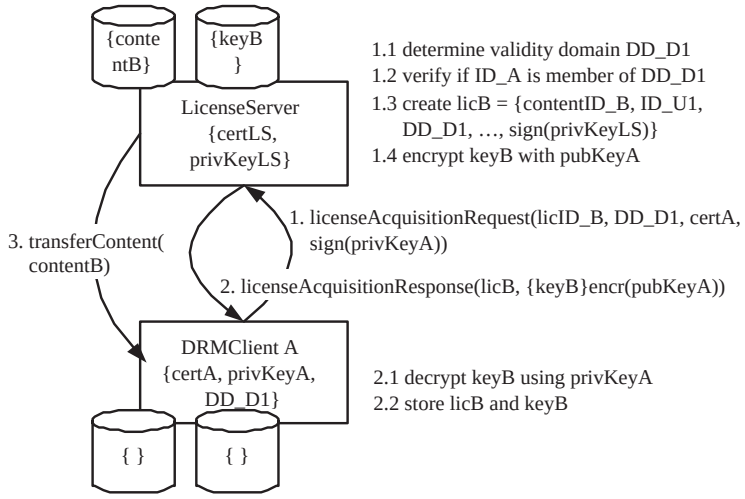


Fig. 12.8. Content and license acquisition.

uses the device certificate (certA) to verify that it is compliant. Of course, the license server also verifies that the signature of the request is from the device.

The license server furthermore requires that the domain (D1) has an acceptable policy. The license server verifies the signature on the DD certificate (DD_D1) to determine if it can trust the domain it issues content to. In systems with a fixed domain policy like here any domain created by a trustworthy DomainManager is accepted by a license server. The trustworthiness of the DomainManager follows from its certificate. Alternatively, it is possible to have different domain policies. In that case the domain policy should be indicated in the DD certificate. That enables the license server to determine if it delivers the license, withholds the license, or charges more. Finally, the license server checks if the requesting device is part of the domain. For this purpose it verifies that the device ID from the certificate (ID_A) is listed in the provided DD certificate (DD_D1).

After the checks the license server continues with the creation of a license (licB). This license binds the content (contentB) to the user (U1) and domain (D1), and also identifies the relevant content key (keyB). The license server encrypts the content key with the public key of the requesting device (pubKeyA) before it responds to the request.

The DRMClient of the device (A) verifies that it receives the correct license (licB) for the requested content (contentB), that it is bound to the right domain (D1), and issued by the license issuer to which the request was sent. This helps to detect accidental and malicious errors, and prevents license acquisitions from rogue license servers.

The transfer of the content (contentB) completes the license and content acquisition. Figure 12.8 depicts a simplified case where the content is served by the same server as the license. As explained earlier there is no security related to this transmission from the perspective of the content owner and license issuer. The device can verify that it received the correct content using the metadata in the content container and verifying the integrity of the content using information in the license. As a final step the device stores the license, key and content. This leads to the situation depicted in Figure 12.9.

Content and License Exchange
Content and license exchange concerns the organization of content and licenses over devices. This is especially relevant in domain based DRM systems because content may be rendered on any domain device. Content and license exchange only changes the location of content and licenses, but not the ownership. We talk about transfer of ownership later.

The exchange of a user's license between two of his domain devices is depicted in Figure 12.9. The responding DRMClient (A) has the encrypted content (contentB) and the corresponding license (licB) and content key (keyB). These are exchanged with the requesting DRMClient (B), which afterwards holds a copy as depicted in Figure 12.10.

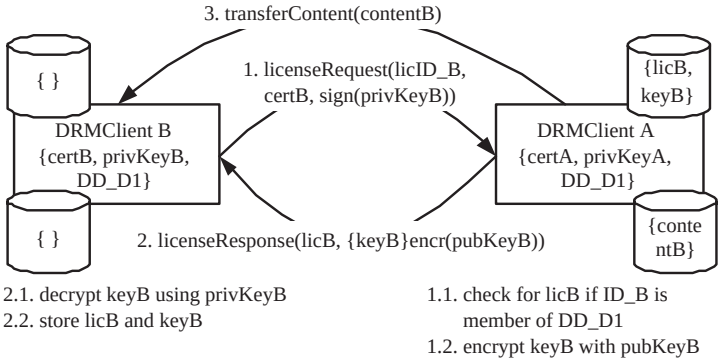


Fig. 12.9. Content exchange between domain devices.

The license exchange protocol starts with a license request. The request indicates the desired license and contains the necessary proof to convince the responder (A) to send the license. The proof consists of the requesters certificate and signature. The responding DRMClient (A) uses the certificate and signature in the request to determine that the requesting DRMClient (B) is compliant. Furthermore, it must be assured that both DRMClients belong to the same domain. For this purpose the responding DRMClient (A) retrieves the license (licB) from its license store, compares the domain indicated in the

license (DD_D1) with the DD certificate (DD_D1) it possesses, and verifies that the request or's ID (ID_B) is also listed in the DD certificate.

After the responding DRMClient (A) has done all checks it encrypts the content key (keyB) with the request or's public key (pubKeyB), and sends the response including the license (licB) and encrypted content key. The requesting DRMClient (B) decrypts the content key and stores both the license and the content key after verifying that it received the requested license.

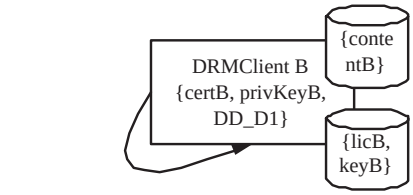
The protocol above only shows the basic steps. In a more elaborate version the DRMClients exchange their DD certificate to ensure that both possess the latest version. This ensures that both have the same view on the domain, which may have changed since recent domain changes. Another improvement is to sign the response to convince the requesting DRMClient that the license originates from the intended responder. Also, a challenge/response should be included to prevent that an old response is replayed. Finally, revocation should be taken into account, i.e., the responding DRMClient checks that the requesting DRMClient is not listed on a certificate revocation list.

The above protocol assumes that the receiving device is member of the user's domain to which the license is bound. Alternatively, the user could have authenticated to the receiving device, which is depicted in Figure 12.11 and also discussed in more detail in the next section.

Content Access

Content access is the general term for usage of content such as rendering and printing. This involves DRM processing like license evaluation and content decryption. It takes the secure content container and the license as input. Furthermore, context is an important parameter, e.g., date/time for subscription based licenses, but also current authentication sessions.

The process for content access only involves the device itself as depicted in Figure 12.10. In our example the DRMClient (B) has knowledge of the license (licB), its device identity (certB), domain information for the domain (DD_D1), and has access to the content key (keyB). The DRMClient verifies that it may use the license based on the user ID (ID_U1) listed in the license. This user ID must match the user ID in the DD certificate (DD_D1). In addition, the domain ID (ID_D1) in the license and DD certificate must also match. The DRMClient verifies that its identifier (ID_B) is listed in the corresponding DD certificate (DD_D1). One may question why this verification is still necessary, since licenses can only be distributed to devices after they have proven to be member of the domain as described for the license exchange protocol. However, time may have passed after the distribution of licenses. In the meantime the device could be deregistered from the domain. Therefore, it is verified upon content access that a device is still member of the domain. To complete the evaluation also the other conditions stated in the rights expression of the license are verified. After that the DRMClient releases the proper content decryption keys to enable decrypting and rendering of the content.



1. accessRequest(contentID_B)
1.1. evaluate licB license expression
1.2. check if ID_B is member of DD_D1
1.3. use keyB to decrypt contentB

Fig. 12.10. Content access on a domain device.

An alternative course of action for license exchange and content access is based on user authentication instead of domain membership of the DRMClient. The main difference is that the necessary proof now comes from the UserIdentity device and not from the DD certificate. Figure 12.11 depicts the relevant protocols for our example.

The first protocol is the authentication protocol (steps 1–3). To ensure presence of the user, a proximity verification is performed between the user's authentication token and the device. An unilateral challenge/response authentication convinces the DRMClient (C) that UserIdentity (U1) is present. Furthermore, the authentication response can serve as proof to other components that the DRMClient authenticated the user's token. This solution is just a basic version of the protocol and some extensions can improve security. An example is mutual authentication where UserIdentity also authenticates DRMClient and includes the identity in the response. Another security improvement is the inclusion of the validity time of the authentication in the proof.

The license exchange protocol works slightly different when based on user authentication (see Figure 12.11, step 4). The requesting DRMClient (C) includes the proof from the UserIdentity (U1). The responding DRMClient (B) uses this instead of the DD certificate. The responding DRMClient (B) verifies that the authentication proof it gets contains a reference to the same user ID (ID_U1) as the license (licB). The proof may be any signed statement, for example the authentication response message. If the proof contains a validity period then the DRMClient must verify that it is not expired to prevent that the authentication happened too far in the past. The DRMClient (B) also performs the other standard checks and responds with the license, key and content. The requesting DRMClient (C) stores these for the subsequent content access.

Content access follows the standard steps described before and depicted in Figure 12.10, except that the proof is used from the UserIdentity token (Figure 12.11, step 6). Next to the standard checks, it suffices for the DRMClient

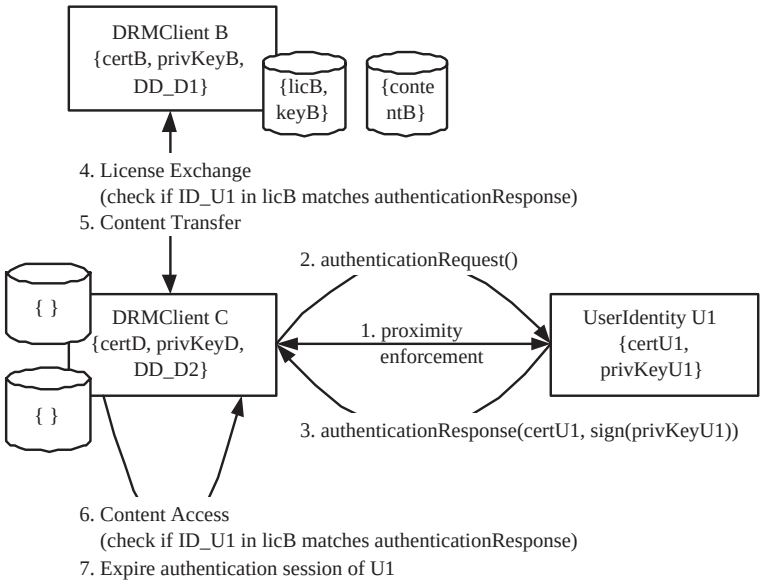


Fig. 12.11. License exchange and content access after user authentication.

(C) to check that the authenticated user ID (ID_U1) matches with the license (licB).

Finally, the authentication session on the DRMClient (C) expires (step 7). After this no licenses can be obtained from other devices for the domain (D1) or the user (U1), and the content cannot be rendered anymore. Securitywise it is no problem to keep the license and content on the DRMClient for future use.

As a closing note we should mention that the figure shows a basic version of identities and certificates stored on UserIdentity. In a flexible solution the token has separate certificates for compliance and user identity. This allows for example that users later obtain a token by buying a token and register it with their identity. These organizational and infrastructural aspects of user authentication and identity management have been largely omitted here.

Content Transfer

Content ownership transfer enables giving away or trading of multimedia content. The typical means to realize content transfer in a DRM system is to bind the license to a new user, which we call license transfer. License transfer is technically more challenging than license exchange, because it is not enough to distribute the license to another device. Instead, a new license must be created or the old license must be amended. Both approaches have to deal with trust issues. We take the approach to let the current owner (or one of his devices)

create a transfer license, which has the advantage that it works offline. This transfer license declares that the owner transfers a license to another user. This transfer license can only be used in conjunction with the original license. This approach maintains the integrity of the old license. Thereby trust issues stay limited to the transfer itself performed by the device. This avoids attacks where standalone devices create new licenses. In such an attack not only the owner could be changed but also the conditions in the rights expression. A possible extension is to exchange the transfer license together with the original license at the license server for a new license. After that the transfer license can be discarded.

The license transfer protocol for our example is depicted in Figure 12.12. The interaction is very similar to license acquisition. The major differences are the creation of the transfer license and the revocation of the license on the domain devices of the old owner. The protocol starts when the requesting DRMClient (C) requests the license (licID_B) to be transferred and provides the new domain (DD_D2) including the user information (ID_U2). The responding DRMClient (B) verifies that the requesting DRMClient (C) is compliant, and that the domain (DD_D2) is genuine. It also verifies that it is entitled to transfer the license, for which we adhere to the rule that it must belong to the domain of the license. After that it creates the transfer license (transferLic), which rebinds the license from the owner (U1) to the new owner (U2). Furthermore, it should initiate revocation of the licenses (licB) in the old domain (D1), because those should not be used anymore after the ownership has been transferred. After that, it sends the license, the transfer license, and the content key (keyB) to the requesting DRMClient (C). As for the other protocols the content key is encrypted with the public key of the requesting DRMClient (C). The requesting DRMClient (C) receives and stores the licenses and key. As an additional check the requesting DRMClient (C) verifies if the other DRMClient is entitled to transfer it. Here, this check consists of checking if the responding DRMClient belongs to the domain at the moment of transfer. This is verified using the DD certificate (DD_D1). To complete the transfer also the content is sent to the requesting DRMClient.

Rendering of the content is now possible on the requesting DRMClient (C). However, the evaluation process now also requires the evaluation and interpretation of the transfer license. For example, a rendering DRMClient must check that the transfer license is issued by a compliant device. Exchange of this content and license with other domain devices in the domain (D2) is possible using the license exchange protocol. This license exchange should also include the transfer license. For efficiency reasons it is best if the license and transfer license are kept closely together from this point onwards since they cannot be used apart.

License revocation is essential for license transfer because content may only be rendered by the new owner and not by the former. For license revocation no good general solution has been found yet. Also current commercial DRM

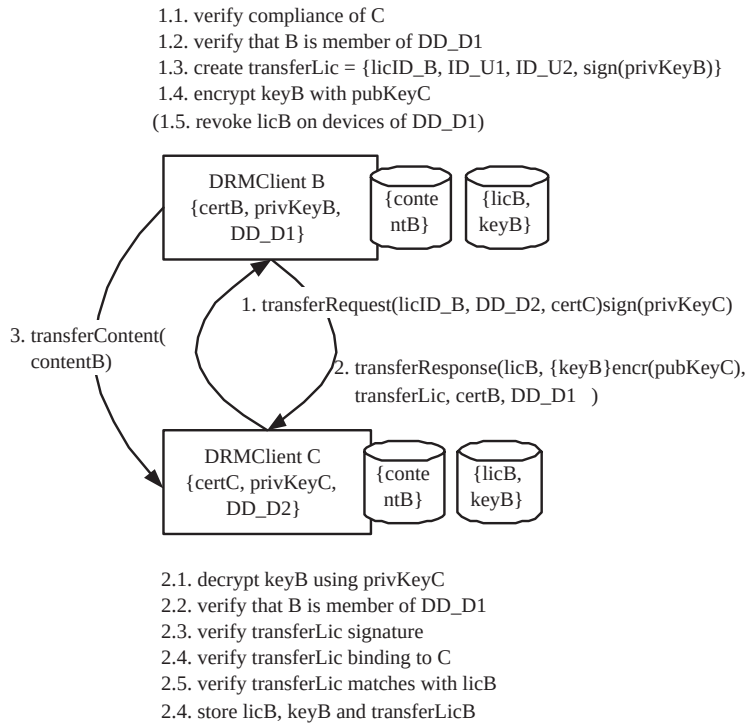


Fig. 12.12. Content transfer between users and their domains.

systems do not support license revocation. Therefore, license revocation is a topic that still requires research.

12.5 Summary

In this chapter the foundations of DRM have been introduced. DRM is based on copyright law. However, DRM goes further since it also covers content usage. The scope of DRM ranges from protecting audio/video entertainment content to enterprise rights management to protect business data, e.g., in the area of healthcare.

A general principle found in the DRM architecture is the separation between content and licenses, where the former can be characterized as data and the latter as essential metadata. The DRM architecture assumes trustworthy compliant devices to enforce the security of licenses and content. This provides the foundation for preventive DRM in which only allowed actions on content are possible.

A case on person-based and domain-based DRM showed that users need a clear DRM concept. The PED-DRM concept enables fair scenarios such as accessing family content at home and accessing personal content anywhere. Key management is also important in DRM and has been illustrated in a number of protocols for license and content management.

12.6 Further Reading

More information on DRM architecture, technology, research and news can be found in the following sources. Several books [1, 30] provide an introduction to DRM and various architectural and technological aspects. To get an insight in the structure and engineering side of DRM one could read the OMA DRM version 2 architecture overview and DRM specifications [22]. Scientific conferences and workshops in the field of DRM are the annual ACM DRM workshop which exists since 2001, the annual IEEE workshop on DRM Impact on Consumer Communications since 2005, the annual IFIP Conference on Communications and Multimedia Security (CMS), and the conference on Digital Rights Management: Technology, Issues, Challenges and Systems (DRMtics) which had its first edition in 2005. For DRM news, technological developments and accessible overview information one can access online Internet sources like DRMWatch [8], DRM News Blog [7] and the Wikipedia's article on DRM [34].

DRM is a topic that has gone through a long history already and is still being researched and standardized. A number of current research topics are introduced below together with some references to existing work. A first topic is research in the field of person and identity-based DRM, e.g., the Personal Entertainment Domain concept [20], and in the OPERA project [33]. Closely related to this is the further work on domain research, e.g., secure content exchange in OMA DRM [24], domain management, and license state management in domains. The introduction of person identities in DRM also raises privacy and user control issues in the area of DRM [2, 3, 28]. DRM interoperability [19] gets higher on the agenda now actual DRM systems are getting introduced in the market, giving momentum to initiatives like Coral [5]. Furthermore, DRM systems start to allow import from and export to other content protection systems. For example in OMA work is ongoing to unify secure flash storage with DRM [25]. A final topic that requires further research is the transfer of ownership of content, e.g., to give it away or to trade, which raises issues like license revocation.

References

1. E. Becker, W. Buhse, D. Günnewig, and N. Rump (Eds.). *Digital Rights Management: Technological, Economic, Legal and Political Aspects,* LNCS2770, Springer-Verlag, 2003.

2. C. Conrado, F.L.A.J. Kamperman, G.J. Schrijen, and W. Jonker. Privacy in an Identity-based DRM System, *Fourteenth International Workshop on Database and Expert Systems Applications,* 1–5 September 2003, Prague, pp. 389–395, 2003.
3. C. Conrado, M. Petkovic, and W. Jonker. Privacy-Preserving DRM, *Secure Data Management (SDM) 2004,* Toronto, Canada, LNCS 3178, Springer, 2004.
4. ContentGuard. *eXtensible rights Markup Language (XrML),* 2003. `http://www.xrml.org/`.
5. Coral Consortium Corporation. *Coral Consortium Architecture Specification: version 1.0,* March 2005. `http://www.coral-interop.org/`
6. Digital Video Broadcasting (DVB). *Support for use of scrambling and Conditional Access (CA) within digital broadcast systems,* ETR 289, ETSI, 1996.
7. DRM News Blog. `http://www.drmblog.com`, 2007.
8. DRMWatch. `http://www.drmwatch.com`, 2007.
9. A.M. Eskicioglu and E.J. Delp. An overview of multimedia content protection in consumer electronic devices, *Signal Processing: Image Communication,* Elsevier, 2001.
10. J. Gildred, A. Andreasyan, R. Osawa, and T. Stahl. *Protected Entertainment Rights Management (PERM): Specification Draft v0.54,* Pioneer Research Center USA Inc, Thomson, 9-2-2003.
11. S.A.F.A.van den Heuvel, W.Jonker, F.L.A.J.Kamperman, and P.J.Lenoir. Secure Content Management in Authorised Domains, *IBC Conference Publication,* pp. 467–474, Amsterdam, IBC2002, 15-9-2002.
12. R. Iannella. *Open Digital Rights Language (ODRL) Version 1.1,* W3C, 2003.
13. International DOI Foundation. *The DOI Handbook,* 2005.
14. Internet Streaming Media Alliance. *ISMA Encryption and Authentication Specification 1.0,* Feb. 2004.
15. MPEG-21. *Information technology – Multimedia framework – Part 3: Digital Item Identification,* ISO/IEC 21000-3, 2003.
16. MPEG-21. *Information technology – Multimedia framework – Part 4: Intellectual Property Management and Protection Components,* ISO/IEC 21000-4, 2006.
17. MPEG-21. *Information Technology – Multimedia Framework – Part 5: Rights Expression Language,* ISO/IEC 21000-5, 2004.
18. W. Jonker and J.P. Linnartz. Digital Rights Management in Consumer Electronics Products, *IEEE Signal Processing Magazine, Special Issue on Digital Rights Management,* 2004.
19. R.H. Koenen, J. Lacy, M. Mackay, and S. Mitchell. The Long March to Interoperable Digital Rights Management, *Proceedings of the IEEE,* vol. 92, issue 6, pp. 883–97, June 2004.
20. R.P. Koster, F.L.A.J. Kamperman, P.J. Lenoir, K.H.J. Vrielink, Identity-based DRM: Personal Entertainment Domain, *Transactions on Data Hiding and Multimedia Security I, LNCS 4300,* Springer-Verlag, Berlin Heidelberg, pp. 104–122, 2006.
21. Marlin Developer Community. *Marlin Architecture Overview,* 2006.
22. Open Mobile Alliance. *DRM Architecture: Approved Version 2.0,* 3-3-2006.
23. Open Mobile Alliance. *DRM Content Format V2.0,* April 2005.
24. Open Mobile Alliance. *Secure Content Exchange (SRE),* Work Item Document, 2005.

25. Open Mobile Alliance. *Secure Removable Media Profile (SRMProfile),* Work Item Document, 2005.
26. F. Pestoni, J.B. Lotspiech, and S. Nusser. xCP:peer-to-peer content protection, *IEEE Signal Processing Magazine, vol. 21, issue 2, pp. 71–81, March* 2004.
27. M. Petkovic, M. Hammouténe, C. Conrado, and W. Jonker. Securing Electronic Health Records using Digital Rights Management, *10th International Symposium for Health Information Management Research (iSHIMIR),* Greece 2005.
28. M. Petkovic and R.P. Koster. User Attributed Rights in DRM, *First International Conference on Digital Rights Management: Technology, Issues, Challenges and Systems,* 31 October–2 November 2005, Sydney, Australia, LNCS 3919, pp. 75–89, 2006.
29. Thomson Multimedia. *SmartRight: Technical white paper, version 1.7,* January 2003.
30. B. Rosenblatt, B. Trippe, and S. Mooney. *Digital Rights Management: Business and Technology,* M&T Books, 2002.
31. UPnP Forum. *UPnP AV Architecture,* June 12 2002.
32. R.Vevers and C.Hibbert. Copy Protection and Content Management in the DVB, *IBC Conference Publication,* pp. 458–466, Amsterdam, IBC2002, 15-9-2002.
33. S. Wegner (ed.). *OPERA – Interoperability of Digital Rights Management (DRM) Technologies,* OPERA Eurescom Project 1207, August 2003.
34. Wikipedia. *Digital rights management,* `http://en.wikipedia.org/wiki/Digital_rights_management`, 2007.