DISCOVERING THE CORE SECURITY REQUIREMENTS OF DRM SYSTEMS BY MEANS OF OBJECTIVE TREES

Hugo Jonker[1,2] and Sjouke Mauw[1]

[1] *University of Luxembourg, Faculty of Science, Technology and Communication*
6, rue Richard Coudenhoven-Kalergi, L-1359 Luxembourg, Luxembourg

[2] *Eindhoven University of Technology, Department of Mathematics and Computer Science*
P.O. Box 513, NL-5600 MB Eindhoven, The Netherlands
E-mail: hugo.jonker@uni.lu, sjouke.mauw@uni.lu

Keywords Digital Rights Management, Security Requirements, Objective Trees.

DISCOVERING THE CORE SECURITY REQUIREMENTS OF DRM SYSTEMS BY MEANS OF OBJECTIVE TREES

ABSTRACT

The use of Digital Rights Management (DRM) systems involves several stakeholders, such as the content provider, the license provider and the user, each having their own incentives to use the system. Proper use of the system implies that these incentives can only be met if certain security requirements are fulfilled. Quite some attention in literature has been devoted to specific security aspects of DRM systems. The contributions of this chapter consist of deriving a systematic overview of core security requirements for DRM systems. This chapter conducts a stakeholder analysis, gives an objective tree for each relevant stakeholder, and develops a simple, generic conceptual model to arrive at the set of core security requirements.

1. INTRODUCTION

There is a precarious balance between dissemination of information (to the general public) and stimulation of innovation and art. The easier it is to spread new information, the less possibilities to profit there will be for innovators to reap the fruits of their labour. On the other hand, spreading innovation and art is considered beneficial to society.

The introduction of computers has had a profound impact on this balance. With computers, it is trivial to create a perfect copy of content – a term used to indicate a work of art, such as music, literature, movies, etc. This coupled with the widespread availability of broadband internet connections means that completely new venues for spreading content to the public at large have come into existence. This enables a business model that consists of selling and delivering digital versions of content online. The main point of concern for such a business is to prevent unsanctioned redistribution of the delivered content.

Digital Rights Management (DRM) systems have been created for this goal. The purpose of a DRM system is to protect (digital versions of) content. Content is bound to a license, and the content is only accessible under the terms stated by the license. Since the year 2000, there has been a strong push into the research and development of DRM systems. There has been work on various related security aspects such as secure storage (Shapiro & Vingralek, 2002), traitor-tracing (Kiayias & Yung, 2003; Safavi-Naini & Wang, 2003), watermarking (Cox, Bloom & Miller, 2001), fingerprinting (Haitsma & Kalker, 2002; Prechelt & Typke, 2001) and tamper resistant code (Horne, Matheson, Sheehan & Tarjan, 2002; Chang & Atallah, 2002). There have also been various proposals for models of DRM systems with specific properties (OMA, 2004; Serrão, Naves, Barker, Balestri & Kudumakis, 2003; Guth, 2003; Popescu,

Kamperman, Crispa & Tanenbaum, 2004).

These proposals incorporate various security requirements. Some of these requirements assure core DRM functionality, whereas other requirements realise the specific properties for which that architecture was constructed (e.g. interoperability: MOSES (Serrão, Naves, Barker, Balestri & Kudumakis, 2003), Coral (Coral Consortium, 2006)). The emphasis of such proposals is usually on the latter type of requirements. It is not uncommon that the requirements assuring core DRM functionality receive a lesser treatment. These requirements are often not all made explicit, nor is a justification for them provided. Which of these requirements are made explicit varies from proposal to proposal, which means that the set of requirements that assure core DRM security is scattered. There are several reasons to make this core explicit. The first and foremost reason is that security is an enabling factor for DRM systems. DRM systems are designed to provide a solution for a security problem. An understanding of (the justification for) the core security requirements is crucial for fundamental comprehension of the security of DRM systems. Moreover, knowledge of the core security requirements is instrumental in the construction and verification of DRM systems. Such knowledge enables developers to better understand the consequences of security trade offs. In practical systems, such trade offs between desired features and security requirements are not uncommon.

For example, Apple's iTunes allows the user to create a CD of protected music. Naturally, Apple realised that such a CD could be used to copy music. Nevertheless, this feature was deemed more important than the costs in terms of loss of security. In this case, an informed decision has been made. In other respects, some of the design decisions of iTunes seem less well-informed and have a negative impact on the overall security of the system. A more detailed examination of iTunes follows in Section 4.3.

This chapter uses a structured approach to identify core security requirements and provide a justification for them. The goal of this chapter is focused on the security aspects of the design of DRM systems and our aim is to systematically derive core security requirements for DRM systems. Although there is a wealth of methodologies supporting system analysis and design, the methodologies for deriving security requirements are only at their infancy. Therefore, our research will start by identifying some useful methodologies and combining their strengths.
In order to provide a base for the found security requirements, we describe a model which limits itself to the core processes of a DRM system. The generic nature of this core model of DRM functionality implies that requirements found for it are applicable to most DRM systems. The extensibility of this core model indicates that it can be augmented to accommodate additional functionality, and therefore that this model suffices for our needs.

The rest of this document is organised as follows: Section 2 details the approach we used to arrive at our security requirements, resulting in a list of security properties and a generic

process model of DRM systems. These form the basis of the security requirements in Section 3. Section 4 examines the practical application of this research. And finally, we present some conclusions in Section 5.

2. PROBLEM ANALYSIS

To establish the core security requirements of DRM systems, we performed a problem analysis. The analysis led to a terminology and a description of the desires of the various involved parties. These desires are used in Section 3 as the foundations for the security requirements of DRM systems.

The problem analysis consisted of three steps. The first step established core stakeholders, their incentives and relevant terminology of DRM systems. The second step consisted of deriving the desired security properties from the incentives and the terminology. The third step consisted of deriving a process model. The process model captures the core operations occurring in DRM systems.

2.1. Establishing terminology, core stakeholders and their incentives

The first step in deriving security requirements is a *stakeholder analysis*. The purpose of this step is to determine the individuals (or roles) that have an interest in using a DRM system, and their incentives for participating. This understanding of their incentives is important, as these incentives lead to security requirements.

To establish the stakeholders and their incentives, a method similar to and inspired by several existing methodologies from the field of Information Systems has been used. We based our research on a variety of methodologies, such as domain analysis (see e.g. Prieto-Díaz, 1990), stakeholder analysis (for an overview see Pouloudi, 1999), system decomposition (see e.g. Sommerville, 2004). Normally, these methods assist in designing a system. However, our goal was not to design a system, but to focus upon a system's security aspects. Parts of these methods were accordingly adapted to capture the security aspects of DRM systems.

The problem analysis leads to the following list of core stakeholders and their roles in a DRM system:

Stakeholder	Role
Media company	Creates content
User	Acquires content
Distributor	Intermediary between *user* and *media company*

Each of these stakeholders has their own, specific incentives for using DRM. DRM systems

deal with providing users access to content. This observation together with a generalisation of the stakeholders' roles leads to the following formulation of the core functionality of a DRM system:

> *A DRM system is a system that enables* users *to access* digital content *according to access conditions as specified in a* license. *Licenses and content are provided by* distributors, *who in turn acquire content and generic access conditions from* content creators.

The above description distinguishes three types of participating roles: the content creator, who creates content; the distributor, who distributes content; and the user, who desires access to content. Although developers (who create a DRM system) and network providers (who transport content) are stakeholders too, and provide essential functions for an operational DRM system, they perform no operations on the content itself. Hence, we will not study their security requirements in this chapter – network security and e-commerce security are thus beyond the scope of the current discussion.

The analysis indicates that the core roles of content creator and distributor are executed by companies (e.g. media companies). As companies are legal entities, they are firmly embedded in a legal framework and thus there exist numerous non-technical solutions to ensure that companies adhere to access agreements. This observation holds less for individual persons, which allows for an asymmetry in DRM systems: there are fewer deterrents to prevent individuals from circumventing agreements then there are deterrents preventing companies. Hence for individuals strict technological enforcement of access agreements between them and companies is necessary. This leads to a natural tendency in DRM development to focus on enforcing access control at the user's side, and to focus less (or not at all) on access control at the distributor's side.

Each of the core roles has various reasons for taking part in a DRM system, and thus related security concerns. The following outlines their incentives and security needs:

Content creator

The role of content creator is executed by stakeholders that create content, such as media companies. They do not seek interaction with users directly, but are satisfied with leaving this to an intermediary – i.e. distributors. DRM can be used to enforce the conditions set on content by the content creator for the distributor. However, this can also be dealt with by non-digital means (contracts, agreements between companies).

Content creators use DRM technology to support new business models. For instance, they can create a bundle of desired content and other content. Such a bundle could increase the value of the content for users (e.g. by including “the making of” footage), or it could increase revenue

for the content creator (e.g. by including commercials). Additionally, using DRM technology it is possible to offer a revenue-generating alternative for traditional downloading. This means that DRM technology can open a new market.

As digitised content facilitates widespread copying of content, content creators need to be assured that the DRM system that safeguards their content will only allow access to content to the right party, when conditions in the right license have been met. Hence, for content creators, the security property desired is:

c1. Content is only accessible by untampered components created by the official system developers, under the conditions of a valid license issued by a bona fide distributor.

Distributor

The distributor can use DRM technology to offer tailor-made access to content to users. On top of that, overhead compared to selling physical media is substantially reduced, because digital content requires up little physical space and the presentation of the content can also be done digitally. By offering a clearly legitimate and known-quality alternative for downloading, distributor can open up a new market. And lastly, as content is bound to a license created by the distributor, access to content distributed in this way will comply with access conditions set by the distributor.

The distributor too desires the DRM system to prevent unlicensed copying. More precisely, any content distributed by the distributor should only be accessible in accordance with the access conditions set by the distributor. Moreover, in order to compete with downloading from dubious sources, communication must be secure and reliable. And naturally, the system must secure the distributor against attacks. Hence the desired security properties for distributors are:

d1. Content is only accessible by a device with a valid license issued to that device, originating from the distributor, under the terms stated in that license.
d2. The DRM system precisely delivers the content that has been requested, with the license as requested, in the correct format, at the desired time to the user.
d3. Other parties cannot influence (hack/break) the distributor's side of the communications between the distributor and the user.

User

Users will be drawn to DRM systems because DRM systems offer a legitimate, known-quality alternative to more dubious sources of content. Ease of use is an important consideration in this

regard: if use of the system or acquisition of content becomes bothersome, a user might turn back to other sources. Another advantage that DRM systems can offer users is the possibility to restrict the access to (and thus the cost of) content to precisely what the user wishes. For users, it is important that the DRM system offers an improvement over existing content distribution channels, otherwise there is no incentive for users to switch to DRM. This can, for example, be in terms of ease of use or availability. Such improvements can be offset by deterioration of other aspects – privacy concerns could turn users away from a DRM system.

The ubiquitous availability of a DRM system is thus an important property for users, as are privacy considerations. Furthermore, as the development of the DRM system in general does not involve the end-user, it is important that a user can trust the system to act according to his wishes. Hence, users desire the following security properties:

u1. The user can precisely acquire a consumable form of the content that the user desires, at the moment the user desires it.
u2. Neither content nor licenses can be linked to the user.
u3. Conditions of and requests for content and/or licenses are fully controlled by the user.

Taken together, the above properties form a solid foundation for the core functionality of DRM systems – and therefore, they also provide sound underpinnings for the security properties of DRM systems.

2.2. Conceptual process model

The second step of the problem analysis consists of the development of a *conceptual process model*. This model relates the basic processes in the DRM system to each other, and can be refined to provide a basis for identifying security requirements. This process model is then combined in Section 3 with the list of security properties to establish core security requirements of DRM systems.

To ensure that the model is applicable to as many DRM systems as possible, it must be as generic as possible. In order to derive such a generic model, we start with one component for the three core roles in a DRM system: the content creator, the distributor and the user, see Figure 1. Components for non-core roles are left out as they can be introduced when necessary, as such additions constitute refinements of the core model. The resulting model is subsequently refined to incorporate various specifics of DRM systems.

The model of Figure 1 is read as follows: the content creation component creates content, which is forwarded to the distributor. The license binding component binds a license to the content,

after which content and license can be sent to the user. The content rendition component at the user side renders the content.

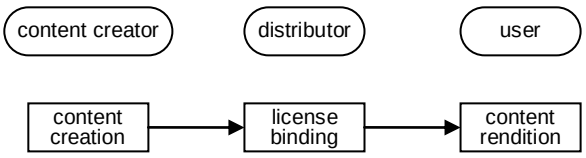

Fig. 1. Basic process model.

The first refinement to this model is instigated by the observation that to be perused by users, the content is eventually transformed into an analogue form. This means that there exist two variants of the content: digital content (which the system is to protect), and analogue content (which a user can consume). As all DRM systems need to convert content to a user-readable form, the generic model can be refined to incorporate this distinction without harming its genericity. Note that after conversion, more processing can be done. However, this is no longer on digital content and it is not a necessity in a DRM system. For both these reasons, we consider that a DRM system governs content until conversion to analogue.

The two types of content allow us to refine the content rendition process: one process to convert digital content to analogue content (*analogue conversion*), and one process that extracts the content from the package created by the *license binding* process from the distributor (the *content extraction* process). This refinement is depicted in Figure 2. Note that in general, analogue conversion is the final step in the process. Hence, the analogue conversion process is depicted as the last process in the model.

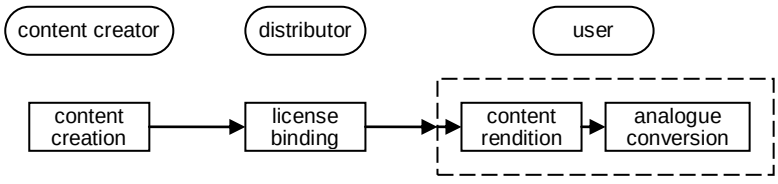

Fig. 2. Process model incorporating rendition refinement.

The next refinement is prompted by noting that DRM-protected content cannot be accessed without a license. This does not imply that a license must be bundled with the content in all cases, or even that a license must exist when the content is protected. It suffices that the resulting content is not accessible without a valid license. To express the possibility of the separate existence of licenses and protected content, the license binding component is further refined into two parts. The *content encapsulation* process provides the user with protected content, while the *license creation* process provides the user with a matching license for

protected content. This leads us to the conceptual model depicted in Figure 3.

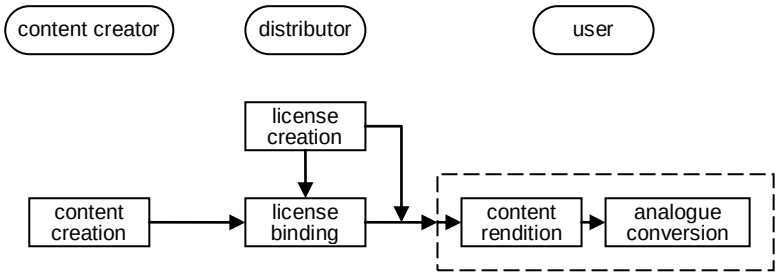

Fig. 3. Generic process model of DRM systems.

This generic process model now incorporates the core roles and models the core processes in a DRM system. The model can be refined to comply with virtually all existing DRM architectures. For example, additional roles like content issuer and license issuer can be incorporated to comply with the functional architecture of the Open Mobile Alliance DRM architecture (OMA, 2004).

The goal of this chapter is to establish core security requirements of DRM systems. As further refinements constrain the applicability of the model (i.e. they harm the genericity of the model), the model is not further refined.

2.4 Problem analysis results

The problem analysis has provided two results: a list of security properties desired by the core stakeholders, and a conceptual process model of the processes taking place in a DRM system. Taken together, these results elucidate security requirements of the core functionality of a DRM system.

Completeness of these descriptions can have a large impact on which security requirements are found. After all, the more complete the given descriptions are, the more complete the derived security requirements will be. The methods we adapted to arrive at the given descriptions support a systematic derivation of security requirements, but do not guarantee completeness of the results. Despite this, we believe that, due to the systematic approach, the descriptions are sufficiently exhaustive for our goals.

3. ESTABLISHING CORE SECURITY REQUIREMENTS USING OBJECTIVE TREES

In this section, the security properties as described in Section 2.1 are used as a basis for

establishing the security requirements per stakeholder upon the generic process model, as shown in Figure 3. In this section, these resulting properties are translated into security requirements upon the model. Both the process model and the incentives were deliberately kept widely applicable, so that the resulting security requirements will also hold in general for DRM systems. To describe security requirements on a specific setting or DRM system, the process model as well as the below requirements can be further refined. An example of such a refinement is the inclusion of the role of Certification Authority.

The refinement of security properties was based on a technique described in Schneier (2000): attack trees. We applied this technique to systematically analyse the defensive objectives of a system – in this case, the generic process model as shown in Figure 3. In this application, the trees no longer represent attacks, but objectives, this technique is best called Objective Trees. An Objective Tree is a refinement tree, refining properties into requirements in a specific setting – the generic process model. The results of this systematic analysis are presented below.

3.1 Content creator

Figure 4 shows the objective tree for the content creator. Objective c1 was established precisely in Section 2.1, and is abbreviated in the figure as “content inaccessible unless ok”.

Note that the content creator requires the system to keep the content within the system, just as the distributor does (requirement d1). The difference between their respective security properties is that the content creator additionally desires content to be secured until it is handed to the distributor. From then on, their respective security properties governing accessibility of the content coincide. Hence, in Figure 4, only requirements on the system up to the distributor need to be considered. Further requirements are part of the distributor’s objective tree and are detailed there.

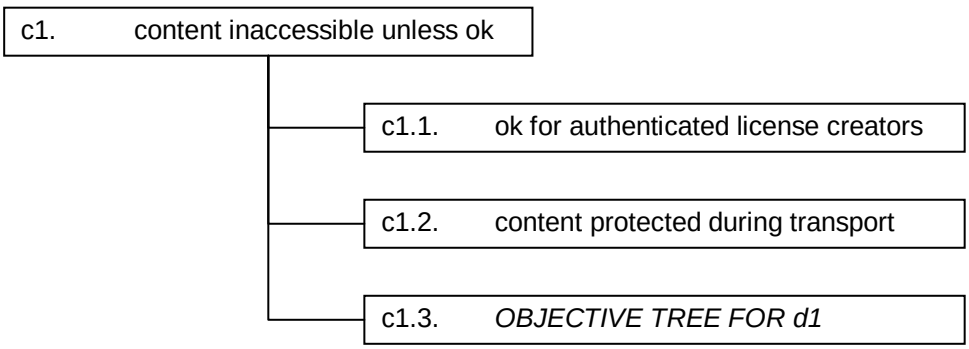

Fig. 4. Objective tree of security requirements for the content creator

3.2 Distributor

In Section 2.1, three desired security properties were listed for the distributor. As property d2 is the distributor's version of property u1, the objective tree for u2 serves as an objective tree for both properties. In Figure 5, the remaining properties are compactly described as "content inaccessible unless ok" (d1) and "safe and secure communications" (d3).

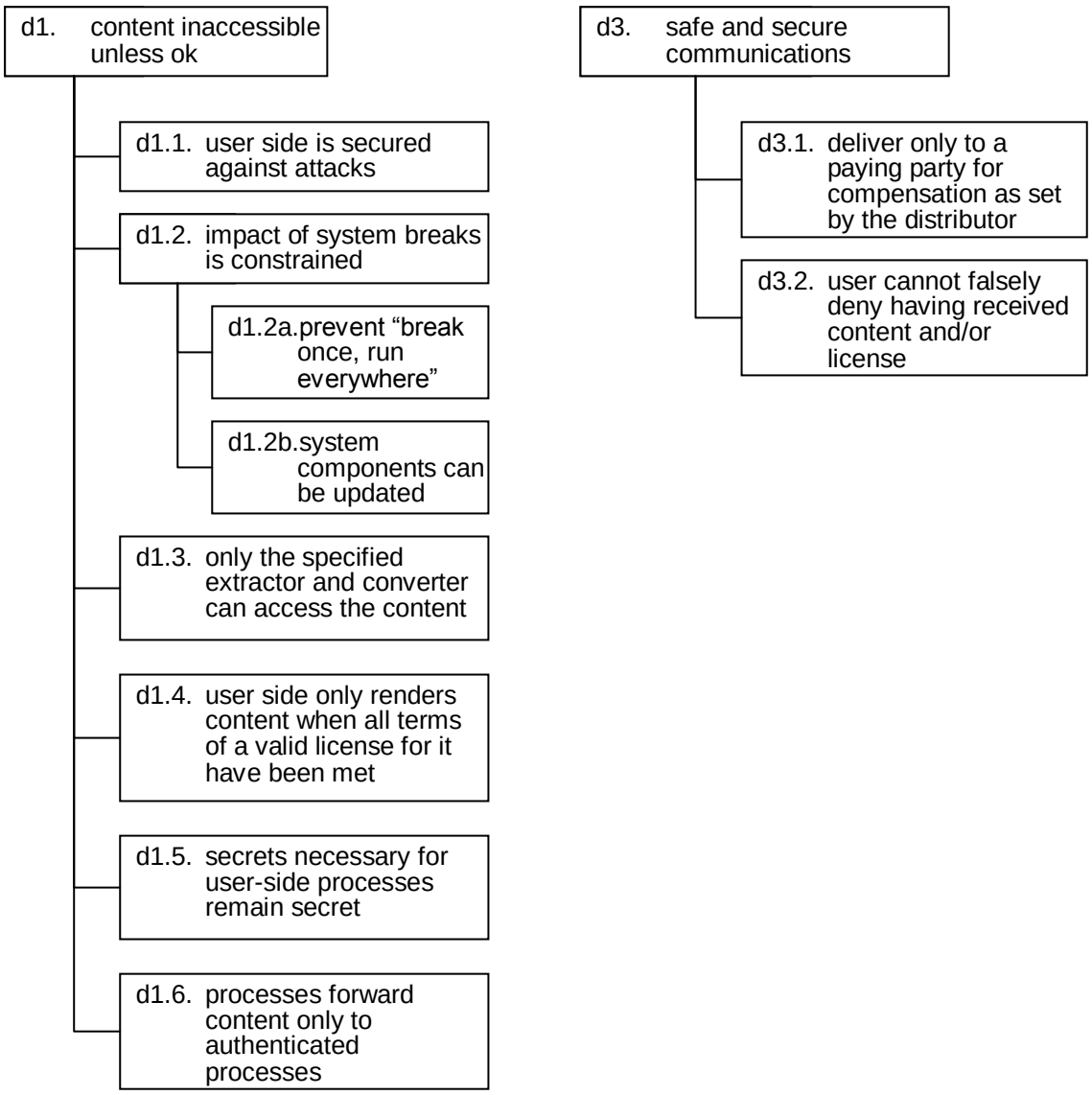

Fig. 5. Objective tree with security requirements for the distributor.
(a) for property d1, and (b) for property d3.

3.3 User

The security properties for the user, detailed in Section 2.1, are abbreviated in the objective trees of Figures 7-8 as follows: “delivery on demand” (u1), “privacy” (u2), “order control” (u3). Note that the objective tree of Figure 7 is the objective tree incorporating the requirements for properties u1 and d2.

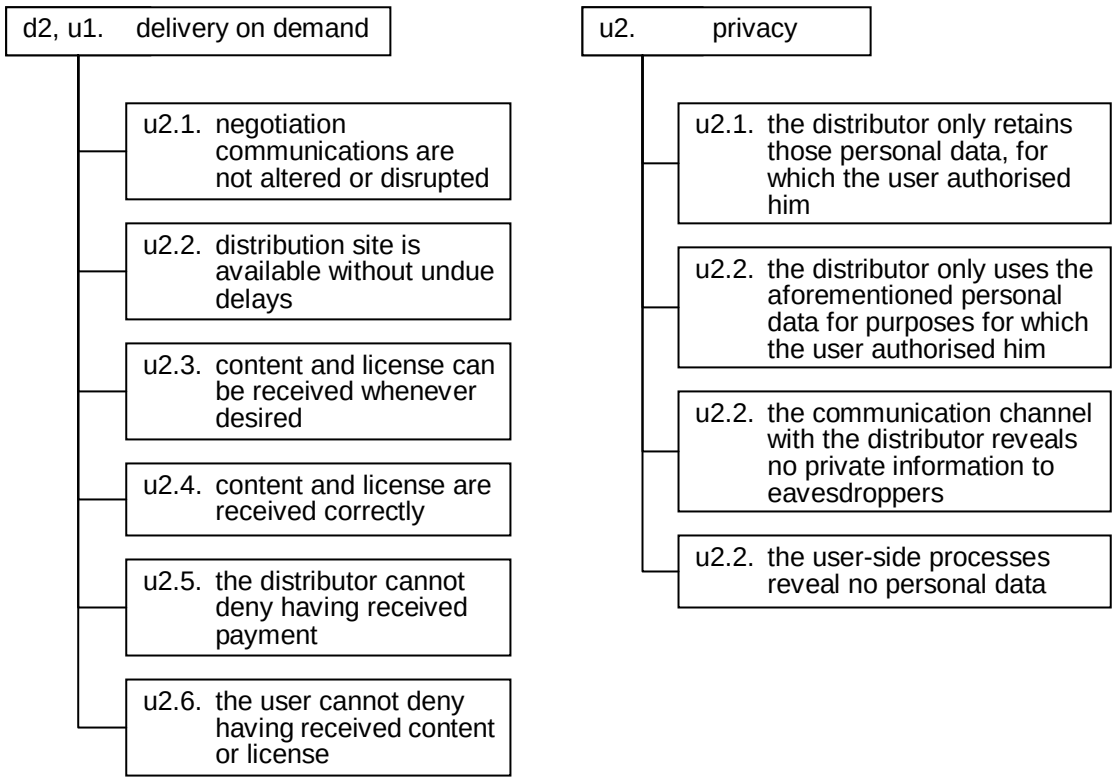

Fig. 7. Objective tree for properties (a) u1,d2, and (b) u2.

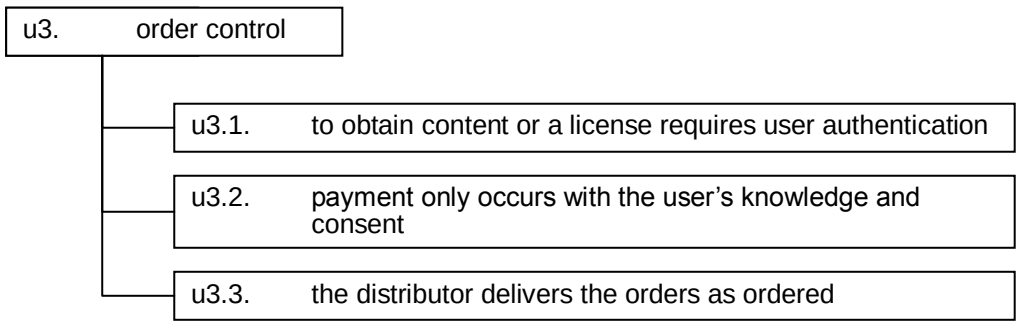

Fig. 8. Objective tree for property u3.

3.4 Consequences

The security requirements established in the previous section have some consequences, which are mentioned below.

It follows from the requirements derived from the distributor's desired security properties (d1-d3) that the components on the user's side should function as a trusted computing base. The requirements following from property u2 indicate that the distributor should provide a privacy statement. Requirement u3.3 implies that the distributor provides a security policy.

4. APPLICABILITY

In this section we will illustrate the use of the generic process model and derived security requirements to practical DRM systems. It is not our intention to provide a complete analysis of existing DRM systems, but to indicate the possibilities of applying our findings.

Our first case study was performed on a system from a Dutch company which develops DRM systems. This case study was based on our work, previously reported in Jonker (2004). The system matched our model very well, and our security analysis pointed out a missed requirement in the system. They took into consideration implementing a mechanism to ensure requirement u2.5 (non-repudiation for the distributor).

In the remainder of this section, we compare the results of this chapter (i.e. the generic process model and the security requirements upon it) to several high-level, generic DRM systems. This is followed by noting how to use the results in verifying DRM design. The section concludes by applying the found requirements to two popular DRM systems, *Windows Media DRM* from Microsoft and *iTunes* from Apple. These systems are interesting for a number of reasons: these systems are the most successfully deployed DRM systems in the current market, and both have been successfully attacked. Observe that both systems are refinements of the model depicted in Figure 3.

4.1 Comparison to generic DRM systems

Several DRM systems are being developed with a focus on genericity.
Examples include Coral (Coral Consortium, 2006), which focuses on interoperability; OMA DRM (OMA, 2004), which is mainly geared towards mobile devices; and Marlin (Marlin Developer Community, 2006), which is aimed at providing DRM for content accessed using varying services and devices in the bounds of a domain. All three of these have a broad target, and thus were developed as generic DRM systems. Because of this, they are perfectly suited to illustrate how to relate our generic results to real DRM systems.

Coral

The Coral architecture is developed to act as a basic building block for DRM systems. By itself, it does not constitute a DRM system, but is intended as a core framework that enables interoperability of DRM systems. This means that content is fed to it from one DRM system, then content is processed and finally, with an appropriate set of access conditions, content is forwarded to another DRM system. As such, interoperability of DRM systems seamlessly matches our generic process model.

As Coral focuses on providing interoperability, it has a vastly expanded set of roles. Most of these roles constitute specific refinements of the distributor's role, that enable interoperability of licenses between DRM systems.

As Coral acts not as a DRM system, but an interoperability layer, the role of the content creator is not strictly necessary. This functionality is taken over by the roles of content importer and exporter.

Open Mobile Alliance DRM

The OMA DRM system's primary focus is to enable DRM on mobile devices, such as cellphones. In this context, super distribution and accessibility of content for devices grouped into domains is important. Like Coral, OMA DRM is developed for implementation. The design documentation for OMA DRM provides an architecture, which acts as the counterpart to the generic process model, and a requirements document that incorporates a counterpart to the security requirements.

The functional architecture of OMA DRM closely resembles the generic process model. It consists of a content issuer (the content creator), a rights issuer (the distributor) and a DRM agent (the user). Nonetheless, the security requirements are all aimed at protecting content while interacting with the user. The focus of the security requirements of OMA DRM is upon requirements following from properties c1 and d1. Other requirements, especially the user's security requirements, are underrepresented in OMA's DRM. OMA DRM's specifications are at an implementation level, lacking a generic nature. This has caused their security requirements to focus on implementation aspects of security, and understate other requirements.

This observation does not imply that there OMA DRM suffers from a security risk – but the set of requirements used as input for designing the system omits several requirements established in Section 3.

Marlin

Marlin is a DRM architecture being jointly developed by several large consumer electronics companies (Sony, Samsung Philips, Panasonic) with additional input from a specialised DRM company (Intertrust). Given this strong CE manufacturer backing, it comes as no surprise that

Marlin is targeted at providing DRM for CE-devices, and ensuring interoperability of DRM across all devices belonging to one user. The Marlin developer community has created Octopus, a reference core impementation of a DRM system complying to the Marlin framework. Octopus, like OMA, focuses upon the user, and thus the distinction between content creator and distributor is not made explicit.

That notwithstanding, the three actors as specified in the core Marlin architecture conform to the three roles in the generic process model. Marlin's focus on CE devices and portability of content amongst devices owned by the same user has led to a refinement similar to one in OMA, namely the introduction of domains. This refinement introduces new security concerns, however, as Marlin recognises the same three core roles as the generic process model, these concerns are addressed by requirements following from the stated desired security properties. Unfortunately, Marlin security requirements are not publicly available, so a comparison between them and the stated security requirements in Section 3 cannot be made.

4.2 Verifying DRM design

The list of security requirements can be used as a checklist when verifying security aspects of a design for a DRM system. The generic nature of the process model implies that it is applicable to most designs for a DRM system. Therefore, the requirements derived from it should be met by any matching system.

Additionally, the process model can be refined to model a specific DRM system in more detail. This refined model could be combined with the stated requirements in order to derive more detailed requirements for this specific DRM system.

4.3 Application to popular DRM system

Both Microsoft and Apple have created DRM systems that enjoy a large use base. In this section, we take a closer look at these two systems using the established security requirements as a guideline.

Windows Media DRM

Microsoft has developed a DRM system compatible with their *Windows Media Player*. They created two file formats that can be enhanced with DRM protection: Windows Media Audio (.WMA, for audio content) and Windows Media Video (.WMV, for mixed video and audio content).

Microsoft's DRM system has a role delivering protected content to the user. This content can only be accessed with a legitimate license. This manner of operating conforms to the process

model depicted in Figure 3.

There have been two well-known successful attacks on Microsoft's DRM system (described more fully by Hauser & Wenz (2003)). The first attack consisted of replacing part of the playback device that is responsible for rendering audio (the audio driver). Protected music could now be captured after it had been converted to a form understandable by the audio driver, and saved to disc.

In our proposed security requirements, this attack would have been prevented by satisfying requirements d1.3 and d1.6. The part of the DRM system on which this attack worked, was the same for each installation. This means that the attack was possible on all installed systems, which violates requirement d1.2a.

Microsoft has released a new version of Windows DRM that was not vulnerable to this attack. Naturally, content packaged under the new system was not accessible with the old system.

A tool called 'FreeMe' is available that attacks content protected for the new version. The creator of the tool has included a detailed analysis of the protection of protected files. The tool is able to determine the keys used to encrypt the content. This is a violation of requirement d1.5. The manner in which the key could be retrieved was the same for each installed system, which again violates requirement d1.2a.

Microsoft has since patched their DRM system. The patch implements measures that support requirement d1.2a and uses another method to hide the used keys. Patched systems are no longer vulnerable to this particular attack.

iTunes

iTunes is Apple's online music store. Music is sold in the Advanced Audio Coding format (AAC, see ISO (1997, 2004)). The audio data inside the AAC file is protected by encryption. The default license allows protected files to be played on up to five different computers, and to burn them to CD.

The manner in which iTunes operates, conforms to the model of Figure 3. Content is delivered to a distributor, which binds the content to a license. The user can then acquire the bound content and an appropriate license.

iTunes has suffered from two well-known attacks. The first is by Jon Lech Johansen and captures the decrypted, digital contents before it is converted into analogue form. This attack is similar to the first attack upon Microsoft's Windows DRM system, and the remarks made then

apply in this case as well – this attack exploits the lack of compliance to requirements d1.3, d1.6 and d1.2a.

A second attack on iTunes bears a resemblance to the second attack on Windows DRM: keys can be learned by outsiders, which enables an attacker to compromise the encryption of protected content. The keys used by iTunes (to protect the key to the encrypted audio) are stored encrypted with a system key. It is known how to reconstruct this key[1] for the Windows platform and for Apple's portable mp3 player. This means that it is possible to remove the encryption from the audio data for these platforms. The remarks made for the second attack on MS DRM apply here as well – this attack exploits the lack of compliance to requirements d1.5 and d1.2a.

Johansen later found a third attack on the system. He created a player (called PyMusique) that replaces the iTunes player to acquire songs. According to the description of the method[2], the main difference with the official player is that it skips applying DRM to the downloaded file.

The attack thus consists of replacing a component of the DRM system by an attacker-controlled component. However, no component should send the content to another component, unless the other component identifies itself as a legitimate component. Allowing this constitutes a violation of requirement d1.6.

5. CONCLUSIONS

Digital Rights Management systems offer a method for content creators to allow their work to be spread digitally, without loss of recompensation. Security is the foremost enabling factor of DRM systems.

This chapter used a methodical way to derive security requirements from the incentives of the parties involved with a DRM system. A stakeholder analysis identified the key roles taking part in a DRM system, and the main incentives related to these roles. A domain analysis captured the core terminology of DRM systems, from which a process model was derived.

The key roles, their incentives and the core terminology together led to a description of each role's desired security properties for a DRM system. Taken together with the process model, these security properties allowed us to establish the core security requirements of DRM systems.

The method used was justified by validation of the result of each step to known literature and

[1] e.g., see `http://www.hymn-project.org/docs/hymn-manual.html#how-hymn-works`

[2] `http://www.daeken.com/2004/08/24/itunes/`

comparison to a DRM system in development by a Dutch company.

The applicability of our findings was justified by comparison to several existing DRM systems. Exploited weaknesses of both Microsoft Media DRM and Apple's iTunes could have been identified in an early state using the requirements we presented. In the case of the system in development at a Dutch company, we were able to point out requirements that had been overlooked.

This chapter takes a practical stance towards security: it presents a list of requirements to be fulfilled. Compliance with these requirements may not be straightforward – see for example the second attack on Microsoft's Media DRM, which attacked a part that at first might have seem to comply to the relevant requirements presented here. This problem can be prevented using formal verification. Compliance of the used protocols with secrecy and authentication can already be formally verified. There is ongoing research into formally expressing other requirements (e.g. privacy) for protocols. Further developments can lead to establishing a theoretical basis in which it is possible to verify that a trusted computing base complies to the requirements upon it.

The process model used for establishing the requirements can be refined to include more details for a more thorough analysis of a particular system. This would then lead to further refinement in the requirements. Such a refinement requires a loss of the generic nature and therefore would result in confining the applicability of the resulting findings to systems exhibiting specific characteristics.

ACKNOWLEDGEMENTS

The comments and commentary of ir. Lex Schoonen and ir. Jan Verschuren were invaluable during our research, and we are grateful for their constructive commentary.

REFERENCES

Chang, H. & Atallah, M.J. (2002). Protecting software code by guards. In *Security and Privacy in Digital Rights Management*, Springer LNCS 2320, 150-175.

Coral Consortium (whitepaper, 2006). *Coral consortium whitepaper*.

Cox, I., Bloom, J., & Miller, M. (2001). Digital watermarking: principles & practice. *The Morgan Kaufmann Series in Multimedia and Information Systems*. Morgan Kaufmann.

Guth, S. (2003). A sample DRM system. In Digital Rights Management, Springer LNCS 2770, 150-161.

Haitsma, J. & Kalker, T. (2002). A highly robust audio fingerprinting system. In *Proceedings of the 3rd International Conference on Music Information Retrieval*.

Hauser, T. & Wenz, C. (2003). DRM under attack: weaknesses in existing systems. In *Digital Rights Management: Technological, Economic, Legal and Political Aspects*, Springer LNCS 2770, 206-223.

Horne, B., Matheson, L., Sheehan, C., & Tarjan, R.E. (2002). Dynamic self-checking techniques for improved tamper resistance. In *Security and privacy in digital rights management*, Springer LNCS 2320, 141-159.

ISO, International Organization for Standardization (published standard, 2004). *ISO/IEC 13818-7:2004 Information technology – Generic coding of moving pictures and associated audio information – Part 7: Advanced Audio Coding (AAC)*.

ISO, the International Organization for Standardization (published standard, 1999). *ISO/IEC TR 13818-5:1997/Amd 1:1999 Advanced Audio Coding (AAC)*.

Jonker, H.L. (2004). *Security of Digital Rights Management systems*. Unpublished Master's thesis, Technische Universiteit Eindhoven.

Kiayias, A. & Yung, M. (2003). Breaking and repairing asymmetric public-key traitor tracing. In *Digital Rights Management*, Springer LNCS 2320, 32-50.

Marlin Developer Community (whitepaper, 2006). *Marlin architecture overview*.

OMA, the Open Mobile Alliance (2004). OMA-DRM-ARCH-V2 0-20040715-C DRM architecture.

Popescu, B.C., Kamperman, F.L.A.J., Crispo, B., & Tanenbaum, A. S. (2004). A DRM security architecture for home networks. In *DRM '04: Proceedings of the 4th ACM workshop on Digital Rights Management*, 1-10. ACM Press.

Pouloudi, A. (1999). Aspects of the stakeholder concept and their implications for information systems development. In R.H. Sprague (Ed.), *Proceedings of the 32nd Hawaii International Conference on System Sciences*, Los Alamitos, CA: IEEE Computer Society Press.

Prechelt, L. & Typke, R. (2001). An interface for melody input. *ACM Transactions on Computer-Human Interaction*, 8(2), 133-149.

Prieto-Digitalaz, R. (1990). Domain analysis: An introduction. *Software Engineering Notes*, 15(2), 47-54.

Safavi-Naini, R. and Wang, Y. (2003). Traitor Tracing for Shortened and Corrupted Fingerprints. In *Digital Rights Management*, Springer LNCS 2320, 32-50.

Schneier, B. (2000). *Secrets & Lies: Digital Security in a Networked World*. Wiley.

Serrão, C., Naves, D., Barker, T., Balestri, M., & Kudumakis, P. (2003). Open SDRM – an open and secure digital rights management solution.

Shapiro, W. & Vingralek, R. (2002). How to manage persistent state in DRM systems. In Security and Privacy in Digital Rights Management, Springer LNCS 2320, 176-191.

I. Sommerville (2004). *Software Engineering*. Pearson.

KEY TERMS AND DEFINITIONS

Content
A work of art, such as music, a movie, literature, software, et cetera.

Digital Rights Management
Describes techniques that manage protective measures for content.

License
A virtual object granting specific rights to a specific user for accessing content.

Security requirement
A specific prerequisite that a system needs to fulfil in order to achieve a specific security objective.

Stakeholder analysis
A methodology to determine which parties have an interest in a given situation.

Objective trees
A method to establish goals for stakeholders.

Open Mobile Alliance (OMA)
A standardisation body comprised of most companies in the cell phone market (manufacturers as well as network operators). The corresponding DRM specifications are called OMA DRM.