SECURE LICENSE MANAGEMENT
Management of Digital Object Licenses in a DRM Environment

Carlos Serrão, Miguel Dias
ISCTE/DCTI/ADETTI, Ed. ISCTE, Av. Das Forças Armadas, 1600-082 Lisboa, Portugal

Jaime Delgado
UPC/DAC/DMAG, Campus Nord Mòdul D6, E-08034 Barcelona, Spain

Keywords: License, DRM, Security, rights expression language, content encryption key.

Abstract: In the digital world protecting digital intellectual property is proving to be a hard task. Not only it is complex to provide robust and reliable mechanisms to prevent unauthorized content copying and utilization, but also it is complex to provide a mechanism for specifying and enforcing how content can and will be used. Rights expression languages allow content providers and distributors to syntactically and semantically to express a set of rights that are associated to a digital object. In this paper we will provide the definition and description of the digital object license granting rights life cycle management and processes necessary to secure the license throughout this entire life cycle.

1 INTRODUCTION

Digital Rights Management (DRM) involves the description, layering, analysis, valuation, trading and monitoring of the rights over an individual or organization's assets; in digital forma. Managing the way users/actors can interact with digital objects is one of the major functions of a digital rights management solution (Serrão, Kudumakis, et al, 2005). Modern DRM solutions allow the definition of a set of conditions and rights (REL), under which a specific actor can use a governed digital object. These REL can syntactically bound a digital object identifier, an actor identifier, a content encryption key and set of conditions, together. The goals and purpose of RELs can be characterized as the expression of copyright, expression of contract or license agreements and control over access and/or use.

Two of the used RELs on today's DRM panorama are ISO MPEG REL (a derivative from XrML and ODRL (Open Digital Rights Language) (Xin et al, 2005). Although RELs are a very powerful mechanism for rights expression they have little use if a system is not capable of interpret them and if the system is not capable of imposing the restrictions (if any) presented on the license (Safavi-Naini, 2006). Two main processes may occur while dealing with digital objects licenses: the creation of licenses and the usage and enforcing of the licenses.

The license creation process involves the enumeration and specification of the conditions that will have to be enforced over the digital object and the necessary digital object encryption keys necessary to access the digital object. The second process is the license usage and enforcement and is responsible for the compliance of digital object usage to the rights declared within the license, which can be enforced by the existence of content encryption or scrambling keys that might exist or not inside the license. It is therefore important for the license to be protected and managed properly. It is central to the digital object rights management that licenses, which express the way digital objects, can be used need to be managed throughout its entire life cycle.

Serrão C., Dias M. and Delgado J. (2007).
SECURE LICENSE MANAGEMENT - Management of Digital Object Licenses in a DRM Environment.
In *Proceedings of the Second International Conference on Security and Cryptography*, pages 251-256
DOI: 10.5220/0002129902510256

2 THE LICENSE MANAGEMENT LIFE CYCLE

In the curse of the digital objects rights management development, the existence of a mechanism to specify how that digital object could be used has always been important. Today's digital rights expression languages have a lot to thank to the early work developed by Xerox, giving origin to DPRL (Digital Property Rights Language) that was posterior evolved to XrML by ContentGuard (ContentGuard, 2001).

2.1 The Different Types of Rights Management

DRM systems depend on license creation and enforcement to express how digital objects can be used. This functionality is present in DRM systems in many different ways, through combinations between the existence of a formal rights expression and the content encryption keys (CEK). Several different combinations and scenarios result in the following typology identification: 1) Using REL, CEK inside REL, REL inside the digital object: in this case, a formal rights expression language is used to express the digital object rights, and the content encryption key is placed inside the REL as part of the license. The license is also placed inside the digital object and is part of it – the license is obtained at the same time than the digital object; 2) Using REL, CEK inside REL, REL outside the digital object: in this case, a formal rights expression language is used to express the digital object rights, and the content encryption key is placed inside the REL as part of the license. In this case the license is not part of the digital object and therefore it may be obtained at a different moment than the digital object; 3) Using REL, CEK outside REL, CEK inside the digital object: in this specific case a rights expression language is used and the content encryption key is not part of the license, however this key is part of the digital object; 4) Using REL, CEK outside REL, CEK outside the digital object: in this case a rights expression language is also used, however the content encryption key is not part of the license or the digital object. This means that both the license and the CEK need to be obtained in different moments apart from the digital object; 5) Not using REL, CEK inside the digital object: in this case a rights expression language is not used but the content encryption key is place inside the digital object; and 6) Not using REL, CEK outside the digital object: in this final case, no rights expression language is used and the content encryption key needs to be obtained to access the digital object. All the presented cases can be implemented in DRM solutions and scenarios although the most active and representative are those who use a rights expression language to represent licenses, that contain the content encryption key, like for instance the Windows Media DRM. There are different cases in which the license is contained or not inside the digital object himself. There are also some cases where a REL is not used to express the content access rights and everything is left to the rendering software or device – this is for instance the Apple iTunes FairPlay DRM (Serrão, Dias et al, 2006).

2.2 License Management Life Cycle

The creation and usage of rights expression language based licenses are involved in a cycle that goes from the digital object creation to the digital object usage. The enumeration and description of such cycle is quite important because it provides the mean to identify which are the basic procedures in the cycle and which are the crucial security mechanisms that need to be implemented to make the system robust (Figure 1). The process starts with the object capture and its encoding into a digital form, giving origin to a digital object. This process involves the choice of the appropriate encoding mechanisms taking into account where and how the digital object is going to be used (Serrão, Serra et al, 2006). The appropriate protection and packaging mechanisms are also selected and the content encryption keys are selected and applied during the protection of the digital object (Nutzel et al, 2006). Depending on the digital object protection strategy, a single encryption key can be used to protect entirely the object, or several keys may be used to cipher different parts of that same object. The digital object is also uniquely registered and assigned with a proper identifier that will be used to identify uniquely the digital object in the digital world. The keys used to cipher the content are registered on the system, and assigned to the unique digital object identifier generated on the previous step of the life cycle. These keys should be stored on a secure digital container and will be used on the future to allow the access to the digital object.

This concludes the digital object preparation and the different digital assets are made available to final users using different distribution channels. When distributing the digital object to final users the conditions for its distribution are set-up and an optional negotiation process with the end user may

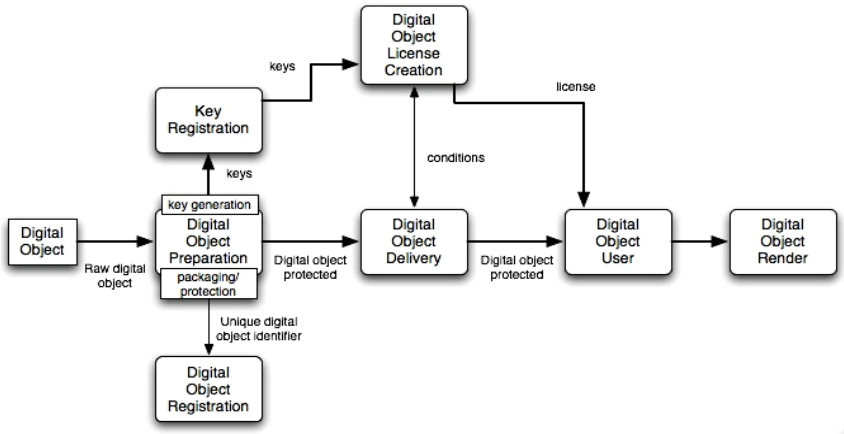

Figure 1: The digital object license life cycle. This life cycle starts with the creation of the digital object and the specification of the conditions under which the digital object can be used.

actually occur. Usage conditions can be expressed using rights expression languages that bound the digital object usage conditions, the content encryption key and the user or device identifier. In previous section we have identified the different types of relations between the rights expressions languages, the content encryption keys and the digital objects. In this case, we would like to simplify the process by stating that the licence contains the content encryption keys, the license is not part of the digital object and that it obtained in a different moment than the digital object itself. When the user receives the digital object and the license, this content encryption key and the inherent usage conditions need to be extracted from the license and used on the user digital object handling system to allow fair access to it, while upholding the conditions defined by the copyright holders and the specific conditions acquired by the users.

3 SECURITY IN THE LICENSE MANAGEMENT LIFE CYCLE

We have introduced in the previous section the description of the license management life cycle (Figure 1).

The process to generate keys is always a complex problem. The generation of keying material should follow a set of standards to have the necessary random data to be generated securely. Therefore the keys ($C_{EK[1]}$, $C_{EK[2]}$,… $C_{EK[n]}$) used to cipher the digital objects need to be generated, stored and handled securely. The generated keys are dependent of the type of object and of the protection that is going to be used on the digital object (Figure 1). These digital object protection keys need to be stored securely to be used in the future. This is necessary to avoid the compromise of these keys and to allow its posterior secure usage on the production of licenses. The key generation mechanism (Encryption Key Creation Service - EKCS) should possess a key-pair ($EKCS_{pubk}$, $EKCS_{prvk}$) and a digital certificate issued by a trusted entity (a certification authority) ($Cert^{CA}_{EKCS}$). During this stage the digital object is also registered on the Digital Object Registration Service (DORS). Also the DORS holds a key-pair ($DORS_{pubk}$, $DORS_{prvk}$) and a certificate issued by a trusted entity ($Cert^{CA}_{DORS}$). The DORS issues a Digital Object Unique Identifier (DOUI) that is digitally signed to prevent further alterations during the digital object lifetime: $DORS_{prvk}$[DOUI]. The format of the DOUI could follow one of the available standards (Figure 1).If the EKCS is not responsible for the long-term storage of digital object keys, then a Secure Key Storage Service (SKSS) needs to be used. This SKSS holds a key pair ($SKSS_{pubk}$, $SKSS_{prvk}$) and a certificate issued by a trusted entity ($Cert^{CA}_{SKSS}$). The EKCS after creating the keys sends them encrypted to the SKSS ciphering this information with SKSS public-key, contained in $Cert^{CA}_{SKSS}$. Additionally this information is combined with the DOUI: $SKSS_{pubk}\{ C_{EK[1]}, C_{EK[2]},\ldots C_{EK[n]}, DORS_{prvk}[DOUI]\}$. During this process the copyright owner (CO) also defines the license conditions for the digital object. The CO has also a key pair (CO_{pubk}, CO_{prvk}) and a digital certificate ($Cert^{CA}_{CO}$), and will digitally sign the conditions, expressed using a rights expression language under which the digital object can be used: CO_{prvk}[conditions]. These conditions will be securely stored by the Secure License Service (SLS), and will be indexed by the DOUI. Whenever an end-user (U) desires to obtain a protected digital object it

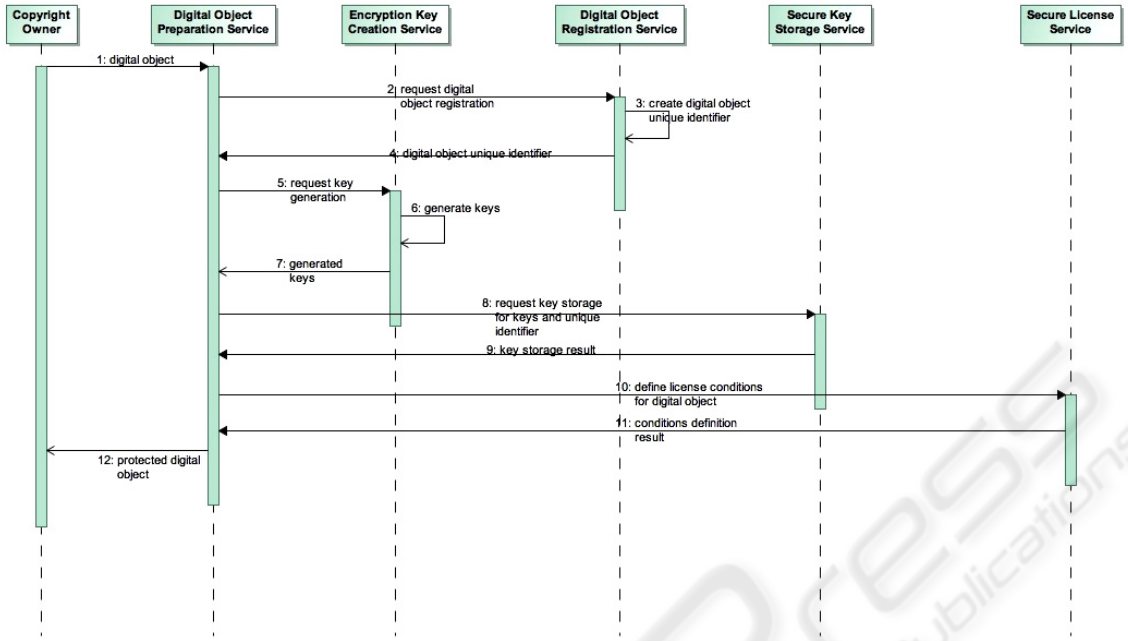

Figure 2: Definition of the digital object license conditions.

has to be authenticated by the rights management system that governs the digital object. To do this the user has a key pair (U_{pubk}, U_{prvk}) and a digital credential ($Cert^{CA}_{U}$). The U will use this credential to authenticate and start the digital object acquisition process. The U uses its credential to establish a secure session with the Digital Object Distributor (DOD) and the SLS, through the Digital Object Rendering Device (DORD). The DORD, in this scenario represents a general-purpose device (hardware or software based) capable of performing some functions over a digital object. During this process, the DOD loads the specific licensing conditions for the digital object that were previously established by the copyright owner. There are two possibilities in this step: either the copyright owner allows the negotiation of some conditions or the terms and conditions, or the license is closed and final. If the second situation occurs the end-user will have only the same type of usage defined for that specific governed digital object (Figure 2).

The U uses its credential to establish a secure session with the Digital Object Distributor (DOD) and the SLS, through the Digital Object Rendering Device (DORD). The DORD, in this scenario represents a general-purpose device (hardware or software based) capable of performing some functions over a digital object. During this process, the DOD loads the specific licensing conditions for the digital object that were previously established by the copyright owner. There are two possibilities in this step: either the copyright owner allows the negotiation of some conditions or the terms and conditions, or the license is closed and final. If the second situation occurs the end-user will have only the same type of usage defined for that specific governed digital object (Figure 3). After the definition of the specific conditions between the U and the DOD, the DOD instructs the SLS to produce a license for a given digital object (using the signed DOUI ($DORS_{prvk}$[DOUI])), the U identification ($Cert^{CA}_{U}$) and the conditions signed by the DOD (DOD_{prvk}[conditions])). The SLS receives this information and validates this by the verification of the digital signature of the U, the DORS and DOD.

This provides the warranty do the SLS that the request has not been tampered by some external and malicious entity. After these validations have been performed, the SLS verifies if the conditions present on the DOD request are valid and allowable by comparing them with the previous agreement with the CO. If they are, the license can be produced and the keys necessary to access the digital item can be provided to the U. With this information the SLS can obtain the CEK from the SKSS. The SLS authenticates to the SKSS using $Cert^{CA}_{SKSS}$, and uses the DOUI ($DORS_{prvk}$[DOUI]) to retrieve the appropriate keys to the digital object. These keys are ciphered with the SLS public key to avoid its compromise: $SLS_{pubk}\{ C_{EK[1]}, C_{EK[2]},\ldots C_{EK[n]}\}$.

These keys will be placed inside the license for further usage, and will be given to the entity trying to access the digital object. During the license

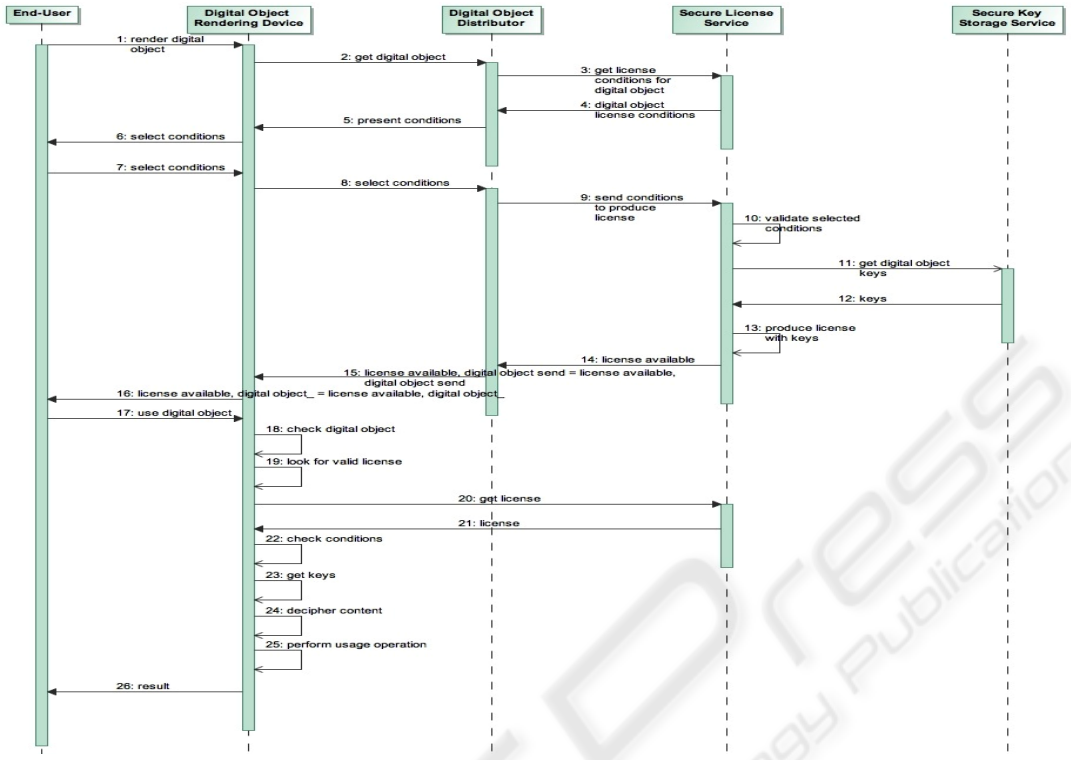

Figure 3: The digital object license acquisition process.

production phase it is extremely important that the key (or keys) contained are protected from peeking eyes. Therefore the SLS ciphers these keys with the receptor (user or device) public keys, avoiding therefore that some unauthorized entity may obtain the keys and use the digital object: $U_{pubk}\{ C_{EK[1]}, C_{EK[2]},\ldots C_{EK[n]}\}$. These keys are placed together with the end-user entity identity, the digital object unique identifier, the licensing conditions and a time validity: $U_{pubk}\{ C_{EK[1]}, C_{EK[2]},\ldots C_{EK[n]}\}$, $DORS_{prvk}[DOUI]$, conditions, timedate. This bundle is digitally signed by the SLS to avoid unauthorized modifications of it: $SLS_{prvk}[U_{pubk}\{C_{EK[1]}, C_{EK[2]},\ldots C_{EK[n]}\}, DORS_{prvk}[DOUI]$, conditions, timedate].

After the license (LIC) is produced a notification is returned to the DOD and optionally to the digital object-rendering device (DORD). Whenever the DORD tries to access the digital object, it verifies if it is protected or not. If it is, the DORD checks for the existence of a valid license on the system. If the license can be found, it is verified to check if the DORD is allowed to perform the requested action over the digital object or not. This validation process involves not only the verification of the license on the system, but also the verification of the license content, such as the acquired conditions and the time validity. The DORD should implement also some security mechanisms that will allow not only the secure storage of the licenses but also the mechanisms to handle the secure persistent storage of state information (such as render counters and others). This is still one of the major security breaches on software-only based DRM systems (Shapiro et al, 2002). If a license for the digital object is not yet available on the system, the DORD contacts the SLS (the information about the proper SLS to contact can be placed inside the digital object, or in a more interoperable manner DORD can contact any SLS without any concern if the requested license was or not produced by the same SLS), requesting a license for that digital object (using the DOUI) and the U identification: $DORS_{prvk}[DOUI]$, $Cert^{CA}_{U}$. The information contained on this request will be signed by the U: $U_{prvk}[DORS_{prvk}[DOUI], Cert^{CA}_{U}]$. This will avoid the modification of the request by some man in the middle attack. This information is received by the SLS, verified and checked against the information on the data storage, and if a matching license is found it is returned to the DORD. The license conforms to the format previously described and is signed by the SLS. When the license is on the DORD, it is securely stored and any state information (such as play counters will have to be

instantiated). While accessing the digital object the DORD securely retrieves the digital object protection keys from the license, and using them to render the decipher the content and allow the execution of the request operation over the digital object. The described process corresponds only to one of the many possible scenarios that were identified, in which the combination of rights expression languages, content encryption keys and digital objects may coexist (Zeng, 2006).

4 CONCLUSIONS AND FUTURE WORK

The authors of this paper have focused their work on the presentation of the way most digital object rights management solutions handle with the management of the digital representation of rights. We have also identified and briefly described a set of different scenarios, about the usage of digital rights expression languages for expressing digital object licenses, the presence of the content encryption key inside the licenses and the presence of such licenses inside the digital objects. This identification has resulted in six different scenarios, and the most relevant one (implemented in the most significant rights management solutions today) has been selected and the license management life cycle was described. After the identification and description of the major processes in the selected scenario of license management life cycle model, the authors have identified the basic security procedures that make the license management processes effective on the digital objects rights management. Crucial aspects such as confidentiality, integrity and authentication are of extreme importance and therefore need to be used with care to offer trust across the entire license management life cycle. This represents work in progress, and as a future work, the authors of this paper will extend the proposed license management life cycle model and analyse it in terms of the different scenarios identified and proposed in the paper. We will also try to identify from real existing rights management solutions how they handle license management, and how it can be mapped to an identified scenario and to the general life cycle model. The final goal for this would be to provide a generic license management framework that can be easily interoperable between the different rights governing solutions.

REFERENCES

Xin Wang, T. DeMartini, B. Wragg, M. Paramasivam, and C. Barlas. 2005. The MPEG-21 rights expression language and rights data dictionary. Multimedia, IEEE Transactions on 7, no. 3: 408- 417.

Nützel, Jürgen, and Anja Beyer. 2006. How to Increase the Security of Digital Rights Management Systems Without Affecting Consumer's Security. In : Emerging Trends in Information and Communication Security, 368-380.

Safavi-Naini, Reihaneh, and Moti Yung. 2006. Digital Rights Management: Technologies, Issues, Challenges and Systems. Springer.

Shapiro, William, and Radek Vingralek. 2002. How to Manage Persistent State in DRM Systems. In : Security and Privacy in Digital Rights Management : ACM CCS-8 Workshop DRM 2001, Philadelphia, PA, USA, November 5, 2001.

Zeng, Wenjun, Heather Yu, and Ching-Yung Lin. 2006. Multimedia Security Technologies for Digital Rights Management. Academic Press.

Serrão C., Torres V., Delgado J., Dias M., 2006, "Interoperability Mechanisms for registration and authentication on different open DRM platforms", in International Journal of Computer Science and Network Security, Vol. 6, Number 12, Pages 291-303.

Serrão C., Dias M., Delgado J., 2006, "Using Service-oriented Architectures towards Rights Management interoperability", in Proceedings of the International Joint Conferences on computer, Information and Systems Sciences and Engineering (CISSE06), University of Bridgeport, USA.

Serrão C., Dias M., Kudumakis P., 2005, "From OPIMA to MPEG IPMP-X - A standard's history across R&D projects", in Special Issue on European Projects in Visual Representation Systems and Services, Image Communications, Volume 20, Issue 9-10, Pages 972-994, Elsevier.

Serrão C., Serra A., Dias M., Delgado J., 2006, Protection of MP3 Music Files Using Digital Rights Management and Symmetric Ciphering, Proceedings of the 2nd International Conference on Automated Production of Cross Media Content for Multi-channel Distribution (AXMEDIS2006), Leeds, UK.