10

DRM & Security Enabling Mechanisms Leveraging User Centric Multimedia Convergence

Anastasios Fragopoulos, John Gialelis and Dimitrios Serpanos
Industrial Systems Institute (I.S.I.)
Greece

1. Introduction

There have been considerable efforts to have audiovisual multimedia systems and applications converge, in home environments with homes as spaces of convergence, and for nomadic users with advanced mobile devices as points of convergence. These trends are important but also have limitations that have to be addressed and overcome, i.e. home-centric systems fail to account for increased mobility and desire to provide continuous service across spatial boundaries outside the home; device-centric convergence, e.g. in 3G phones, supports nomadic use but provides a very limited user experience as no single device and interface will fit many different applications well; furthermore, in both cases there are a lot of different security aspects that arise and have to be taken in care. The trend in our era, is to move beyond home and device-centric convergence toward truly user-centric convergence of multimedia, where the user is acting as the point at which services (multimedia applications) and the means for interacting with them (devices and interfaces) converge.

One of the biggest challenges that we are facing in the deployment of architectures in such environments is related to, on the one hand, the security and protection of digital contents that interchanged between users in such pervasive ubiquitous computing environments and on the other hand with the provision to the users with security mechanisms that allow them to perform secure transactions (e.g. authentication, privacy protection, secure data transfer, etc.) in those environments. Moreover protection of Intellectual Property (IP) is a necessity in modern multimedia architectures. The concerns of content creators for loss of revenues constitute a strong obstacle to the wide deployment of architectures that involve distribution of IP protected digital content. Today the end-users are equipped with different types of small devices that allow them to be the digital contents creators, thus creating digital content that wish to share with third parties. In most cases, the end-users would like to have mechanisms which would give them the possibility to protect the content which have created and possibly to set their own usage rights over it, thus specifying towards third users how their digital content shall be used.

Digital Rights Management (DRM) mechanisms constitute of various technologies that have been developed and deployed by content providers, creators, distributors, in order to

protect their digital media from illegally, unauthorized and without the appropriate rights usage of their products, while let them to use possibly unsafe media like Internet for delivering their products with less hesitation and anxiety about non-legitimate usage of their content. Moreover, the increasing capabilities of embedded systems combined with their decreasing cost have enabled their adoption in a wide range of personalized entertainment services, applications and services, leading thus to a user-converged networked multimedia environments. Furthermore, as dynamicity in networks, embedded security and interoperability in DRM systems become the critical aspects in such networked ecosystems, new emerging frameworks for secure, user-converged digital content delivery are required, leading to specific treatment for the design and deployment of DRM systems that take in care the resource-demanding nature of security in embedded systems, (Fragopoulos & Serpanos, 2005), (Fragopoulos et al., 2009). Thus, it is a requirement to provide integrated DRM mechanisms in such services and applications that target delivery of IP protected content to a large base of clients. Considering the technical problems that result from add-on security solutions to independently developed network services, the design and deployment of architectures with security and DRM as inherent requirements will lead to secure solutions that will increase the trust placed by content providers on the system and thus, it will lead to wider availability of services to a larger population.

In this book chapter, we describe extensive research that has been done in the areas of DRM management and embedded security, towards the design and deployment of an integrated architecture that exposes security functionalities and DRM mechanisms, focusing on user-centric and nomadic environments. In that context we have taken specific care how we could adapt our architecture in order to cope with mobility management issues, while providing interoperability towards other similar architectures.

The architecture that we describe, (a) provides the capability to the user to act as content creator who set its own usage rules to his content, thus protecting digital content from unauthorized usage from non-legitimate users; (b) operates over heterogeneous network technologies; (c) provides to the end-user friendliness; (d) provides adequate security mechanisms, under possible attacks; and, (e) is adapted to mobility frameworks, providing secure access to DRM protected multimedia digital contents, (enabling DRM in session migration – high mobility environments).

Towards implementing licensing for DRM-protected multimedia contents, we have focused mainly to the usage of newly proposed MPEG-21 standard, (International Standards Organization [ISO], 2005), (Burnett et al., 2006), which has been recently proposed for primary use in the area of multimedia world, allowing the seamless, interoperable, transparent and universal delivery of multimedia digital contents to the end-users in a dynamic environment. More specific, our research work involved usage of two parts of the standard, (a) Part 4, MPEG Intellectual Property Management and Protection (IPMP), which provides mechanisms for protection of digital item, since security problems may arise from the fact that, the digital item's description, i.e. its structure, contents, attributes and metadata, is a clear XML document and it is easily visible to anyone and vulnerable to non-authorized usage; and, (b) Part 5, MPEG Rights Expression Language (REL), which provides a simple XML-based data model which allows to the content creator to meta-describe the license that describes the usage rules over a specific digital content. The use of MPEG-21, leads to a DRM scheme that is adaptive to the end-user needs, i.e. different users must have different usage rights over the same digital content, while also characterized by interoperability.

2. Security & Digital Rights Management (DRM) in user-converged multimedia environments

There have been considerable efforts to have audiovisual systems and applications converge, especially in home environments where homes can be considered as spaces of convergence, and for nomadic users with advanced mobile devices as points of convergence. State-of-the-art research initiatives have as primary scope to progress beyond home and device-centric convergence toward truly user-centric convergence of multimedia, having as long term vision "The User as the Multimedia Central", i.e. the user as the point at which services (multimedia applications) and the means for interacting with them (devices and interfaces) converge, (see following figure for the depiction of this vision).

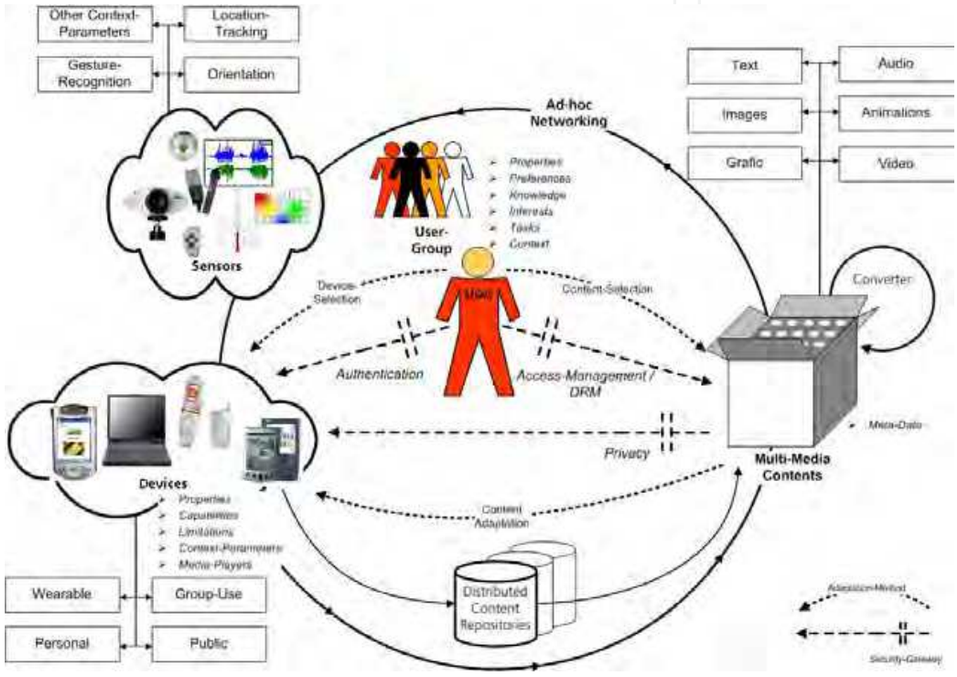

Fig. 1. User-converged multimedia services, (Intermedia, 2008)

As it can be seen to the previous figure, the user has access to different kinds of devices for interaction. From all accessible devices in a specific user context, the user selects his choice of devices himself and other devices are selected by the system automatically. The devices are used for multi-modal interaction and multimedia output. Based on the characteristics of the user(s) and the devices, the system selects the most appropriate content to be delivered. The content is available on distributed repositories including servers in the environment, networked repositories or small repositories hosted on small personal devices. When the best fitting content was found, it is delivered to the best fitting device that was selected beforehand. Potentially, the content needs to be transformed (e.g. to another media-type, another quality, or just in screen size) to fit the preferences of the user, the network bandwidth and the capabilities of the output device. In order to ensure private use of content and secure information transfer, the user needs to authorize to get access to different devices, the content selection process must consider DRM issues and the content presentation on the target device has to prevent displaying any private content on public displays.

One of the biggest challenges that we have to face in the deployment of architectures in such environments is related on the one hand, with the security and protection of digital contents that interchanged between users of such architectures and on the other hand with the provision to the users with security mechanisms that allow them to perform secure transactions (e.g. authentication, privacy protection, secure data transfer, etc.) in those environments. Intellectual Property (IP) protection is a mandatory request in modern multimedia architectures. The concerns of content providers for loss of revenues constitute a strong obstacle to the wide deployment of services that involve distribution of IP protected content. But the traditional content providers are not the only ones, if we consider the fact that today the end-users are equipped with different types of small devices that allow them to be the digital contents creators. In most cases, the end-users would like to have mechanisms which would give them the possibility to protect their artefacts and possibly to set their own usage rights over it, thus specifying toward third users how their digital content shall be used. Thus, it is a requirement to integrate DRM solutions in services and applications that target delivery of IP protected content that resides into the home network of the user. Considering the technical problems (system weaknesses) that result from add-on security solutions to independently developed network services, the approach of our effort to develop architectures with security and DRM as inherent requirements will lead to secure solutions that will increase the trust placed by content providers on the system and thus, it will lead to wider availability of services to a larger population.

2.1 The necessity of DRM

The rapid evolution of technologies for communication between computing systems in comparison with, (a) the high availability of broadband services at high data rates, (b) the great improvements in technology of hardware components of computers, and (c) the usage of Internet as a communication media for various entities, facilitate the convenient distribution of digital multimedia contents between users, most of the times illegally and without the necessary licensing.

The proliferation of the pre-mentioned technological achievements allow to the content creators, distributors, etc. to convey their profit instantly, while the end users to have the requested digital content in their hands, as soon as they requested and paid for it. But, after the contents creators and distributors send their goods to the end users they lose control over it. These combined with the fact that the end users have various digital technologies in their hands, which allow them to copy, and possibly redistribute digital contents, leads to significant loss of revenue for the digital contents creators and other interesting parts. So, in order to avoid such occurrences, the companies have developed and deployed various mechanisms which are used as countermeasures against unauthorized usage of digital content by non-legitimate users. Such mechanisms are defined under the umbrella of "Digital Rights Management", (DRM). *Digital*, since it refers to mechanisms for protection of digital contents; *Rights*, since the mechanisms allow the usage of contents under specific rights, which are set by the creators and distributors; *Management*, since the parties that deploy DRM over specific content have the ability to handle and manage its usage logically, even after the digital content has been distributed.

In general, Digital Rights Management (DRM) mechanisms constitute of various technologies that have been developed and deployed by content providers, creators, distributors, in order to protect their digital media from illegally, unauthorized and without

the appropriate rights usage of their products, while let them to use possibly unsafe media like Internet for delivering their products with less hesitation and anxiety about non-legitimate usage of their content. In a typical DRM system, digital content is accessed by the end-users according to access conditions and terms that are specified into a license that accompanies the content. In general, licenses are provided by the license creators, who create them according to directives specified by the content owners/providers/creators. Various methods and a numerous DRM systems have been proposed, and developed for protection of Intellectual Property (IP), either from the academic community or by digital content industry. What characterizes most of them is the lack of interoperability, i.e. different content providers use different non-standardized non-interoperable DRM systems which may create great problems in contents usage by legitimate users.

But, what exactly means interoperability in the context of DRM? Gasser and Palfrey (Gasser and Palfrey, 2007) define interoperability looking issues regarding the different DRM stakeholders, i.e. users, content creators/owners and vendors, stating that, "In the context of DRM the term interoperability encompasses consistent functioning of the overall system including security and access, such that the system is able "mutually to use" information in the form of usage rules, content and technical measures "in all the ways in which they are intended to function". This would apply even when content from different interoperable services is used and when such content is used on different interoperable devices. For the consumer, interoperability means he can choose different devices and use them with different services. For the content producer or content aggregator interoperability means he is not locked in to one distribution channel that forms a gatekeeper to the marketplace. For the device and ICT developer, interoperability means that his products can be used with different content services – and that a gatekeeper does not form around a specific DRM technology."

A more formal definition of DRM interoperability has been given by Heileman et al, (Heileman et al, 2005), who mention into their survey paper that, "It seems that everyone has a notion of what interoperability means, which generally revolves around the idea of "things_ working together. A slightly more formal definition related to technology is: "*The ability of one technology to interact with another technology in order to implement some useful functionality. It is possible to make nearly any two DRM technologies work together in a manner that satisfies this definition. Specifically, by building translation services, it is often possible to make one DRM regime work with a different DRM regime. At the current stage of development of DRM markets, this approach to interoperability makes sense. However, in order to facilitate the continued development of DRM markets, more detailed notions of interoperability of DRM technologies must be developed. In this sense, the real issue is not interoperability per se, but rather the level of interoperability that allows better DRM solutions to be created…*"

2.2 Communication & networking technologies

Wireless technologies represent a rapidly emerging area of growth and importance for providing either ubiquitous access to a backbone wired network or formulate autonomous ad hoc wireless networks. The Access Point (AP)-infrastructured wireless networks architecture is based on at least one AP providing a server function. All kind of communication between all wireless nodes should pass through this AP. This AP might be connected to a wired backbone network as well. On the other hand, mobile Ad hoc Networks (MANETs) are autonomous networks consisting of routing nodes (or some

routing nodes with other nodes that do not route) that are free to move about. They may be connected to a larger network e.g. the Internet, or operate as an isolated intra-network. Wireless networks can be categorized according to the extent of their coverage area into: Wireless Local Area Networks (WLANs), Wireless Wide Area Networks (WWANs), and Wireless Personal Area Networks (WPANs).

Recently, industry has made significant progress in resolving some constraints to the widespread adoption of wireless technologies. Some of the constraints have included bandwidth and high infrastructure as well as service cost. Wireless technologies can support and provide cost-effective solutions. Wireless is being adopted for many new applications: to connect computers, to allow remote monitoring and data acquisition, to provide access control and security, and to provide a solution for environments where wires may not be the best solution.

As networks become more and more complicated and applications more and more demanding, a very common network topology for state-of-the-art multimedia is a hybrid wired/wireless architecture. Hence, the need for interoperability of heterogeneous networks with hybrid structure is in doubtfully a major requirement, when integrating communication scenarios for indoor and outdoor applications.

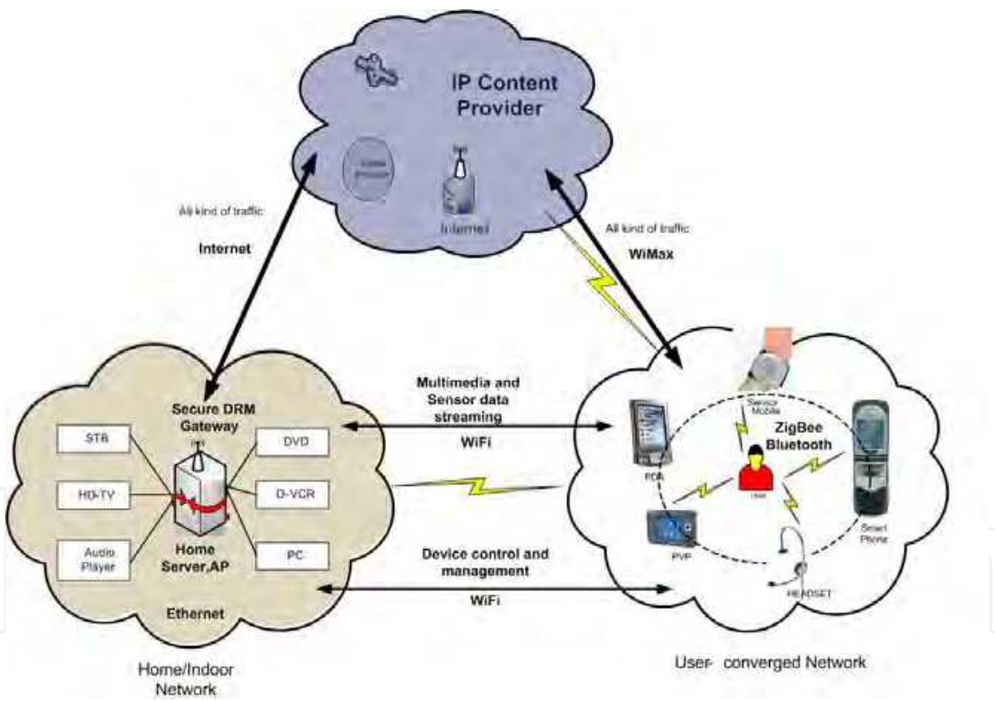

Fig. 2. Depiction of an architecture that consists of the Home Network, the user-converged network and the content provider.

On the other hand, when dealing with a hybrid wired / wireless network, questions arise regarding Quality-of-Server (QoS) and power awareness issues especially concerning the wireless part of the hybrid network. Integration of QoS and power awareness in wireless networks is nowadays a growing research area as high throughput, timeliness and power efficiency is demanded by several home and other applications. Thus, trade-offs especially between QoS parameters and power consumption must be considered to a network with

mobile nodes. Thus, the objective is to define and depict an integrated architecture of a DRM system in a user – centric framework involving indoor and outdoor heterogeneous networking conditions for multimedia and sensor applications.

When dealing with multimedia and sensor applications for both indoor and outdoor networking environments, the user-centric approach should be taken into consideration, (see Figure 2 for a depiction of a user-converged architecture). Dynamic networking, multimedia sharing, content adaptation, personalized interfaces and data security are only some of the main challenges when dealing with the user-centric approach. This approach involves aspects of both integration and convergence. The term integration refers to the fact that different networking techniques should be incorporated in order transparent network communication to be achieved. On the other hand, convergence in such systems corresponds to seamless content sharing among multiple devices connected to each other. Moreover, as long as both indoor and outdoor networking is concerned, the user-centric approach can be split out into the home-centric and device-centric convergence. Multimedia and sensor convergence allows the end-user to adopt various communication interfaces as they become available, transparently, without any interruption.

According to the given parameters, all data are adapted to the available device characteristics.

The framework network architecture, in terms of networking infra-structure, consists of two physical media, one wired and one over the air. The heterogeneous topology corresponds to the wireless WiFi, Bluetooth, ZigBee or WiMax standards and to the wired Ethernet and Internet based protocols.

Technically speaking, in an indoor/home environment, the user, with the aid of a PDA, can communicate with various multimedia and sensor devices as well as the home server through the WiFi (IEEE 802.11b and g) technology, supporting multimedia and sensor content applications. WPAN standards like Bluetooth (IEEE 802.15.1) or ZigBee (IEEE 802.15.4) are more appropriate for connecting to a PDA several wireless devices such as, oximeter sensor, for heath care monitoring or head sets for entertainment, within the body area of the final user. On the other hand, in an outdoor environment, the user has the ability to communicate with the home server through a WiMax (IEEE 802.16) access point and a Home Gateway internet-based link. In that point of view, both ad hoc and infra structured topologies are engaged in regard to wireless communications. As holds for all multimedia and sensor applications or possible communication scenarios, the user perceptive quality defines tight QoS requirements that the heterogeneous network should support.

2.3 General requirements

Considering the previous mentioned aspects of security, DRM and communication technologies, we can make some initial assumptions regarding the design and deployment of such an architecture which will expose security & DRM mechanisms to its end users. To be more specific, our architecture has to fulfil the following basic requirements,

1. Protect digital content from unauthorized usage from non-legitimate users, which are not authorized to have access to it;
2. Operate over heterogeneous wired and wireless network technologies;
3. Provide to the end-user friendliness, i.e. the part of the DRM system that handles access, management and rights of digital content should be very friendly and easy to be accessed by the end-users;
4. Provide adequate security mechanisms, under possible attacks;

5. Define the extent to which our system shall be interoperable with other systems. More specifically, we stress into this requirement since the interoperability is a major requirement of the modern DRM architectures in the context of the multimedia world;
6. Provide extensibility and renewability, i.e. we would like our system to be easily adaptive and flexible under changing requirements, new ideas, new business models, etc;
7. Provide trustworthiness, which means that the DRM system must reassure that it will behave according to the terms and rights that have been agreed;
8. Protect the privacy of the participating entities in value chain, i.e. a misuse of a DRM system could lead to user's privacy breaches;

2.4 The DRM process model

As we have mentioned, one of the key issues in user-converged multimedia architectures is related to the Intellectual Property of contents being distributed. As a matter of fact, multimedia distribution across several kinds of devices and networks, which can communicate among them, facilitates the convenient distribution of digital artifacts like audio, video, etc., among users, most of the times illegally and without the necessary licensing. Digital Rights Management (DRM) mechanisms consist of various technologies that have been developed and deployed by content providers, creators, distributors, in order to protect their digital media from usages that may be illegal, unauthorized and without the appropriate rights of their products. At the same time, DRM facilitates the use of possibly unsafe media like the Internet for delivery of these products with less hesitation and anxiety about non-legitimate usage of their content. This problem is even more stressed in Personal Area Networks, where content distribution among users and their usage generally takes place in an autonomous and uncontrollable form, due to the likely absence of a public connection.

One of the bases of our architecture is the identification of the stakeholders in the DRM chain, in order to define and understand more deeply the security requirements of the proposed DRM architecture. We can identify three major partners in the chain, i.e. the content creator, content (license) provider and the user, see following figure, (Fig.3),

Fig. 3. The three key stakeholders in a typical DRM system.

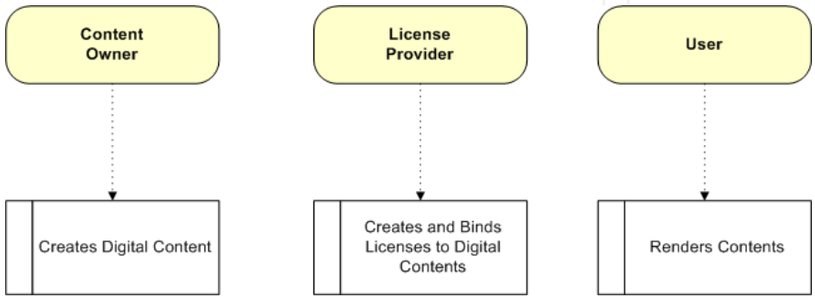

Fig. 4. A DRM architecture stakeholders and their main roles and functionalities.

The role of the content creator is to generate digital content and to gain profit by providing it to intermediate resellers like, for example, content distributors, who will finally resell it to the end-users. The user access the digital content according to the directives (access rules) that have been set by the content owners and implemented in accordance with the digital content by the license providers. Schematically the role of each part can be seen at the following figure, (Fig.4).

Furthermore, trying to identify in more depth some of the functionalities of each part, we can identify, (a) the license creator has to generate a license according to the directives of the content creator, while after the creation of the license it has to bind – encapsulate it to the digital content, thus creating a packaged content which delivered to the end user; (b) the end user must be equipped with a DRM client that will be comprised of a part that process the packaged content and a part that is responsible for the rendering of the content. Schematically, this process model can be seen at the following figure, (Fig.5),

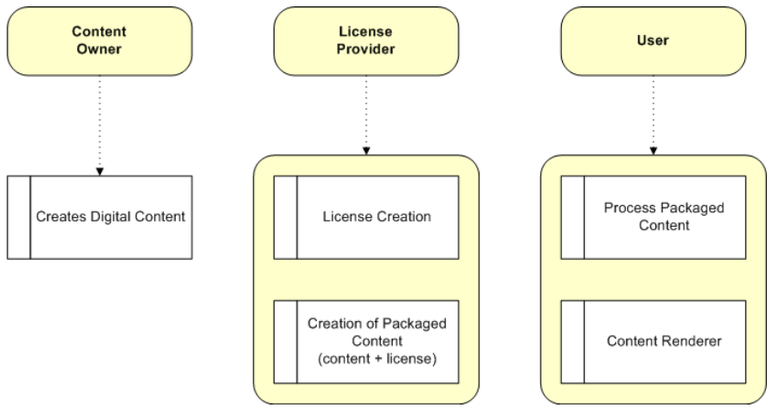

Fig. 5. DRM model that incorporates the packaged content and the breakdown of the content rendition component.

In the context of user's environment, the content owner could be the user by himself, who creates his own digital contents and would like to protect them and possibly share them in a controlled manner with other users, in his vicinity. Summarizing the previous we can identify a security requirements concerning contents owner view, as follows,

a. The digital content shall be accessed by the end user through a device, which shall be able to enforce the digital rights that have been set into the license, which has been provided by the license provider and accompanies the content. Analyzing this a bit furthermore, we can identify the following,

i. License generation for digital content shall be done only by authenticated license creators, i.e. authorization of license creators

ii. The content has to be protected from possible disclosure while communicated towards third parties, i.e. confidentiality of data.

b. The digital content shall be accessible only by a multimedia rendered that shall be capable to "decode" the license that accompanies the content, which comes from the license creator and defines the terms of contents' usage.

c. There must be set mechanisms that assure the integrity of communicated data between the license creator and the end-user, while there must not be any delays in the provision of the data. More specifically,

i. In the client's side renderer, there must be a part that is assumed to be trusted, which shall be responsible for storing the digital content, the license and any other sensitive data. Moreover, that part shall order the rendering of the content and shall also "monitoring" the contents consumption under the terms that have been specified by the license.
ii. Availability to the end user to have access to the digital content and the license, whenever he/she wants
iii. Correct reception of digital content/license by the end user, (integrity mechanisms)

d. The architecture, especially in the client's side, must be equipped with mechanisms tolerant to possible attacks, constraining the impact of the attacks,

i. Prevent spreading of an attack in the whole architecture, i.e. identify possible mechanisms that isolate attacks to a specific part of the architecture, not allowing it to affect the rest parts of the architecture,
ii. The architecture must be easily adaptable to new security requirements that may come to the near future

e. Protection of user' privacy and personal data, i.e.

i. The license creator stores only the private and personal data that the end user permits,
ii. No third party may have access to any personal data of the end user, without agreement between end user and the third party.

Regarding the end user, the following requirements must be satisfied,

i. Authentication between the end user and license creator, both sides, i.e. the license creator has to be authenticated towards the end user, while the end user has also to be authenticated to the license creator,
ii. Assurance that the license creator has set to the license only the terms that have been agreed between the two parties, (end user and license creator),
iii. The data for the authentication of the end user are only available to the end user,

2.5 DRM system – internal components

A typical DRM system involves three main parts, a. the content provider, who simply offers contents under specific rights; b. the DRM platform, which comprises the main part of the system, providing DRM functionalities i.e. secure storage, security primitives, contents provision interface, licensing translation – phrasing, etc; and c. the client's system, which refers to the end user's device that consumes the digital content.

Going one step further, decomposing the DRM platform, we can identify its main subsystems and primary functionalities as it can be seen in figure 6.

The following parts are identified: a. Content provision module (CPM) is the interface between the content provider and the DRM platform. The connection between them can be achieved through a trusted channel e.g. SSL protocol, which shall be established between content provider and content provision module; b. Secure storage, is a secure and trusted environment into which the content shall be stored after it's transferred to the DRM platform. In general, the content is stored in an encrypted form. It may be possible to store and other sensitive information like cryptographic keys, licensing information, etc; c. Licensing Translation Mechanisms, (LTM) which refers to the part that is responsible for the translation of the licenses that accompany the digital contents. The license defines the certain terms and conditions under which the digital content may be used and it is defined by the content provider. This module may include functions that allow the generation of

licenses for a specific content by the instructions that set by the content provider. The licenses are stored at the licenses repository, which can be assumed as a secure environment; d. Content preparation module (PREP), which is the part of the DRM platform that prepares the digital content for distribution to the end users. It performs various functionalities like content's encryption, transcoding of the content depending on user's preferences, enrichment of content with licenses and other metadata. The license is "mixed" in a secure manner with the digital content and the end user receives the digital content as a single secure container; e. Content Distribution Mechanisms, (CDM), which is the part that handles the distribution of digital content through various distribution channels; f. Authorization module, which is responsible for the authentication – authorization of the end user. As soon as the user is identified as a legitimate one the authorization module sends to the user decryption keys.

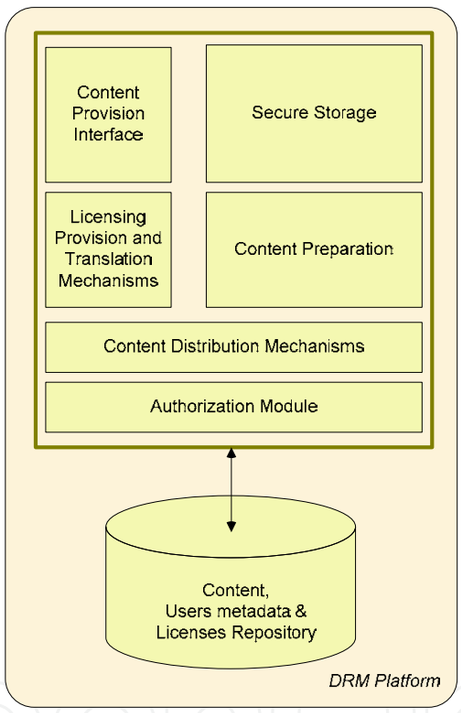

Fig. 6. The main parts and functionalities of the DRM Platform.

Furthermore, (Becker et. al, 2003), it has to be identified the flow of information in typical DRM system, between the content provider, the DRM platform and the end user, (see following figure, for a schematic view).

In step 1, the content provider contacts the CPM establishing a channel for transferring the digital content to the DRM platform; it also sends to the LTM module the license for content's usage or necessary information for its generation. The digital content is stored temporarily to the secure storage space, (step 3) and the license may be store at the licenses repository. In step 4, the PREP module prepares the digital content to be distributed, i.e. applying watermarking, encryption, transcoding – if necessary and licenses embodying. After that the digital container is transferred to the end user's client system through delivery networks, with the help of CDM module, (steps 5, 6). This container is stored in a secure

storage space, which resides into client's system. The user interrogates the authorization module of DRM platform, requesting access for the received digital content. If the DRM platform identifies the user as a legitimate one, it sends to him a "grant access" and some security information related with the specific digital content, e.g. keys for content's decryption. Then the content is decrypted into the client's system and it's rendered according to the usage rights that set by the accompanying license. The user's client system must be equipped with mechanisms that allow the enforcement of digital rights that defined by the license; as an extension, the DRM platform may be kept informed about the rights enforcement.

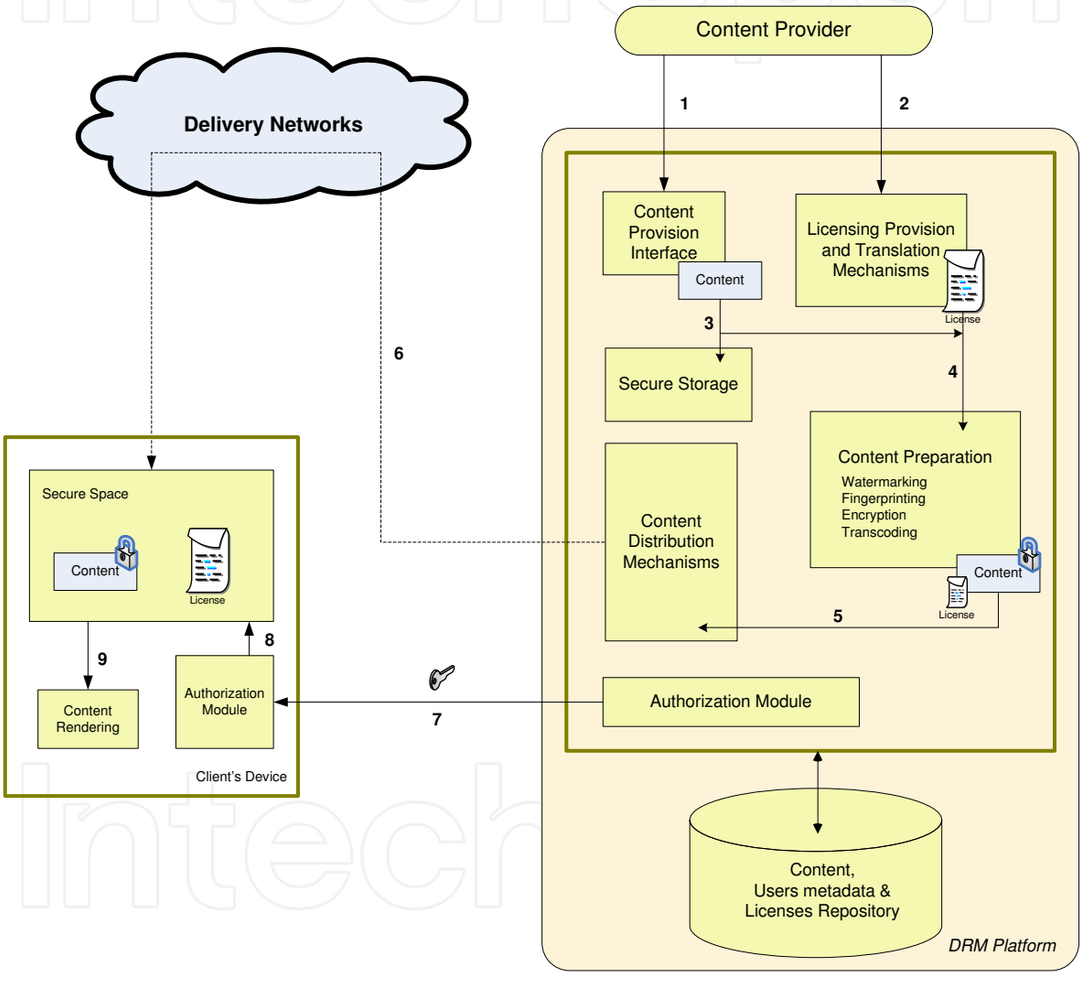

Fig. 7. How information flows in a typical DRM system.

3. MPEG-21 multimedia framework standard

MPEG-21 (International Standards Organization [ISO], 2005), (Burnett et al., 2006) is a standard that defines mechanisms and tools as means of sharing digital rights, permissions and restrictions that are set from content creators over digital contents regarding their contents usage from consumer. It is an XML-based standard that is designed to

communicate machine-readable license information in a ubiquitous, unambiguous and secure manner between peer entities.
MPEG-21 has been proposed for primary use in the context of multimedia world, allowing the seamless, transparent and universal delivery of multimedia content to the end-user, thus solving any interoperability issues.
It is based on two essential concepts: the definition of a fundamental unit of distribution and transaction, which is the Digital Item, and the concept of users interacting with them. Digital Items can be considered the kernel of the Multimedia Framework and the users can be considered as the entities that interact with the digital items inside the Multimedia Framework. At its most basic level, MPEG-21 provides a framework in which one user interacts with another one, and the object of that interaction is a Digital Item. Due to that, we could say that the main objective of the MPEG-21 is to define the technology needed to support users to exchange, access, consume, trade or manipulate Digital Items in an efficient and transparent way.

3.1 MPEG REL

In general, a Rights Expression Language is a XML-based machine-readable language that allows the declaration of rights and permissions concerning the usage of digital contents. The MPEG REL, (ISO, 2005), provides flexible and interoperable mechanisms to support transparent and augmented use of digital resources throughout the DRM chain, thus protecting the digital resources and the rights and conditions, which are specified for them. For instance, it provides mechanisms in support of publishing, distributing, and consuming digital content such as electronic books, digital movies, digital music, broadcast content, interactive games, computer software, and other creations in digital form. It also supports specification of access and usage controls for digital content in cases where financial exchange is not a term of use, and supports exchange of sensitive or private digital content and personal information. The MPEG REL may provide guaranteed end-to-end interoperability, consistency, and reliability among different systems and services. In order to achieve this, it offers richness and extensibility in declaring rights, conditions, and obligations; ease and persistence in identifying and associating these with digital content; and flexibility in supporting multiple usage/business models.

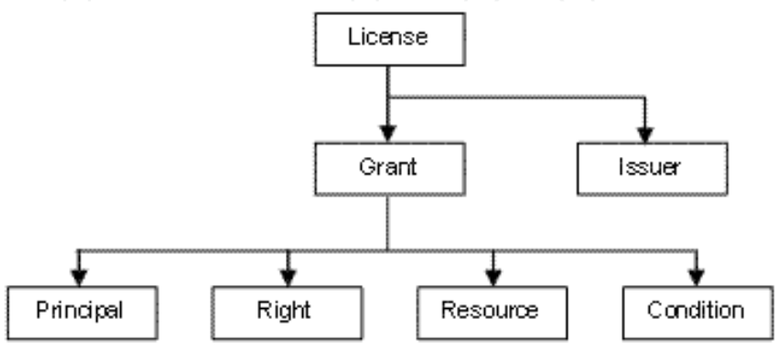

Fig. 8. A schematic view of a simple license that consists of a grant and an issuer.

The MPEG REL adopts a simple data model for many of its key concepts and elements. This data model for a rights expression includes four basic entities. The basic relationship among

these entities is defined by the MPEG REL assertion "*grant*" By itself, a grant is not a complete rights expression that is transferred from one party to another. A full rights expression is called a license and it consists of one or more grants and an issuer, which identifies the entity who has issued the license. Structurally, a grant consists of the following elements, (a) the principal, (b) the right, (c) the resource, and (d) the condition. In the following figure, we can see a schematic view of a license,

The MPEG REL defines the *r:license* element for the representation of a license into a digital item and comprises of two basic parts, (a) the *r:grant* element, that refers to whom the license is granted to and describes the license's main parts, and (b) the *r:issuer* element, which is optional and it contains information that identifies the issuer of the license. Structurally, a grant consists of the following basic elements, (a) the *r:principal* element, which defines to whom the grant is issued; (b) the *r:right* element, which defines an act of the principal to digital item; (c) the *r:resource* element, that defines the object on which the right in the grant applies to; and, (d) the *r:condition* element, that must be met before the right on the resource can be applied.

In the context of REL enhancement, it has been proposed (Delgado et al. 2005) an extension of the REL, in which it is adapted the use of a new element *protectedResource,* (see figure 9 for an abstract view of the elements), which give us the possibility to include into the license some sensitive encrypted information, for example, the key that is used for content's encryption, thus allowing key distribution to the remote end. As we have mentioned, this element comprises information that is protected with some form of (symmetric key and/or public key) encryption and it includes the following elements, (a) the *r: digitalResource* that specifies the digital content; (b) the *xenc:EncryptedData* a in which information about encryption of the resource is provided; (c) the *xenc:EncryptedKey* which contains information about encryption of the key that is has been used to encrypt the resource. Moreover, in the digital resource element, we can specify a hash code of the encrypted digital content, which can be used for the verification of the integrity of the received file to the remote end.

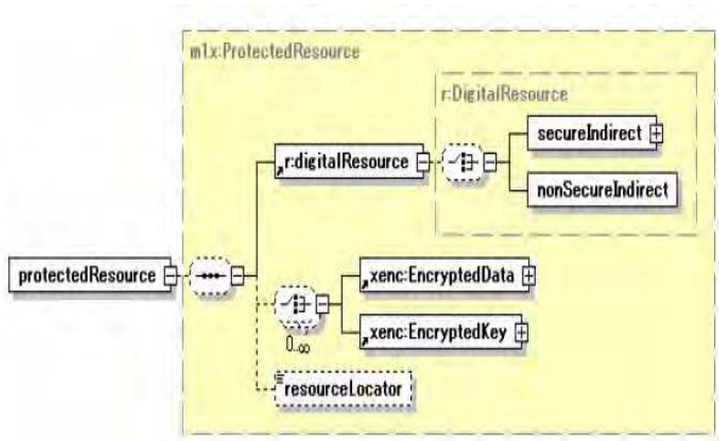

Fig. 9. Structural view of the protected Resource element.

In figure 10 we present an XML-view of a simple license, (comprising the basic elements), which has been extended with *protectedResource* element. Moreover, in figure 16, we present a detailed XML-view of *protectedResource* element, which is used to include into the license, information about the encryption key that we have used to encrypt the digital resource

song1.mpg.enc, with the use of 128-bit AES algorithm; furthermore, the encryption key has been encrypted with the use of public key transport algorithm RSAES-PKCS1-v1_5.

```
<r:license>
      <r:grant>
            <r:keyHolder>
                  . . . . .
            </r:keyHolder>
            <mx:play>
            <mlx:protectedResource>
                  . . . . .
            </mlx:protectedResource>
      </r:grant>
      <r:issuer>
            . . . . .
      </r:issuer>
</r:license>
```

Fig. 10. An XML view of a simple MPEG-21 license, where the *protectedResource* element is included.

3.2 MPEG-21 intellectual property management and protection

The security problems may arise from the fact that, the digital item's description, i.e. its structure, contents, attributes and metadata, is a clear XML document and it is easily visible to anyone and vulnerable to non-authorized usage. Due to that fact, the MPEG-21 includes a part named Intellectual Property Management and Protection (IPMP), (ISO, 2006), which provides mechanisms for protection of digital item. More specifically, MPEG-21 IPMP in conjunction with the MPEG-REL (Rights Expression Language) provide a framework that enables all users in the digital contents delivery chain to express their rights and interests in digital items and to have the assurance that those rights and interests will be persistently and reliably managed and protected across a wide range of networks and devices. The core notion in MPEG-21 IPMP is related with the IPMP tools that are used to protect the digital item. Those tools are not pre-described by the standard, but each user, vendor, etc., may define and implement his own set of tools, which perform basic security functions like encryption/decryption algorithms, authentication and data integrity mechanisms, watermarking, fingerprinting. With the use of MPEG-21 IPMP components, we may protect the whole Digital Item or a part of it through the encapsulation of the original DIDL elements that we want to protect, with additional information (IPMP Info) that refer to mechanisms and tools for the protection of the original elements. MPEG-21 IPMP defines a new set of IPMP DIDL elements, which have the same role and semantics as an element defined in DIDL. The structure of an IPMP DIDL element can be seen to the following figure, (Fig.11).

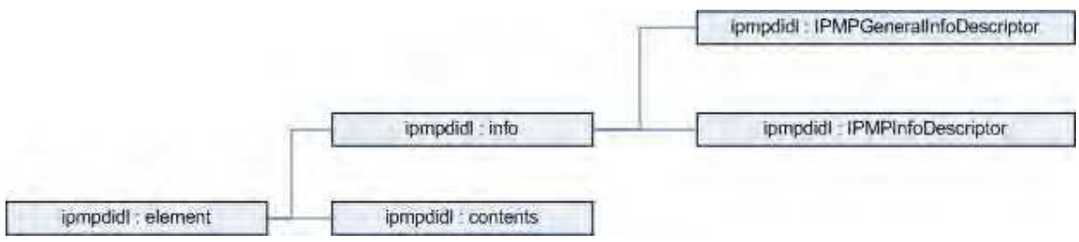

Fig. 11. Structure of an IPMP DIDL element.

The *ipmpdidl:info* element contains information about protection and usage rules of the digital content, which may be categorized to, (i) information about protection of the whole digital item, which is included in the child element *ipmpdidl : IPMPGeneralInfoDescriptor* and (ii) information about protection of a certain part of the digital item content, which may be categorized to, *ipmpdidl : IPMPInfoDescriptor*, see figure 11. Both pre-mentioned child elements have two purposes of existence,

a. to describe the tools that are used for digital items protection, and
b. to provide a set of licenses that accompany the content and define its usage rules.

4. Security-related aspects

In a user-centric environment that is enhanced with Digital Rights Management (DRM) mechanisms, we can identify different types of security requirements which have to be taken in care on the design and architecture implementation. Basically, we can identify tow different directions that may identify the security requirements that we have to take in care, named, (a) application-level security requirements; and, (b) embedded systems security requirements.

4.1 Application-level security aspects

A variety of security aspects can be identified when one tries to design and implement computing systems architecture. Those aspects are more emphasizing in the case of architectures that serve DRM functionalities. Our proposed architecture fulfils the basic security requirements, and in all cases we have tried to use state-of-the-art algorithms and methodologies. The basic security requirements that have to be fulfilled can be summarized as follows, (Schneier, 1996)

a. *Confidentiality*: it must be assured that stored or transmitted data are well protected from possible disclosure. In our approach sensitive information is stored, handled and transmitted in an encrypted manner.
b. *Integrity*: the data that have been transmitted are the original ones and have not changed medially; hashing methodologies are used when it is necessary.
c. *Authentication*: refers to the capability of mutual identification between various parties in a transaction. In our case we have used various types of authentication techniques, like public-key cryptography and digital signatures.
d. *Access Control*: it means that only legitimate users must have access to specific computing resources. This is the basic objective of our DRM architecture.

4.1.1 Authentication mechanisms

Authentication refers to methods and mechanisms which allow to an entity to prove to a remote end its identity, i.e. in a transaction between two end-users over a possibly unsafe communication network, there must be mechanisms that assure that each part can be authenticated by a remote end. In a DRM system, for example, a device or a user must be authenticated to the content provider's network in order to be able to participate in various transactions related with digital content consumption. User's authentication can be achieved depending whether the user: (a) knows something, e.g. a PIN, a password; (b) posses something, e.g. a token, a smart card, etc.; or (c) has something inherent, e.g. a biometric characteristic. Password based authentication mechanisms are quite weak, so we shall not

make any further comments. A strong authentication method is based on challenge-response protocols, where an entity (claimant) can prove its identity to a remote end by exhibiting the possession of a secret strongly associated with the claimant without the necessity to reveal it explicitly to the remote end. Challenge-response can be achieved with symmetric key techniques, public-key techniques and zero-knowledge protocols.

4.1.2 Encryption/decryption mechanisms

We define two main categories of cryptographic algorithms that are basically used for encryption/decryption of data; a. the symmetric algorithms and b. the asymmetric (public-key) algorithms. Roughly speaking, symmetric algorithms use the same key for encryption and decryption, while in asymmetric algorithms two different keys are used, a private one for encryption and a public one for decryption. Each participating entity must possess a pair of keys (a private key, *PKprv* which is held secretly from common knowledge and a public key *PKpub* which is publicly known).

Symmetric algorithms have the disadvantage of the use of a single key between entities that take part in encryption/decryption transactions, having as their main problem the efficient and secure key distribution between participating entities. Widely used symmetric algorithms are Triple Data Encryption Standard (DES), RC2, IDEA, Advanced Encryption Standard (AES - Rijndael), (Schneier, 1996), (Menezes, 1996). In a typical DRM system the digital contents are encrypted / decrypted using symmetric cryptographic algorithms. In the context of our work, we have implemented optimized AES algorithm.

On the other hand, the asymmetric algorithms (often called public-key algorithms) lack the problem of key distribution, since each part has a set of keys (private - public) and whenever it encrypts data it uses the public key of the part to whom encrypted data shall be transferred, while the recipient of encrypted data can decrypt those data using his private, thus deriving plaintext message, see following figure for a schematic depiction of pre-mentioned. Those algorithms rely on hard mathematical problems thus requiring high computational and processing capabilities, which make them inappropriate for encrypting/decrypting huge amount of data. Well known public key algorithms are the RSA (Rivest – Shamir - Adhelman) algorithm, the Diffie – Hellmann algorithm, and Elliptic Key Algorithms, (Schneier, 1996), (Hankerson et. al, 2004). In the context of our work, we have implemented the RSA algorithm.

4.1.3 Privacy issues

As we have argued before, whatever security mechanism is employed by a system, by any means it should take care and protect user's privacy. Especially, in the deployment of secure DRM systems, where there is an explicit relationship between end users and contents providers, with the latter ones having as their primary goal the protection of their digital assets, protection of end user's privacy is an issue that requires special treatment. As a recent example of a DRM mechanism that led to breach of user's privacy, we shall make a note in the Sony BMG case (Roush, 2006); SONY BMG in its effort to build a DRM mechanism to protect its commercial digital assets (CD's), embedded in each CD a "specific" program which it's proved to act as a rootkit , allowing hackers and non-legitimate users to gain access to end users computing devices without their permission, leading thus to violation of user's privacy. This was one of the biggest technological blunders in the history of modern computing leading SONY to recall its commercial products and publicly apologize for the scandal.

The Encyclopaedia of Cryptography [Henk et. al., 2005] defines privacy as an entity's ability to control how, when and to what extent personal information about the entity shall be communicated to third parties, while Anderson (Anderson 2001) defines privacy as the secrecy for the benefit of an individual entity, where secrecy refers to generic mechanisms that do not allow unauthorized usage and access of data and resources. There are various ways, mechanisms and techniques that we can employ to protect user's privacy. Adams (Adams 2006) classified the privacy's technologies for online environments into four levels; in level 1, he distinguishes technological and societal techniques, whether it based on technology (computers, devices and software) or humans, respectively; in levels 2 and 3, he classifies techniques based on actions that taken part from the participating entities in a scheme into which privacy's issues may arise, i.e. an entity may provide its personal information towards a third party, or an entity that holds a user's personal data may perform something with those data; finally, in level 4, Adams categorizes privacy techniques according to the threat model under consideration.

4.2 Embedded security aspects

The increasing capabilities of embedded systems combined with their decreasing cost have enabled their adoption in a wide range of applications and services, from financial and personalized entertainment services to automotive and military applications in the field. Importantly, in addition to the typical requirements for responsiveness, reliability, availability, robustness and extensibility, many conventional embedded systems and applications have significant security requirements. However, security is a resource-demanding function that needs special attention in embedded computing. Furthermore, the wide deployment of small devices which are used in critical applications has triggered the development of new, strong attacks that exploit more systemic characteristics, in contrast to traditional attacks that focused on algorithmic characteristics, due to the inability of attackers to experiment with the physical devices used in secure applications. Thus, design of secure embedded systems requires special attention, (Fragopoulos et al., 2009). Secure embedded systems must provide basic security properties, such as data integrity, as well as mechanisms and support for more complex security functions, such as authentication and confidentiality. Furthermore, they have to support the security requirements of applications, which are implemented, in turn, using the security mechanisms offered by the system.

In user-centric environments, the devices that are carried by the users, basically, are comprised of small and highly constrained embedded systems. It is the increasing capabilities of embedded systems, which combined with their decreasing cost, have enabled their adoption in a wide range of applications and services, from financial and personalized entertainment services to automotive and military applications in the field. Importantly, in addition to the typical requirements for responsiveness, reliability, availability, robustness and extensibility, many conventional embedded systems and applications have significant security requirements. However, security is a resource-demanding function that needs special attention in embedded computing. Furthermore, the wide deployment of small devices which are used in critical applications has triggered the development of new, strong attacks that exploit more systemic characteristics, in contrast to traditional attacks that focused on algorithmic characteristics, due to the inability of attackers to experiment with the physical devices used in secure applications. Thus, design of secure embedded systems requires special attention.

4.3 Trusted computing aspects

Trusted Computing[1] gives us the possibility to implement a Digital rights management system which would be very hard to circumvent. For example, in case that a user downloads a video, remote attestation could be used so that the video file could refuse to play except on a specific music player that enforces the creators' rules that specified into the accompanying license; this means that specific media players would be able to play user's music. Sealed storage could be used to prevent the user from opening the file with another player or another computer. The music would be played in protected memory, which would prevent the user from making a non-authorized copy of the file while it is playing; and secure I/O would prevent capturing what is being sent to the rendering module output. Circumventing such a system would require either manipulation of the computer's hardware, capturing the analogue (and possibly degraded) signal using a recording device or a microphone, or breaking the encryption algorithm. Utilizing Trusted Computing aspects could lead to new business models for use of software (services) over Internet. By strengthening the DRM system, one could base a business model on rendering programs for a specific time periods or "pay as you go" models. For instance one could download a music file which you only could play a certain amount of times before it became unusable, or the music file could be used only within a certain time period. Furthermore, Trusted Computing Components can be the core blocks of embedded DRM and security mechanisms in small, portable, limited-resources devices that are used by nomadic users. In the following part of this sub-section, we briefly describe some already proposed DRM architectures which utilize as core blocks Trusted Computing Environments aspects.

(Messerges and Dabbish, 2003) proposed DRM architecture for use in highly constrained environments like mobile phones, setting as a basic demand that the part of the DRM policy has to be implemented with the use of a trusted system. Their architecture comprises of a. the DRM manager, which is responsible for authentication of licenses and digital contents, for enforcement of digital rights and for content decryption; b. Security Agents, are in close co-operation with the security hardware and provide: (a) Memory management (secure storage), (b) implementation of basic cryptographic operations, and (c) Key management. After digital content's decryption is sent to a trusted agent, which is an application that is trusted to access and manipulate decrypted content data. The overall picture is depicted in figure 12.

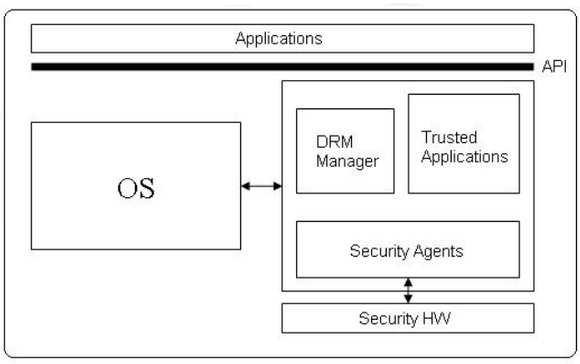

Fig. 12. A DRM architecture, which utilizes Trusted Computing Systems, (Figure has been adapted from (Fragopoulos, 2005))

[1]Trusted Computing Group, http://www.trustedcomputinggroup.org/

Based on the notion of Trusted Computing Systems, Lipton in (Lipton et al, 2002) and Serpanos in (Serpanos and Lipton, 2001) proposed the use of a special hardware agent, named spy, for protection of Intellectual Property in several environments. Their basic idea relies on the fact that even if the content provider side is secure, one cannot trust the client, which, in general, is considered as an un-trusted source. They proved that the existence of a special, tamper-proof hardware component that resides on the client system, as shown in Figure 13 is necessary to ensure protection of content provider Intellectual Property.

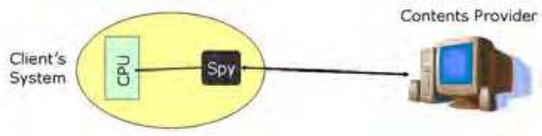

Fig. 13. Spy component as a trusted system for protection of Intellectual Property, (Figure has been adapted from (Fragopoulos 2005)

The spy is a tamper-proof hardware module, acting as a passive I/O system, allowing detection of I/O activity, having limited memory and computational power. It works in master mode in the client system and is monitoring the CPU. In a Video-on-Demand environment, for example, the user downloads the video from the content provider and stores it temporarily in the RAM of client system. There is an application that renders the digital content and it is assumed that it is the only application that runs on the system. If the user tries to make non-legitimate use of content, e.g. copying or transmission of the reproduced (decrypted) data, then some kind of I/O activity must take place at the client system, such as a disc access or a network transmission; such activity will be noticed by the spy, which, in turn, will notify the content provider in order to stop video transmission. Figure 14 shows the operation of this simple protocol, which performs the previous mentioned actions.

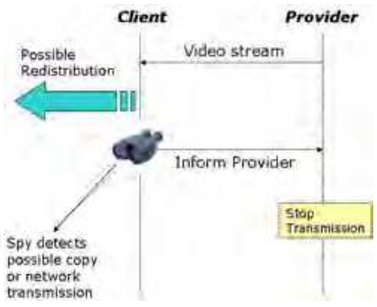

Fig. 14. A simple protocol demonstrating spy operation. (Figure has been adapted from (Fragopoulos, 2005)

4.4 Inter-networked embedded systems security, privacy and dependability aspects

Security, privacy and dependability (SPD) for systems and services that are built from integrated and interoperating heterogeneous services, applications, systems and devices is a very important aspect which must be ensured. Such systems and services must be robust in the sense that an acceptable level of service is available despite the occurrence of transient and permanent perturbations such as hardware faults, design faults, imprecise specifications, accidental operational faults, and deliberate, malicious, attacks.

The main goal is to address the upcoming impact of the Internet of Things to security, privacy, and dependability, from the early stages of design up to final deployment thus, creating new market opportunities by enhancing security, privacy and dependability so as to increase people's confidence in applications, systems, devices and infrastructures that were considered vulnerable or untrustworthy in the past.

Therefore, one target is to enhance security of Embedded Systems (ESs) as stand-alone or networked systems, i.e. at both the node and the network level while special focus should be given to developing technologies for: efficient, reliable, adaptable, and dependable ESs:

- ESs that defend against malicious attacks from intruders, maintain the confidentiality of sensitive data and protect intellectual property.
- Efficient and reliable communications and dependable networks for and utilizing ESs.

Another target is to develop appropriate ES technologies enabling protection of critical public infrastructure, such as transportation/communication/utilities networks and public building/areas. In this respect, special focus will be given on developing ES technologies to:

- Improve mobility of people and goods while preserving privacy;
- Provide support for critical applications, such as protection of infrastructures.

Solutions should contribute to one or more of the following specific objectives:

- Definition of a common conceptual framework to address the requirements for security, privacy and dependability.
- Instantiation of this framework with architectures, components, methods, interfaces and communications, tools and tool chains, to enable the design, development, analysis, validation, and deployment, as well as certification (or qualification).
- Trusted service platforms supporting the governance of the Internet of Things.
- Seamless and secure interactions and cooperation of ESs over heterogeneous communication infrastructures.

Indicative application examples imposing increased SPD requirements

- Telemetric monitoring of vital parameters of patients with chronic diseases is recognized to improve their medical condition and hence their quality of life. As a result of this, there are plenty of products and solutions for personal health monitoring available today that acquire physiological data in real-time. In order for such systems to be widely acceptable and utilized by the medical community and the patients, they must be developed satisfying the security requirements imposed by real-time data communication and protection of sensitive physiological data and measurements, data integrity and confidentiality, and protection of the monitored patient's privacy.
- Wireless Sensor Networks, which can be used as backbone networks in order to convey different types of digital contents that are locally generated by public utility applications. Such environments must be developed satisfying the basic security requirements for real-time, secure data communication, and protection of data and measurements, data integrity and confidentiality.

In such environments, it is imperative to design and deploy efficient and effective network architectures as well as a generic, if possible, communication interface targeted to connect external application networks with different types of "smart" embedded devices satisfying the basic security requirements for real-time, secure data communication, and protection of sensitive data and measurements, data integrity and confidentiality, and protection of the users' privacy. The architectures and the interface must consider the limited resources of the interconnected embedded systems, especially in light of the significant resources required

for implementing security in which, in general, are quite resource-hungry leading thus to significant technical problems.
By utilizing MPEG-21 standard's primitives, which define mechanisms and tools as means of sharing digital rights, permissions and restrictions for digital content from content creator to content consumer, we have shown that protection of transmitted "sensitive" information and enhancement of privacy is accomplished, since there is selective and controlled access to data transferred.

5. Architectural components of proposed DRM Architecture

A typical DRM system comprises of different types of components and should also provide different functionalities; shortly, we identify (a) Secure Storage Containers, which are used to protect the digital content from unauthorized access. Such containers can use various security functions like cryptographic primitives, trusted computing modules, etc; (b) Rights expressions tools, referring to the ways that the usage rights over a specific digital content are expressed. Basically those tools are combinations of languages (Rights Expressions Languages - RELs) with appropriate dictionaries. We have various examples of such methods like XrML[2], MPEG REL, OMA[3] REL, etc; (c) Description and identification of digital content, (digital items and metadata) methods. For example, MPEG-21 standard provides a well defined method for describing all digital content and its relevant metadata; (d) Identification of the involved parties in the chain of a DRM system – Authentication of interacting entities with the digital content; and (e) Forensic DRM components, refer to watermarking and fingerprinting techniques for proving subsequently if any rights violation over a digital content has occurred.
The basic requirements that are expected from a DRM system are:

a. *Interoperability*, which refers to the ability of the system to operate under different types of devices, platforms and architectures,
b. *Security*, which means that the system should provide robustness against possible attacks; and,
c. *Privacy*, i.e. whatever security mechanism is employed by a system, by any means it should take care and protect user's privacy.

Especially, in the deployment of secure DRM systems, where there is an explicit relationship between end users and contents providers, with the latter ones having as their primary goal the protection of their digital assets, protection of end user's privacy is an issue that requires special treatment.
To be more specific, the core components that we identify in our architecture are, (a) the digital content creator, who is the entity that generates digital contents that are stored into the DRM server, has the copyright of those contents, triggers the license generation procedure and has the possibility to assign rights to third parties to generate licenses for his contents, (root grants); (b) the Content/Profile Server, which is the part that contains the digital contents, it is responsible for controlling the whole DRM functionality; (c) a License Server (LS), which is the part that creates valid licenses according to DRMS directives that are associated with a specific digital item - the licenses are MPEG-21 compatible; and (d) the DRM Client, which is a device that is located into the Personal Area Network of the end user

[2]XrML - The Digital Rights Language for Trusted Content and Services, http://www.xrml.org/
[3]Open Mobile Alliance, http://www.openmobilealliance.org/

and it is capable to "consume" the digital content that is provided by the Content Server according to the directives that are specified by the accompanying license. Both the Content Server and License Server are located into the content creator, while the DRM client resides into the Personal Area Network of the end user, to whom the content creator wants to deliver protected digital content. In the following scheme, the three main components can be identified. Furthermore, three communication channels can be seen, (i) end-user devices and License Server, (ii) end-user devices and Content/Profile server, and, (c) License Server and Content/Profile server.

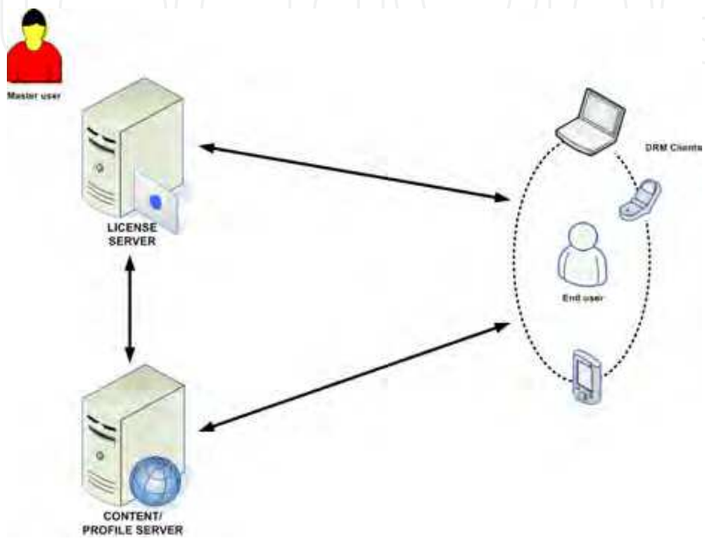

Fig. 15. Basic components and generic view of a DRM architecture with appliance to user-centric multimedia environments.

5.1 License server

The License Server (LS) accepts the request from the content's distributor, (in the general scenario), for the generation of an MPEG-21 compatible license. Initially, the LS execute the MPEG-21 authorization algorithm and in the case that its result is "yes", it creates the license and it digitally signs the license with the public key of the intended user. In a second phase the LS utilizing security mechanisms like TPM or various security libraries, it creates the packaged license, i.e. a simple concatenation of the MPEG-21 license and the content encryption key, then it encrypts the packaged license with the end users' public key and transmits the encrypted license towards the end user. In the following figures, we can see some diagrams that depict a simple scenario, where a user has acquired an encrypted digital content to his device and he requests a license from the distributor, who in turn redirects his request to the LS, (see Figures 6, 7). The basic functionalities that are implemented by the LS are,

1. Access and knowledge of the end-users' public key
2. Access to a set of root grants
3. Access to a set of MPEG-21 licenses, (optional)
4. Takes as input a request from the user that has the appropriate elements for the license creation, e.g. for which digital content, any conditions, *keyHolder* valid information, etc.

Then the LS executes the authorization algorithm, taking in care the users' request, the set of root grants and the possible set of additional licenses. If the result of the execution of the

authorization algorithm is "yes" then it creates a valid MPEG-21 license, it digitally signs it with the end-users' public-key and pass it to the module for the creation of the packaged encrypted license, which is finally transmitted to the user. In case of negative response from the authorization algorithm, the user is acknowledged for the result and the LS end the process.

5.1.1 Architecture of the license server

In order to facilitate the development and implementation as well as the analysis of the needs, the software architecture of the license server is divided into the following levels as they are depicted in Figure 16.

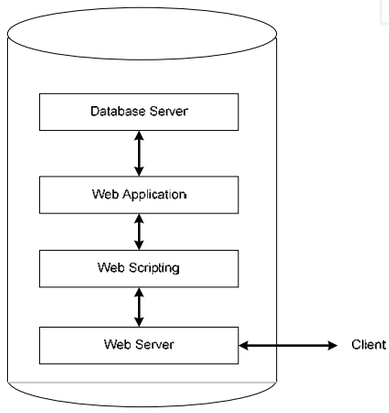

Fig. 16. Basic internal components of the License Server.

5.1.1.1 Web server

The web server is needed in order to provide the HTTP connectivity necessary to exchange the information and also provide an interface with basic messaging information such as to denote success or failure of the requests. This service is provided by the basic functions of the Internet Information Services. This allows an easy connectivity over any TCP/IP connection and instant compatibility with a variety of web-enabled clients.

5.1.1.2 Web scripting interface

A scripting interface based on the ASP scripting language is executed on the web server and it plays the role of the basic interface of the web service. This is the bridge between the HTTP requests and responses and the core functionality of the server. The ASP interface can provide simple mechanisms such as session management, user menu control and also deal with all interactions not related to the license creation mechanism itself.

5.1.1.3 Web application

The web application is a full scale (not scripting) application that is executed by the scripting interface. The web application, written in C#, executes the functionality of the main task of the application. Main tasks are as follows:

- Managing communication
- Authenticating user
- Accessing the license server application
- Encrypting license information

Communication is managed by one end via the scripting interface to the web peer and on the other application to the database server containing license information. As a result, the communication task of the web application is the bridge connecting end-to-end the communication parties. It is the communication task of the web application that translates all requests and messages. The other levels of the architecture simply adapt and transmit the information in the appropriate way. When the main application receives the requests for license creation from the user, it needs to check if the requested license can be legitimately created. In order for this to take place the user needs to be authenticated so this is an important part in the whole security of the system and it executed by the web application.
The last task of the web application involves the creation of the license for the requested content. The license is created after accessing the license server application and retrieving the necessary information. The license will be created in the format of an XML file and its content will be encrypted using the security libraries.

5.1.1.4 Database server

The database server is implemented with a Microsoft SQL server since this provides maximum compatibility with the IIS server and the C# modules. The database server is accessed by the main application and contains all the information related to users and content necessary for the creation of the license.

5.1.1.5 Communication protocol and methods

The proposed web connectivity method requires simple HTTP protocol use. It was chosen not to use security at this level, with the possible use of HTTPS protocol, since there is inherent security embedded in the content as the content of the license is encrypted. This allows avoiding the establishment of keys or the usage of certificates which would further complicate the communication.

5.1.1.6 License templates

As we have already stated the license is meta-described with the utilization of MPEG-21 standard. MPEG-21 contains a specific part, (Part 5), named MPEG REL, which provides a simple XML-based data model which allows to the content creator to meta-describe the license that describes the usage rules over a specific digital content. The use of MPEG-21, leads to a DRM scheme that is adaptive to the end-user needs, i.e. different users must have different usage rights over the same digital content, while also characterized by interoperability.
In the context of our work, the license files are not created dynamically from scratch, rather we have some pre-arranged template files that have some specific fields, which are updated with the corresponding parameters that are passed from the client's web interface towards the License Server, whenever it is requested the generation of a MPEG-21 license.
Considering the structure and creation mechanisms of licenses, as we have already stated the license is meta-described with the utilization of MPEG-21 standard. MPEG-21 contains a specific part, (Part 5), named MPEG REL, which provides a simple XML-based data model which allows to the content creator to meta-describe the license that describes the usage rules over a specific digital content. The use of MPEG-21, leads to a DRM scheme that is adaptive to the end-user needs, i.e. different users must have different usage rights over the same digital content, while also characterized by interoperability. The license files are not created dynamically from scratch; rather we have some pre-arranged template files that have some specific fields, which are updated with the corresponding parameters that are

passed from the client's web interface towards the License Server, whenever it is requested the generation of a MPEG-21 license. In Appendix (section 10.1), we present such a template for an MPEG-21 based license which sets rules and specifies rights for contents usage between two different types of mobile devices.

5.2 DRM client

One the core software components of our architecture is the DRM client application, which as it has been stated before is a software-based application that resides into the end user's device and it is responsible (a) for the enforcement of the license that accompanies a digital content, i.e. allow the unhindered and according to the license's terms "consumption" of the digital content; and, (b) for the rendering of the content.

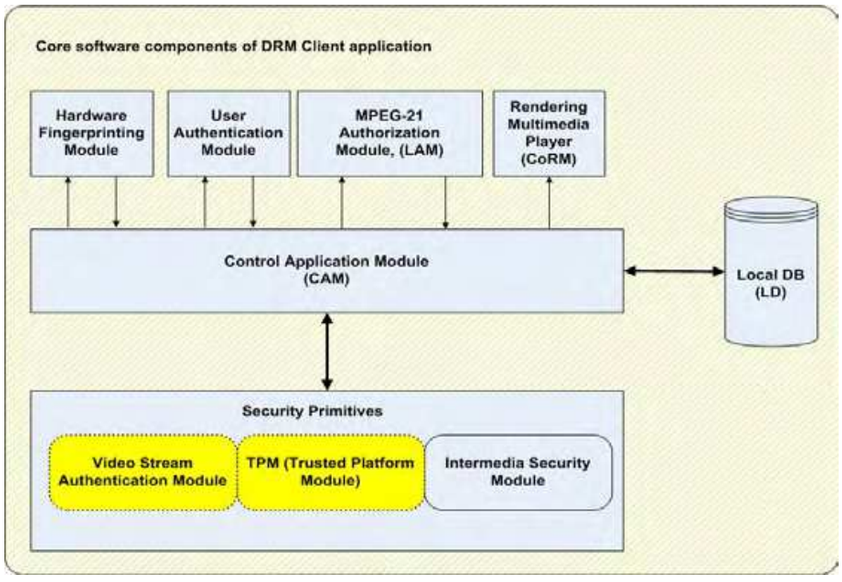

Fig. 17. The core components of the DRM Client application.

In a first approach, both the encrypted digital content and the relevant encrypted license are located in the device's file system, for example, both can feed to the user through a secureSD module. In an extended version, the DRM client is enforced with functionalities that allow to the user to access a pool of available digital contents and request the generation of valid licenses, in a near real-time manner. The basic core components of the DRM client application can be seen at the following figure,

5.2.1 User authentication

Whenever a user initiates the DRM client application, a user authentication process takes part, i.e. a user login interface pop-ups to the user, requesting some credentials (username - password), to be entered by the user, see following figure. Both the username and the password have been given to the user in a pre-arranged manner and are stored to the embedded database. When data are inserted a check is done on the fly and data are compared with those that are stored in the database; in case of erroneous situation the end-user is informed and the application terminates. Also, the user is informed in case that any other error occurs, e.g. for some reason the application cannot initiate a connection with the database.

5.2.2 Hardware fingerprinting module

The hardware fingerprinting module, mainly, is used in order to protect unauthorized delivery of both client application and transferred data (encrypted content and encrypted license) to third party devices. As an example of this, we can think the case in which a legitimate user gives both the DRM Client application and the encrypted data (license and content) to another user – to play it to his personal device, while also provides him with his credentials. To avoid such issues, in a pre-arranged manner, valid hardware fingerprints of devices are stored in the embedded database that accompanies the client application, (Figure 18 depicts Hardware Fingerprint creation and registration towards License Server).

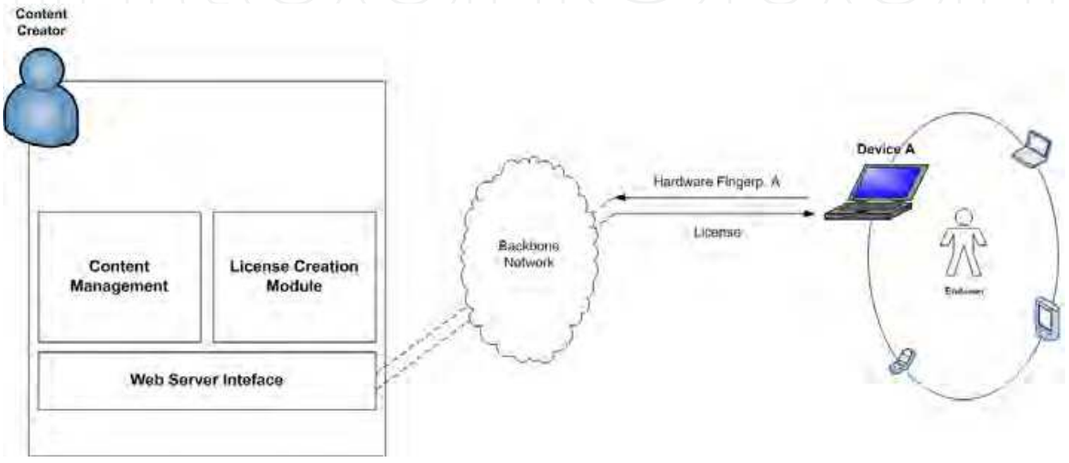

Fig. 18. Creation and registration of device's hardware fingerprint to the License Server.

The hardware fingerprint is a 16 byte unique identification code of a computer, e.g. 4876-8DB5-EE85-69D3-FE52-8CF7-395D-2EA9, and it is generated as message digest from the device's hardware components like for example CPU Id, its BIOS Id, physical hard drive information, motherboard Id, etc. Programmatically, we can get such information utilizing *System.Management* Namespace that is provided from the .net, which provides access to a rich set of management information and management events about the system, devices, and applications instrumented to the Windows Management Instrumentation (WMI) infrastructure.

The device's hardware fingerprint is generated in an a priori manner, i.e. the end-user has to inform the license/content provider about the device or devices that the user is intended to render the content. It is based upon the following device's unique characteristics,

a. CPU Id
b. BIOS Id
c. Motherboard Id
d. HDDs characteristics
e. Video controller ID
f. MAC Address of network card.

5.2.3 Web interconnectivity with license server

The DRM client needs to communicate with the License Server in order to request a license creation and then receive the license file. This communication must be protected using encryption since the license file contains sensitive information that allows decryption of

content. Moreover, the client could be simultaneously authenticating at the time of the license request so sensitive information such as a user PIN should be protected.
Since the amount of data that needs to be exchanged is very small, typically of a few kilobytes corresponding to the XML file containing the license, the methods used to communicate with the license server could be simple and straightforward. It was chosen to use web-based methods for this communication since this allows for easy development using existing web architecture frameworks and good interoperability between the distinct software components of the server.

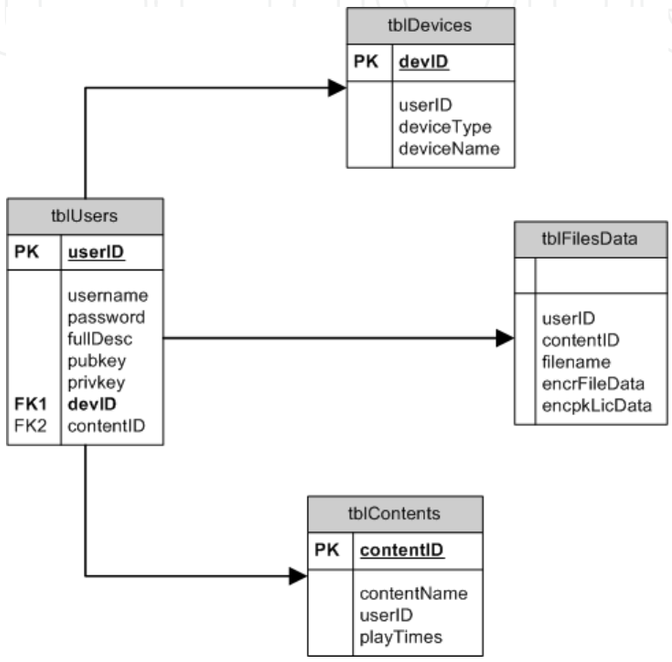

Fig. 19. DRM Client's application embedded database structure, in tables level.

5.2.4 Embedded secure database

An embedded secure database is used for storage and retrieval of data. In general, an embedded database system is a database management system (DBMS) which is tightly integrated with application software that requires access to stored data, such that the database system is "hidden" from the application's end-user and requires little or no ongoing maintenance. It is actually a broad technology category that includes database systems with differing application programming interfaces (SQL as well as proprietary, native APIs); database architectures (client/server and in-process); storage modes (on-disk, in-memory and combined); database models (relational and object-oriented); and target markets. For the purpose of our application, we have used Microsoft SQL Server Compact Edition 3.5 (MS SQL CE[4]), which is an embedded database that allows us to integrate it in our desktop and mobile applications. It takes about 1.5 MB on HDD and consumes about 5 MB of RAM. One of the biggest advantages of using MS SQL CE as the core DBMS system is

[4] MS SQL Server Compact Edition 3.5,
http://www.microsoft.com/Sqlserver/2005/en/us/compact.aspx

that it allows the encryption of the database during its creation with a master password; the password to access the database is hard coded into the application. So it provides a high level of security to the provider of the DRM client application towards the end-user. In the development level, the calls to the database are done with the utilization of C# and SQL primitives, i.e. all transactions to the database, for example data retrieval, insertion and update of data, are done through SQL queries and (INSERT, UPDATE, SELECT) statements.

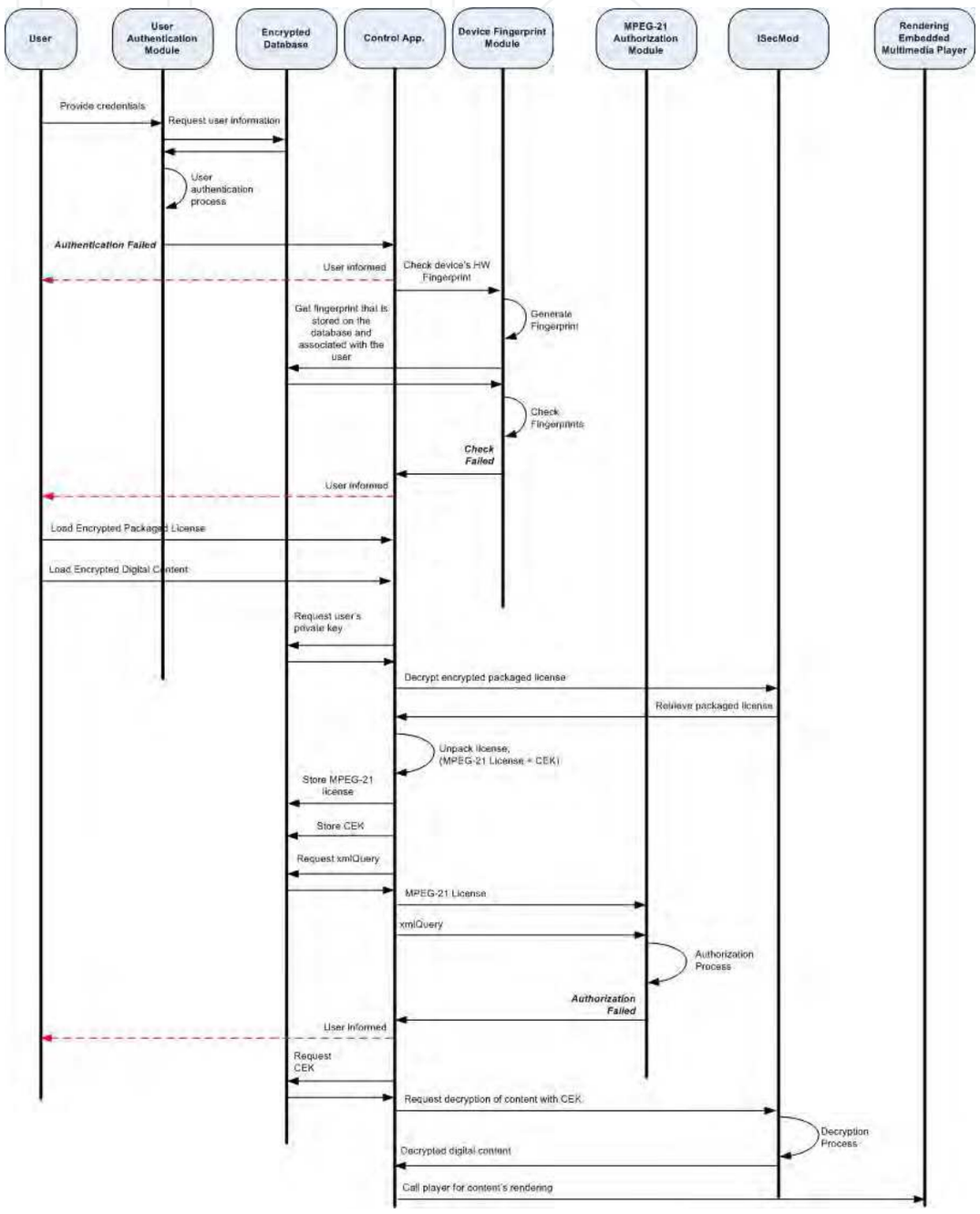

Fig. 20. Information flow between the core components of the DRM Client application.

5.2.5 Control application module

The Control Application Module is the core software component that it is responsible for the supervision, control and handling of the pre-mentioned software components. In the following figure it can be seen information flow (sequence diagram) between various internal components of DRM application.

6. DRM and security aspects in streaming data

In this section a description is done regarding, integration of pre-mentioned DRM architecture in a session migration framework. During session migration, a session, (e.g. rendering a multimedia content to device A), is migrated to device B, transparently and without any disruption to the end user. It is easily understood that such architecture, in case that it is needed to be enhanced with DRM functionalities, special methods have to be used, since DRM adds an overhead to normal operation. In (Repetto et al, 2010), it has been proposed an architecture for session migration in high mobility environments, utilizing with security and DRM aspects.

6.1 Session migration architecture integrated with security and DRM aspects

In a typical scenario for pervasive user-centric computing, the backbone architecture must provide to the end user the ability to have seamless access to digital contents in a variety of different types of devices, without loss of session; i.e. if a digital content is rendered on a PDA, when the user is located in front of another personal multimedia device, the session has to be transparently moved from the PDA to this device. This process has to be enhanced with security and DRM aspects.

As it has been stated before, the main objective is to generate a shared vision of user-centric multimedia services for modern nomadic people, who have high level of mobility and are connected to network for most of times. One of the main implications of the user-centric approach is seamless and secure access to content, regardless of its location and users' terminal device(s), which raises new concepts and expectations about Internet access, leading to the need for pervasive media architectures, which have to be aware of security and DRM aspects. Such pervasive environments rely on the presence of lots of devices in the user surrounding and on almost global network coverage, which is not an utopia ever since today, at least for what concerns 2D media: this is a key factor in fostering an interactive and continuous participation of users in the active content chain, but it is not enough to build user-centric systems. Indeed, users should interact with remote applications, content and services in a transparent, continuous and seamless way while having mechanisms that allow protecting their intellectual properties and digital rights, independently of the current device and access network.

From a networking perspective, our vision essentially targets secure user-centric media access in pervasive communication. In this scenario, users create their own content and store it at some location. That may be personal or public content made for profit or for amusement; in all cases it is likely that content has to be shared with other people. Content creators often wish to manage how other users access their content and what kind of operations those people are allowed on the content (e.g., rendering, copy, redistribution, editing), especially when they intend to get profit from it. This usually brings to the concept of license and Digital Right Management, but enforcing the observation of a license on

digital media is a very hard task; currently, it mainly relies on honesty and integrity of users. Several devices may be available for users in their environment: personal or public, fixed or portable, with different capabilities and interfaces. Moreover, a large number of network connections are available: public and private, wired and wireless, covering body, local, metropolitan or wide areas, with different services and performance. In modern computing environments, Digital Right Management is already an issue in itself, but it gets even harder in pervasive communication environments, where multiple devices may be used by users to access secured DRM-enabled digital contents.

In the typical scenario, end users play a leading and twofold role (see Fig. 19): they create content and in the meantime they also are content consumers. On the one hand, content creators are mainly interested in preserving their intellectual properties. On the other hand, consumers require a mobility infrastructure to build the pervasive paradigm. Three entities are required to cope with these tasks: content providers host user's media and make them accessible by other users (content sharing), a DRM system takes the responsibility of enforcing policy rules on content access, rendering and duplication (DRM management) and a service provider must account for user mobility (mobility management). Finally, information about the user's surrounding may really enhance the system (context awareness), providing local information about the physical environment, presence of devices, available services and so on. Figure 21 also reminds the initial approach and assumptions based on the user-centric paradigm.

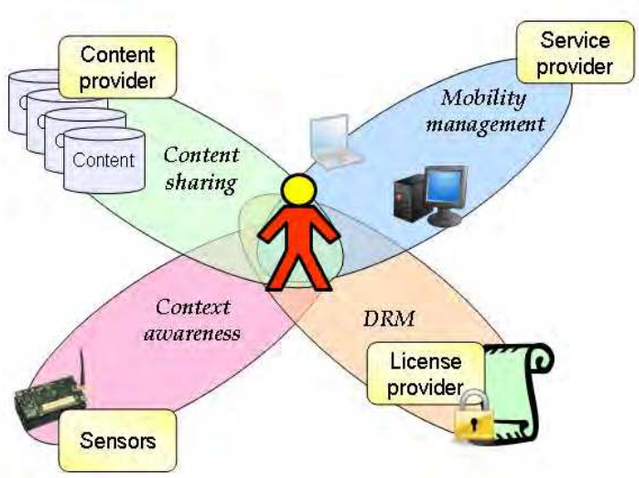

Fig. 21. Content access with DRM for mobile users in the InterMedia approach.

Figure 22 depicts the building blocks of the integrated approach. Content is available at some location (user's home or public repositories). The DRM subsystem is a trustworthy entity that knows about user's rights and policies to be applied to content; it enforces the user's client to not break the owner's agreement and terms of use (the license). As the picture shows, content does not need to be stored inside the DRM system, whilst the user's client must be part of DRM to avoid malicious users to bypass the permissions granted by the license. A mobility component integrates with the DRM system; it allows both terminal mobility (handover) and session migration (users can change their terminal without any session break). Note that no support at all is required at content location (transparency to third parties). The mobility framework includes localization facilities in order to automate

the migration process; users are tracked and their position is matched against device's location to find out the most suitable one.

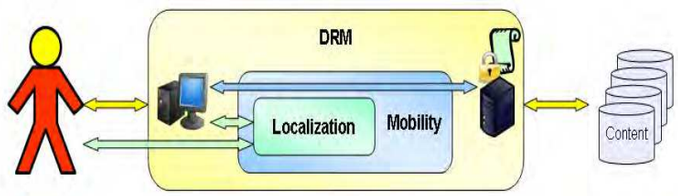

Fig. 22. Overall schematic system architecture.

In a typical DRM architecture, we can identify three main entities: (a) the digital content creator, who is the entity that generates digital contents that are stored into a server, which is located in users' premises. The user has the copyright of those contents and triggers the license generation procedure; (b) the License Server, which is the part that creates valid licenses according to user's directives that are associated with a specific digital item – the licenses are MPEG-21 compatible – and is responsible for controlling the whole DRM functionality. The license server may also contain the digital contents or it can have access to some private repositories where they lie; and, (c) the DRM Client application, which is a software application that resides into end user's device. This application is responsible for enforcing the license regarding digital content's "consumption". The basic actions, (see Figure 23, for a schematic depiction), that the content provider (the user that creates digital content) has to perform in the chain of a DRM protected media delivery towards a third party, are: (a) protect and secure the digital content; (b) generate a license that describes the usage rights over the digital content; (c) provide a secure way for contents' and licenses' transmission towards the end-user; (d) enforce the license on end-users' devices; and, (e) inform content creator/owner about the license enforcement status.

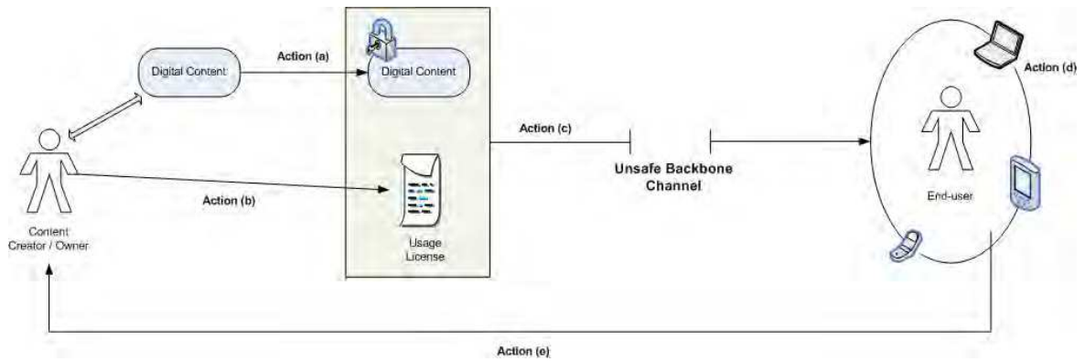

Fig. 23. Actions due to delivery of a DRM-protected digital content towards a remote entity.

6.2 Common testbed for architecture evaluation

We have implemented the whole architectural framework in a common test bed. Most of the architectural parts described in previous sections are already working; we are currently designing the internal architecture of the most complicated elements.

Our test bed specifically addresses multimedia real-time streaming, with integrated security mechanisms and DRM functionalities. It substantially implements the general scenario

described in previous section with three main actors: content creators/owners, content providers and content consumers. Content providers are trustworthy parties that create licenses on behalf of content owners and give access to their media. The implementation requires a Media Repository, a Content Server, a User Client and a Context Server (see Figure 24). These elements are made up of the architectural frameworks depicted in previous section; here we discuss how the whole framework is expected to work, describing the interfaces and relationships among the above components, without going into details about their internal software architecture.

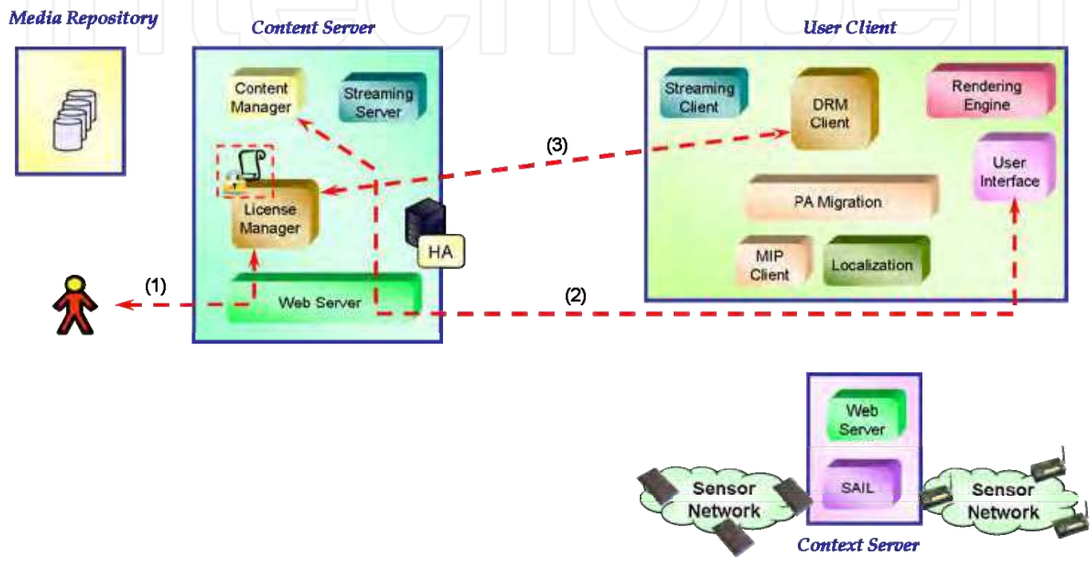

Fig. 24. Test bed for evaluating proposed architecture; initial signaling.

For the sake of generality, we assume content is stored anywhere in the Internet. Content creators register their media and the license policy by mean of some interface, for example a web interface (1). Users browse content list at the Content Server, again through the same interface (2). Once the user has selected the media, he requests a license to get it (3) and he is usually charged for this – License Manager is responsible for performing this action. User mobility implies the session may be handled by heterogeneous devices or even different implementations of the same application, and our proposed DRM system is ready to cope with this issue. Indeed, the license might be bound either to the user or to its device. Due to the pervasive nature of the scenario, the license is bound to the user, since this is a strict requirement to allow digital right management in pervasive environments. Thus, the License Manager component generates a license for that user (user-centric approach); this is possible by binding the license to a Personal Address (PA) which is assigned to the user at the same time (as well as all cryptographic material for MIP operations). Furthermore, as added security we add some device related information to the license, like for example a unique hardware fingerprint, that allows access to content only to authorized devices. This procedure makes the system dynamic. However, at the current stage of implementation we do not provide the web interface and a communication protocol between the DRM Client and Server components; instead, we get the license and the PA offline. The PA Migration adds the PA to the network interface and the MIP Client registers its current location with the HA, following the standard MIP procedure (either the client or the local FA addresses can be used.

The Content Server may also be split into two separate parts, content management (Content Manager, Streaming Server and Home Agent) and the License Server; however that would require an additional secure interface between them (the License Manager and the Home Agent (HA) have to exchange cryptographic material). Once the user requests the content, the DRM Client checks the license and enforces any restriction therein. If this check is successful, the Streaming Client requests the media by some suitable protocols; we chose RTSP for our implementation. The PA is used at this step, thus all packets are routed through the Mobile IP infrastructure (HA and eventually local FAs). This assures the request comes from an authorized client, as MIP tunnels are encrypted. At this point the Streaming Server locates the content and streams it, again through the MIP tunnel (see Figure 25). The Rendering Engine takes care of playing the media through the available interface.

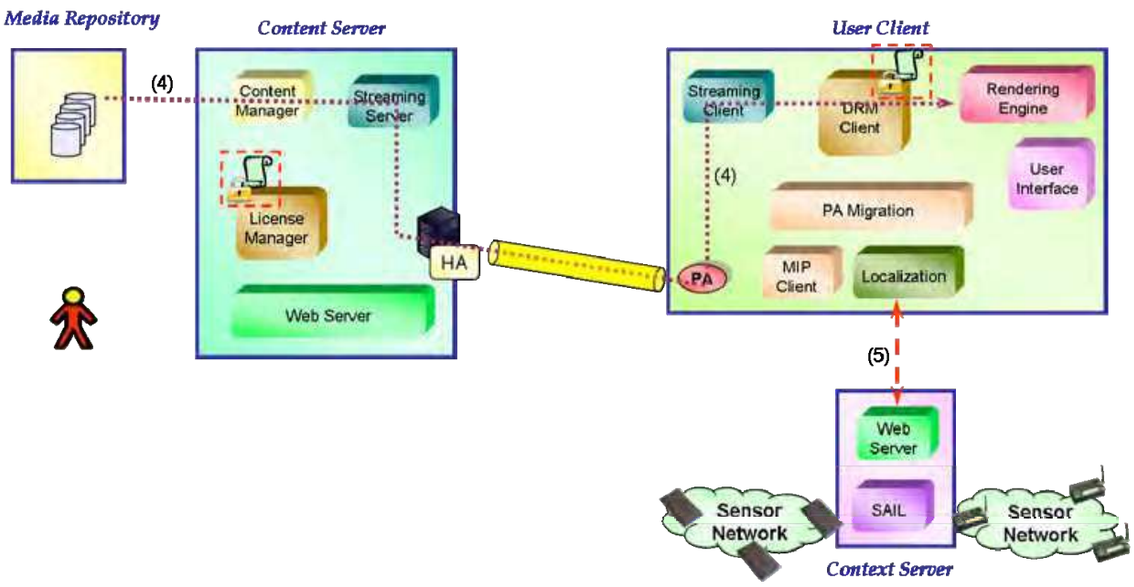

Fig. 25. The testbed architecture: media are routed through the MIP infrastructure.

At session start-up, the User Client also subscribes to the Context Server; for all session duration the Localization module polls the latter to timely detect any user movement and to discover any device close to him. Currently, we use HTTP for this interface. When the migration is needed, the PA Migration function saves the current context (license, PA, cryptographic material, media codec, transport ports, etc.) and transfers it on the new device; the MIP Client at the new device update the MIP registration and then session is restored transparently to any component behind the HA.

Furthermore, in Figure 26, we depict a sequence diagram that shows all the actions that take part, towards a session migration scenario between two or more devices. Moreover, in this figure it is depicted the request of device hardware fingerprint from the License Server, which uniquely identifies the specific device.

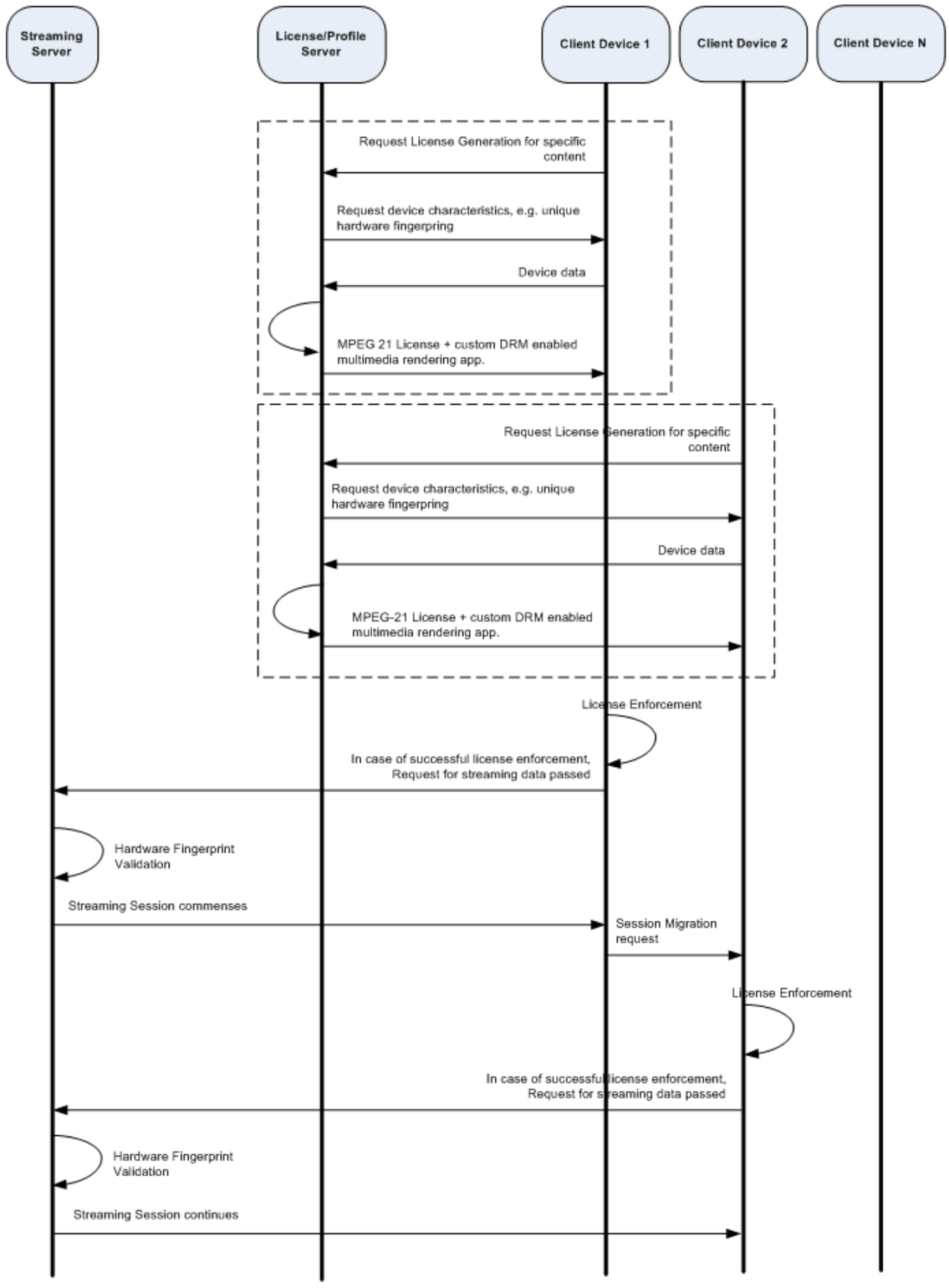

Fig. 26. Session migration scenario between two or more devices.

7. Research initiatives – related work

In this section we present, in short, some recent research initiatives and European projects, which utilized security and DRM aspects, in the context of user-centric multimedia converged nomadic environments. Mainly, we are focusing in Intermedia Network-of-Excellence, (Intermedia, 2006), and CHIRON, (Chiron, 2010).

7.1 Intermedia

There have been considerable efforts to have Audio Video systems and applications converge, in particular in home environments with homes as spaces of convergence, and for nomadic users with advanced mobile devices as points of convergence. These trends are important but also have limitations that we seek to address and overcome: home-centric systems fail to account for increased mobility and the desire to provide continuous service across spatial boundaries outside the home; device-centric convergence, e.g. in 3G phones, supports nomadic use but provides a very limited user experience as no single device and interface will fit many different applications well.

INTERMEDIA project, (Intermedia, 2006), investigated to progress beyond home and device-centric convergence toward truly user-centric convergence of multimedia. Its vision was The User as Multimedia Central: the user as the point at which services (multimedia applications) and the means for interacting with them (devices and interfaces) converge. Key to our vision is that users are provided with a personalized interface and with personalized content independently of the particular set of physical devices they have available for interaction (on the body, or in their environment), and independently of the physical space in which they are situated. Towards this vision is has been investigated a flexible wearable platform that supports dynamic composition of wearable devices, an ad-hoc connection to devices in the environment, a continuous access to multimedia networks, as well as adaptation of content to devices and user context.

The project's main objectives can be summarized to the following, (a) Vision toward device-free user-centric media environments; (b) Constructing multidisciplinary research groups; and (c) Building a common platform with a vision towards semantic convergence.

Towards the context of security and DRM, the following directions have been utilized; (a) Current applications consume a large amount of optimal quality multimedia data suited for a variety of devices, networks, and capabilities. Anywhere and anytime feed the need of content consumers to have the ability to transparently switch the consumption of their session to another (mobile) terminal without any disruption of the session. INTERMEDIA targeted to seamless continuity in multimedia sessions over heterogeneous networked devices through transparent handover of multimedia sessions and accompanied contents. This concept is called session mobility and can be realized using different MPEG-21 technologies for instance; (b) Protection of Intellectual Property (IP) through DRM solutions is a necessity in modern multimedia services. The concerns of content providers for loss of revenues constitute a strong obstacle to the wide deployment of services that involve distribution of IP protected content. Thus, it is a requirement to integrate DRM solutions in services and applications that target delivery of IP protected content to a large base of clients. Considering the technical problems that result from add-on security solutions to independently developed network services, the approach of our efforts to develop architectures with security and DRM as inherent requirements will lead to secure solutions that will increase the trust placed by content providers on the system and thus, it will lead to

wider availability of services to a larger population. Such a delivery architecture needs to be scalable, in order to accommodate an increasing client population, leading to a requirement for optimized use of such resources as bandwidth and computational power, while achieving the target Quality-of-Service (QoS) characteristics for all clients of the service.

7.2 Chiron

The CHIRON Project (Chiron, 2010) intends to combine state-of-the art technologies and innovative solutions into an integrated framework designed for an effective and person-centric health management along the complete care cycle.
In this vision,

- CHIRON will address and harmonize the needs and interests of all the three main beneficiaries of the healthcare process, i.e., the citizens using the services, the medical professionals and the whole community;
- CHIRON will position the citizens at the core of the whole healthcare cycle by considering them as "persons" with specificities and identities and will empower them to manage their own health;
- CHIRON will enlarge the boundaries of healthcare by fostering a seamless integration of clinical setting, at home setting and mobile setting in a concept of a continuum of care;
- CHIRON will speed up the move from treatment of acute episodes to prevention;
- CHIRON will provide the physicians with extensive support for treatment monitoring and management, timely decisions and appropriate actions in both the clinical and home environments;

In CHIRON - rather than aiming at coming up with new security means- our activities on security will advance the SoA by ensuring seamless integration of security features in an extremely heterogeneous environment. So our contribution will be,

- Security architecture featuring:
- Strong security on resource restricted devices such as wireless sensor nodes and within the well equipped hospital infrastructure.
- Seamless integration of devices stemming from different administrative domains e.g. integration of user devices into the hospital architecture and vice versa.
- End to end security in an environment which integrates heterogeneous communications technologies
- Privacy architecture which exploits the security architecture to ensure confidentiality of data and which allows to adapt the level of confidentiality of data according to external conditions such as type of diseases, role e.g. patient versus doctors/nurses.

With regard to privacy whereas many privacy and data protection means aim at a declarative approach only, there are limited approaches for technical protection. To the best of our knowledge there are only a few approaches14 that try to make sensitive data available to a third party while ensuring secrecy of that data.
Main activities of technological innovation will be focused in the following areas:

- Authentication, Authorisation and Accounting (AAA), in order to guarantee the access, the quality of the service and the reliability of an operator.
- Security, in order to guarantee data confidentiality and protect the system through the predictive discovery of attacks.

- Privacy, in order to prevent access or processing of data for purposes other than supposed. Further to control inference of information from the aggregated medical data or even from the mere use of the protocol independent from the actual data content.

7.3 Enabling privacy in person centric e-health environments using DRM aspects

Telemetric monitoring of vital parameters of patients with chronic diseases is recognized to improve their medical condition and hence their quality of life. It also improves treatment adjustments, reaction time in acute cases and helps to reduce duration and costs of hospitalization. As a result of this, there are plenty of products and solutions for personal health monitoring available today that acquire physiological data in real-time. In order for such systems to be widely acceptable and utilized by the medical community and the patients, they must be developed satisfying the security requirements imposed by real-time data communication and protection of sensitive physiological data and measurements, data integrity and confidentiality, and protection of the monitored patient's privacy. By utilizing MPEG-21 standard's primitives, we argue that protection of transmitted medical information and enhancement of patient's privacy is accomplished, since there is selective and controlled access to medical data that sent toward the hospital's servers, (Fragopoulos et al, 2010).

In a person-centric e-health monitoring environment the primary goal is to collect, process, monitor and store medical data from different types of Wearable Embedded Monitoring Devices (WEMDs), which are located on the patient, while furthermore forward those information into general-purpose computing devices – for more sophisticated and complex processing. In such environments different security aspects may arise, so in order to be able to identify the security mechanisms that should be taken in consideration and possibly implemented into the proposed architecture for person-centric e-health monitoring infrastructure, we have to identify and classify, (i) possible attackers and malicious users of the aforementioned environment; (ii) security and privacy requirements. In general, security is a fundamental requirement in modern computing systems, but in user-centic environments focused in e-health, security is a critical and imperative requirement that needs special care in all levels, since in such systems there is flow of sensitive information between various entities.

Although DRM architectures are mainly used to the multimedia world, we argue that utilization of DRM aspects using MPEG-21 standard's primitives, can lead to protection of transmitted medical information and enhancement of patient's privacy, since there is selective and controlled access to physiological data that sent toward the hospital's servers.

A lot of research work (Fragopoulos et al, 2009), (Fragopoulos et al, 2010), (Leister et al, 2009), (Jafari et al, 2010), is towards this direction, while furthermore we are investigating issues related with the safe destruction of the medical data after their viewing; in that context the use of Trusted Platform aspects.

8. Conclusion

In the context of ubiquitous and pervasive computing environments, considerable efforts have been done in order to have audiovisual systems and applications converge, especially in home environments where homes can be considered as spaces of convergence, and for nomadic users with advanced mobile devices as points of convergence. One of the biggest challenges that we have to face in the deployment of architectures in such environments is

related to, on the one hand, with the security and protection of digital contents that interchanged between users of such architectures and on the other hand with the provision to the users with security mechanisms that allow them to perform secure transactions (e.g. authentication, privacy protection, secure data transfer, etc.) in those environments. Moreover intellectual property protection is a mandatory request in modern multimedia environments like the ones that are going to be deployed in the InterMedia context. Today the end-users are equipped with different types of small devices that allow them to be the digital contents creators, thus creating digital content that wish to share with third parties. In most cases, the end-users would like to have mechanisms which would give them the possibility to protect the content which have created and possibly to set their own usage rights over it, thus specifying towards third users how their digital content shall be used.

9. Acknowledgment

The work that has been presented in this book chapter comprises part of extensive research work that has been done to the EU Network of Excellence "INTERMEDIA", FP6 – IST-38419. Also, this work has been funded by CHIRON Project, (Chiron, 2010).

10. Appendix

10.1 MPEG-21 license template

Following, we present a detailed XML view of an MPEG-21 license, generated by the License Server, which allows streaming of a specific content to a specific user on two pre-defined devices.

```
<?xml version="1.0" encoding="utf-8" ?>
<r:license licenseID = "0001"
  xmlns="urn:mpeg:mpeg21:2003:01-REL-R-NS"
  xmlns:mx="urn:mpeg:mpeg21:2003:01-REL-MX-NS"
  xmlns:dsig="http://www.w3.org/2000/09/xmldsig#"
  xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
  xmlns:r="urn:mpeg:mpeg21:2003:01-REL-R-NS"
  xmlns:sx="urn:mpeg:mpeg21:2003:01-REL-SX-NS"
  xmlns:m1x="urn:mpeg:mpeg21:2005:01-REL-M1X-NS"
  xmlns:m2x="urn:mpeg:mpeg21:2006:01-REL-M2X-NS"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

  <!--Grant for protected resource with ID=protectedResource1, for -->
  <!--Target Device with identifierm DEV_A -->
  <r:grant>
   <r:keyHolder>
    <r:info>
     <dsig:KeyValue>
      <dsig:RSAKeyValue>
       <dsig:Modulus>
                        base64_encoded_modulus_of_user's_public_key
                </dsig:Modulus>
       <dsig:Exponent>
                        base64_encoded_exponent_of_user's_public_key
                </dsig:Exponent>
      </dsig:RSAKeyValue>
     </dsig:KeyValue>
    </r:info>
   </r:keyHolder>
```

```
<!--Device's Domain Indentifier-->
<m1x:identityHolder  licensePartId="domain_A">

  <m1x:idSystem>
    urn:mpeg:mpeg21:2006-01-REL-M2X-NS:DM-1000 <!--Domain Manager URI-->
  </m1x:idSystem>
  <m1x:idValue>
    DO0001 <!--Domain Identifier-->
  </m1x:idValue>

</m1x:identityHolder>

<!--Digital Resource Details-->
<!--Content is protected, i.e. encrypted, and encryption key is encrypted-->
<!--and it is embedded into the protectedResource element-->
<m1x:protectedResource licensePartId="protectedResource1">

 <digitalResource>
  <nonSecureIndirect URI="URI_of_digital_content"/>
 </digitalResource>

 <xenc:EncryptedKey>
  <xenc:CipherData>
   <xenc:CipherValue>
   <!--Digital Content, BASE64 encoded, -->
            <!--encrypted key with user's pub. key-->
                    Base64_encrypted_key_with_user_public_key
   </xenc:CipherValue>
  </xenc:CipherData>
 </xenc:EncryptedKey>

</m1x:protectedResource>

<!--Conditions that has to be apply for the Target Device-->
<r:allConditions>

 <m2x:destinationPrincipal> <!--Allowed Target Device-->
  <m1x:identityHolder>
   <m1x:idSystem>
    urn:mpeg:mpeg21:2006-01-REL-M2X-NS:DM-1000:DO-0001
   </m1x:idSystem>

   <!--Target Device A-->
   <m1x:idValue>
    DEV_A, (Target_Device_ID_Unique_HW_Fingerprint)
   </m1x:idValue>
  </m1x:identityHolder>

 </m2x:destinationPrincipal>

 <sx:exerciseLimit>
  <sx:count>Number of Times that content is allowed to rendered</sx:count>
 </sx:exerciseLimit>

</r:allConditions>

</r:grant>

<!--Grant for protected resource with ID=protectedResource1, for -->
<!--Target Device with identifierm DEV_B -->
<r:grant>

 <m1x:identityHolder licensePartIdRef="domain_A"/>
 <m1x:protectedResource licensePartIdRef="protectedResource1"/>
 <allConditions>
```

```
    <!--TARGET DEVICE B-->
    <m2x:destinationPrincipal>

      <m1x:identityHolder>
        <m1x:idSystem>
          urn:mpeg:mpeg21:2006-01-REL-M2X-NS:DM-1000:DO-0001
        </m1x:idSystem>

        <!--Target Device B-->
        <m1x:idValue>
          DEV_B, (Target_Device_ID_Unique_HW_Fingerprint)
        </m1x:idValue>
      </m1x:identityHolder>

    </m2x:destinationPrincipal>

    <!--Maximum number of play times-->
    <sx:exerciseLimit>
      <sx:count>Number of Times that content is allowed to rendered</sx:count>
    </sx:exerciseLimit>
  </allConditions>
</r:grant>

<r:issuer>
  <keyHolder>
    <info>
      <dsig:KeyName>Rights Issuer Public Key Name</dsig:KeyName>
    </info>
  </keyHolder>
</r:issuer>

</r:license>
```

11. References

Adams, Carlisle; (2006) *A Classification of Privacy Techniques*, UOLT Journal, 2006, http://www.uoltj.ca/articles/vol3.1/2006.3.1.uoltj.Adams.35-52.pdf

Anderson, Ross J.; (2001) *Security engineering: a guide to building dependable distributed systems*, Wiley, 2001, ISBN. 0471389226

Burnett, Ian (Ed(s)). 2006. *The MPEG-21 Book,* Willey, ISBN 0470010118, England

CHIRON European Project, (2010), Cyclic and person-centric Health management, JU ARTEMIS Grant Agreement # 2009-1-100228, 2010-2012

Delgado. J.; Prados, J.; Rodríguez, E.; 2005, *An MPEG-21 REL mobile profile*, ISO/IEC JTC 1/SC 29/WG11/M12229, July 2005, Poznan (Poland).

Fragopoulos, A. & Serpanos D. N. (2005). *Intellectual Property Protection Using Embedded Systems*, in Security & Embedded Systems, Vol. 2, IOS Press (Amsterdam, The Netherlands, pp. 44-56, 2005.

Fragopoulos, A.; Serpanos, D. & Voyiatzis, (2009). *Design Issues in Secure Embedded Systems*, book chapter in Embedded Systems Handbook, 2nd ed., ISBN 9781420074109.

Fragopoulos, Anastasios; Gialelis, John; Serpanos, Dimitrios; (2009), *Security Framework for Pervasive Healthcare Architectures Utilizing MPEG-21 IPMP Components*, International Journal of Telemedicine and Applications 2009: 9

Fragopoulos, Anastasios; Gialelis, John; Serpanos, Dimitrios; (2010), *Imposing Holistic Privacy and Data Security on Person Centric eHealth Monitoring Infrastructures*, in 12th International Conference in e-Health Networking, Application & Services IEEE

Gasser, Urs; Palfrey, John, *DRM-protected Music Interoperability and e-Innovation*, November 2007, Berkmann Publication Series, http://cyber.law.harvard.edu/interop/downloads.html

Hankerson, D.R.; Vanstone S. A.; and Menezes, A. J.; (2004), *Guide to Elliptic Curve Cryptography*. New York: Springer, 2004, pp. 311.

Heileman, Gregory L.; Jamkhedkar, Pramod A., (2005), *DRM Interoperability Analysis from the Perspective of a Layered Framework*, in: Proceedings of the Fifth ACM Workshop on Digital Rights Management, 17-26, Alexandria, Nov. 2005, p. 20.

Henk C.; van Tilborg, A. (Eds), *Encyclopedia of cryptography and security*, Springer, 2005, ISBN. 038723473X.

INTERMEDIA Network-of-Excellence, (2006), *Interactive Media with Personal Networked Devices*, http://intermedia.miralab.unige.ch:80/, FP6 – IST- 38419, 2006-2010

International Standards Organization (ISO), (2004), *Information technology -- Multimedia framework (MPEG-21) -- Part 1: Vision, Technologies and Strategy,* ISO/IEC TR 21000-1:2004

International Standards Organization (ISO), (2004), *Information technology -- Multimedia framework (MPEG-21) -- Part 5: Rights Expression Language,* ISO/IEC 21000-5:2004

International Standards Organization (ISO), (2006), *Information technology -- Multimedia framework (MPEG-21) -- Part 4: Intellectual Property Management and Protection Components,* ISO/IEC 21000-4:2006

Jafari, Mohammad ; Safavi-Naini, Reihaneh ; Saunders, Chad ; and Sheppard, Nicholas Paul; (2010) *Using digital rights management for securing data in a medical research environment*, In Proceedings of the tenth annual ACM workshop on Digital rights management (DRM '10). ACM, New York, NY, USA, 55-60

Leister, Wolfgang; Fretland, Truls ; Balasingham, Ilangko; (2009), *Security and Authentication Architecture Using MPEG-21 for Wireless Patient Monitoring Systems*, International Journal in Advances in Security, vol. 2, no. 1, 2009, http://www.iariajournals.org/security/tocv2n1.html

Lipton, R.J.; Rajagopalan, S.; and Serpanos, D.N.; (2002) *Spy: A Method to Secure Clients for Network Services*, in ICDCS Workshops, 2002, pp. 23-28, http://csdl.omputer.org/omp/proeedngs/dsw/2002/1588/00/15880023abs.htm

Menezes, A.J.; van Oorschot, P. C.; and Vanstone, S. A.; (1996), *Handbook of Applied Cryptography*, CRC Press Inc., 1996

Messerges, T.S.; Dabbish, E.A.; (2003), *Digital rights management in a 3G mobile phone and beyond*, in Proceedings of the 2003 ACM workshop on Digital rights management, 2003, pp. 27-38.

Repetto, Matteo; Rapuzzi, Riccardo; Chessa, Stefano; Lenzi, Stefano; Gialelis, John and Fragopoulos, Tasos; (2010), *The InterMedia Networking and Security Architecture for User Centric Multimedia Convergence*, International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS), 2010, Ottawa, Canada.

Roush, W.; (2006), Inside the Spyware Scandal, MIT Technology Review, May-June 2006

Serpanos, D.N.; and Lipton, R.J.; (2001), *Defense Against Man-in-the-Middle Attack in Client-Server Systems*, in ISCC, 2001, pp. 9-14, http://csdl.omputer.org/omp/proceedings/s/2001/1177/00/11770009abs.htm

Schneier, B. ; (1996), *Applied Cryptography*, (Second Edition), John Wiley & Sons, 1996, ISBN. 0-471-11709-9

Cutting Edge Research in New Technologies

Edited by Prof. Constantin Volosencu

ISBN 978-953-51-0463-6
Hard cover, 346 pages
Publisher InTech
Published online 05, April, 2012
Published in print edition April, 2012

The book "Cutting Edge Research in New Technologies" presents the contributions of some researchers in modern fields of technology, serving as a valuable tool for scientists, researchers, graduate students and professionals. The focus is on several aspects of designing and manufacturing, examining complex technical products and some aspects of the development and use of industrial and service automation. The book covered some topics as it follows: manufacturing, machining, textile industry, CAD/CAM/CAE systems, electronic circuits, control and automation, electric drives, artificial intelligence, fuzzy logic, vision systems, neural networks, intelligent systems, wireless sensor networks, environmental technology, logistic services, transportation, intelligent security, multimedia, modeling, simulation, video techniques, water plant technology, globalization and technology. This collection of articles offers information which responds to the general goal of technology - how to develop manufacturing systems, methods, algorithms, how to use devices, equipments, machines or tools in order to increase the quality of the products, the human comfort or security.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Anastasios Fragopoulos, John Gialelis and Dimitrios Serpanos (2012). DRM & Security Enabling Mechanisms Leveraging User Centric Multimedia Convergence, Cutting Edge Research in New Technologies, Prof. Constantin Volosencu (Ed.), ISBN: 978-953-51-0463-6, InTech, Available from: http://www.intechopen.com/books/cutting-edge-research-in-new-technologies/drm-security-enabling-mechanisms-leveraging-user-centric-multimedia-convergence

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen