Scientia Iranica (2019) 26(1), 103–113

Sharif University of Technology
Scientia Iranica
Special Issue on: Socio-Cognitive Engineering
http://scientiairanica.sharif.edu

EDU-DRM: A Digital Rights Management (DRM) system for K-12 education

A. Özmen[a,*], A. Şanslı[a], and V. Harun Şahin[b]

a. *Department of Computer Engineering, Sakarya University, Serdivan, 54187, Sakarya, Turkey.*
b. *Department of Software Engineering, Sakarya University, Serdivan, 54187, Sakarya, Turkey.*

Received 10 October 2017; received in revised form 18 December 2017; accepted 3 March 2018

KEYWORDS
Digital Rights Management (DRM); Intellectual property; K-12 education; Digital publishing; Secure content distribution.

Abstract. The technological achievements in digital publishing have made paperless education possible even in K-12 education. Aside from high bandwidth distribution infrastructure, the main difficulties of digital publishing are preserving personal information and protecting the rights of copyrighted contents. Although specially designed Digital Rights Management (DRM) systems can be used to control distribution and usage of private and/or copyrighted contents in K-12 education, dealing with a large number of bursty concurrent access requests and changing the access rights of a large number of students from one content class to another at the end of each education period make the problem different from existing ones. This paper introduces a new DRM system, called EDU-DRM, which includes a novel bit based authorization approach to reduce the processing time for authorization requests and automatize the access right adjustments with predefined rules for K-12 education. During the study, an experimental framework is designed using Apache Bench to analyze the proposed approach and evaluate it. The system is compared with XML based authorization approach and the results are presented in the paper.

1. Introduction

Computer and communication technologies have changed dramatically in the last two decades. The form, functionality, and connectivity of the computer systems have evolved, and new products have emerged in the market, such as smart phones, wearable computers, etc. These new computer technologies have changed our view of information by means of creation, access, storage, and usage of digital data. Now, any kind of teaching material, such as books, pictures, videos, or animations, can be stored in digital format. Also, the distribution of the digital content has changed, which is called digital publishing.

While achievements in the technology eases sharing digital data, it creates some new difficulties such as preserving the privacy and protecting the copyright. Especially book authors as well as music and film producers are suffering from illegal replications of their works. During the last decade, many systems have been developed to overcome these problems; they are named Digital Rights Management (DRM) systems. The main aims of a DRM system are to store private or copyrighted material securely and control the distribution or usage. In this view, a DRM system can be used to manage any kind of valuable digital content from static data, such as medical records, up to dynamic stream data, like digital TV/Video broadcast.

In addition to the above examples, privacy and copyright rules are important for the digital materials used in education. Nowadays, the systems that protect the intellectual property of educational publishers are becoming quite valuable, since technological advances

*. *Corresponding author. Tel.: +90 264 2956984*
E-mail address: ozmen@sakarya.edu.tr (A. Özmen)

doi: 10.24200/sci.2018.5345.1219

force publishers to convert the way of publishing to digital. As an example, a project called *Movement of Enhancing Opportunities and Improving Technology*, known as FATIH, was started for K-12 education in Turkey by the government at the end of 2011. One of the most important components of this project is a DRM system to provide and manage digital educational materials to the end-users without violating the intellectual property of providers. Developing such a DRM system, which protects digital rights of the contents, becomes a challenging problem, because the number of end-users is almost 18 million students in Turkey.

In this study, we have developed a new DRM system that includes new approaches to authorization module and client application. The new system is called EDU-DRM, developed for K-12 education, and it is planned to serve millions of students and teachers. In education, the characteristics of requests differ from existing DRM systems. Bursty and concurrent requests create performance bottleneck for such systems that causes long delays [1]. The new approach changed the way of accessing the contents by implementing a logic based access control instead of REL. For example, once a request for a content arrives, EDU-DRM scans user profiles and automatically resolves access rights through authorization process. EDU-DRM encrypts the content in the first step during the upload time and preserves the encrypted form in the system throughout the life cycle, until it is consumed by a specifically developed plug-in reader for web browsers.

2. Related work

2.1. Existing DRM systems

In the literature, there are some commercial and academic DRM systems, some of which are in use and the others have been developed only as research projects. One of the academic studies is "Open and Secure DRM" (OpenSDRM) project. In this project, SSL/TLS connection is preferred to provide secure transmission among DRM components [2,3]. OpenSDRM is known by its open source structure and interoperability, and the contents are encrypted with MPEG-4 standards [4]. Multimedia Information Protection And Management System (MIPAMS) is another important work, which has been developed by Distributed Multimedia Applications Group (DMAG) as a European funded research project [5]. In this system, the contents are encrypted with MPEG-21 standard, and the system is implemented with web services in a modular fashion [6]. AXMEDIS, a European research project, is a Java based DRM system, which was developed for a variety of end-user devices [7]. It was not developed to be open source, but has a lot of open specifications that provide libraries to create independent usage scenarios. Coral is an integrator framework for DRM systems, which mainly focuses on interoperability [8]. This is an important work for unifying different DRM systems, and has been developed by a consortium.

The commercial DRM systems usually hide internal details due to security reasons. One of the commercial DRM examples is Open Mobile Alliance DRM (OMA-DRM) which mainly focuses on standardization process of DRM systems on cellular phones [9]. OMA-DRM uses a specially designed Rights Expression Language (REL), which differs from OpenSDRM and MIPAMS that use MPEG-21 based REL. In Microsoft Windows Media Rights Manager (WMRM) system, audio and video contents are transferred in compressed format [10,11]. However, WMRM works only on Microsoft based operating systems. It has specially designed Audio Stream Format (ASF), Windows Media Audio (WMA), and Windows Media Video (WMV) formats. Adobe PDF Merchant is a specifically designed DRM system for Portable Document Format (PDF) documents [12]. It reaches contents by means of a plug-in in Acrobat Reader. Since it is not supported by some operating systems, such as Unix, and due to some security concerns, Adobe replaced this system with Adobe Digital Experience Protection Technology in 2012. A personal mobile DRM system has been developed by Bhatt et al. for Motorola smart phones [13].

Besides these DRM systems, there are studies on security schemes of DRM systems in the literature. For example, Das et al. [14], Chang et al. [15], Chen [16], and Chang et al. [17] presented secure Enterprise Digital Rights Management (E-DRM) schemes. In some other studies, security issues are studied for peer-to-peer music distribution on mobile DRM devices [18,19]. In this study, a fingerprint for every user is produced using the RSA algorithm, and then it is embedded within the music file in a protected form to identify any unauthorized distribution.

2.2. DRM system architecture

The general architecture of DRM systems is client-server model, because demand distribution is homogeneous and the payload size of the contents is not very large. Once the number of end-users or the content size increases, or content type becomes time-critical, performance bottleneck problem may occur in the client-server architecture. As this study aims to protect the copyrighted educational materials such as books, the DRM systems are examined from the viewpoints of bulky volume contents and user interaction platform.

DRM systems have many uses, from protecting privacy of people to protecting copyrighted materials. However, every DRM system has some specifically designed software modules such as authentication, au-

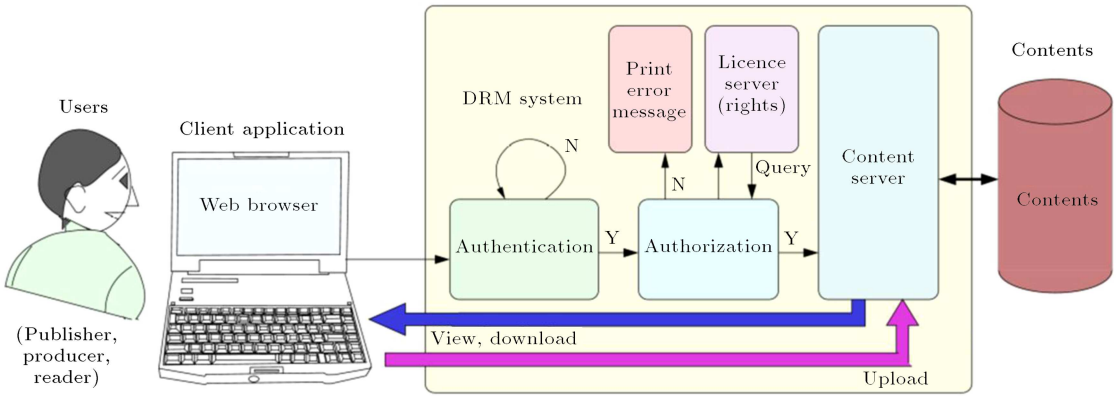

Figure 1. Basic components of a DRM system.

thorization, license module, etc. (see Figure 1). These modules serve the user of the DRM system. Three main user groups of a DRM system are producers, readers, and publishers. A producer is the owner of a content, and is responsible for creating and uploading it. From the viewpoint of a producer, a DRM system must provide authentication, authorization, and the uploading functionalities. A reader, on the other hand, is a person who consumes (views, downloads, prints, etc.) the contents. Sometimes a reader can be a guest who can only access the free materials and preview meta-data information of copyrighted materials. On the contrary, a normal reader is an authorized user who has some kind of authorization on the copyrighted materials, like viewing, downloading, printing, etc. A producer is authorized to set/edit the rights (license rules) of uploaded contents and users, so they need management functionalities.

Regarding the software, there are three main building blocks of a DRM system: content server, license server, and client application. A basic DRM system, which includes these modules, is shown in Figure 1.

The *client application* can be a web browser, a media player, an image viewer, or any other application developed for specific needs. For example, a client application of a publisher is responsible for managing the whole users of the system. The client application of the producer is responsible for the upload operation of the content and its meta-data to the content server. Then, the producer sets up the required rights regarding the uploaded content. General responsibilities of the reader client application are authenticating the reader, browsing the content meta-data, downloading the content in a secure way when needed, and rendering the content.

In general, the *content server* is responsible for managing the content. Content management includes a wide range of operations from uploading and downloading the content and its meta-data to encrypting it and checking the rights of the content by the help of the license server.

The *license server* is responsible for managing and storing the rights of the content and the licenses purchased by users. In conjunction with the content server, the license server helps to store the rights on the content for each user of the system. It also checks the rights on the content for a particular user whenever the user requests that content. The license server fulfills the management of rights by using special languages called Rights Expression Language (REL). REL provides a flexible and inter-operable means for accessing digital contents. One of the well-known RELs is Open Digital Rights Language (ODRL) [20]. It is XML based, and has been developed by W3C ODRL Community Group. The other well-known REL is MPEG-21, which has been developed by Moving Picture Experts Group (MPEG) [21].

Although these are the main building blocks of any DRM system, implementation of any real-life DRM system generally requires the splitting of these blocks to smaller modules to enable the creation of a more modular, flexible, and manageable structure. Creating a modular system also enables the easy integration of the standards such as MPEG-21, therefore making it more interoperable. For this purpose, many DRM systems developed so far include more components such as authentication and authorization. These blocks are responsible for authenticating the users and authorizing them to use the permitted services and contents of the system.

For instance, while a producer is authorized to upload contents to the system, a reader can only be authorized to download specific contents from the system. Besides this, the publisher can also authorize a user to manage a content (read, print, etc.) depending on the authority of the user on the particular content which is stored by the license server. On some systems (e.g., MIPAMS [5]), authentication and authorization services can be divided into two components.

3. EDU-DRM: A DRM system for K-12 education

Technological advances in digital publishing and front-end devices affect educational materials and methods in K-12 education. The materials for K-12 education in this paper are going to diminish gradually in near future and equivalent digital materials will replace them, and privacy and copyright rules become the most important issues for these materials. Some countries have started taking initial steps for this era. For example, a project called *Movement of Enhancing Opportunities and Improving Technology*, known as FATIH, was started by the government at the end of 2011, which is among the most significant educational investments of Turkey. FATIH Project proposes that "Smart Class" project be put into practice in all schools around Turkey. With this project, thousands of schools and classes will be equipped with the latest information technologies and will be transformed into computerized classes (Smart Classes) [22]. In this scope, it is planned to distribute tablet PCs to all primary and secondary education (K-12) students besides interactive smart boards in classrooms.

In terms of FATIH project, digital copies of educational materials such as books and periodicals are distributed to authorized users like students and teachers free of charge in Turkey. The digital contents are prepared by many authors for all grades and copyright costs are paid by the government (Ministry of National Education). FATIH project requires a specific DRM system to distribute the digital contents. The Turkish government have made several project calls to overcome the problem. A novel DRM system that achieves the task must consider the following properties:

1. **Users**: Students, teachers, producers, guests, and administrators are defined as user groups of the system. The student group is the most populated user group, and expected to include more than 18 million pupils. The teachers form the second largest group. Guests are a dynamic group and their population may vary, but it is expected to be smaller than the size of teachers group. A group of officers are assigned as administrators who are responsible for editing meta-data of contents and users. Authentication information of the users is provided by Ministry of National Education;

2. **Contents**: Contents are split into 12 different classes based on K-12 curricula and a pool for newly uploaded but not approved documents. Producers must also enter necessary meta-data for their newly uploaded contents. A new content is first uploaded to the pool, and after approval, it is moved to the respective grade class. Unapproved contents cannot be reached by the students and the teachers;

3. **Rule based authorization**: Since the status of almost all students changes annually, a rule based authorization must be implemented in the DRM system to adapt the access rights respectively;

4. **Users and contents relation**: Students and teachers must be allowed to reach only the contents related to their own grade. Students' grade records and their allowed target contents must change automatically every year. However, the system must easily allow a teacher to access the contents that reside with different grades.

Based on the above specifications, a new DRM system is designed. All possible "access rights" are determined and put into a table (see Table 1). Then, different roles are defined for the users of the system, shown in Table 2. A bit-wise logic based encoding approach is used to implement REL for such complex, flexible management system by consuming the minimum database disk space.

Table 1. Specified access types for EDU-DRM. They cover all kinds of requests, which may come from standard users or administrators of the system.

Access ID	Access types
1	Edit records
2	Monitor
3	Download
4	Upload
5	View
6	Erase
7	Print

Table 2. The roles and respective authorization codes defined in the system. The authorization codes are encoded using binary logic for the access types given in Table 1.

Role ID	Role name	Authorization code						
		1	2	3	4	5	6	7
1	Super admin	1	1	1	1	1	1	1
2	User admin	1	1	0	0	0	0	0
3	Grade admin	1	1	0	0	0	0	0
4	License admin	1	1	0	0	0	0	0
5	Teacher	0	1	1	0	1	0	0
6	Student	0	0	0	0	1	0	0
7	Producer	0	0	0	1	1	0	1
8	Guest	0	0	0	0	1	0	0

Table 3. Some examples of content record. A special access vector is assigned to each record. The access vectors are created using the roles in the system, and they are kept in the license table in the server.

Content ID	Access vector							
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
234	1	0	0	0	1	1	0	0
34	1	1	0	0	1	1	0	0
245	1	0	0	0	0	0	0	1
171	1	0	0	1	0	0	1	0

3.1. EDU-DRM access types

Table 1 shows seven different access types in the system. These access types are assigned to the roles given in Table 2. Some examples of content records stored in license table are shown with their respective access vectors in Table 3. EDU-DRM access types are explained in detail as follows:

- **Edit records**: This access type permits only the admins to modify the profile data in the system. Further access limits are imposed on business logic of the system. For example, a user with the user admin role is not allowed to modify the content profile data; likewise, a user with the content admin role cannot modify the user data. The roles with these access types are expected to fix exceptions caused by generalization of rule based access distribution;
- **Monitor**: This access type is specifically designed to generate different reports from the log tables of the system. These log tables are filled by the activities such as editing meta-data or content browsing. For example, a teacher with this access type can create a report that shows "how many times each student in a class accessed a specific content";
- **Download**: This access type allows downloading an encrypted content by a device. The downloaded content is stored into disk in encrypted format. This feature is not utilized in this version of EDU-DRM, planned for future use;
- **Upload**: Uploading is designed for loading the contents from remote sites to the system. Usually uploading is granted to producers;
- **View**: The access type "view" is created for buffered viewing of a content data without saving it into the disk. For example, a web browser gets the content data as a stream in this type, decrypts it in the buffer, and displays it. Once viewing action ends, no digital data from the content remains in the client machine. A special code has been developed for the server side to block client side caching;
- **Erase**: This access type is provided for deleting contents from the server disk;
- **Print**: This access type enables/disables printing hard copy of the viewed contents.

3.2. Roles

Users of the system are divided into groups to be treated differently when they interact with the system. Eight distinct roles are created in the system level to simplify management of assigning the access rights to the user groups (Table 2). These roles are not same as the database-level roles provided by any database system such as Oracle. Assigning an access type to a role is accomplished by using bit oriented vector. The following list explains the roles and their default access rights:

- **Super admin**: This role is assigned to a group of people with very few members. These users have full access to anything (contents, tables, etc.) in the system;
- **User admin**: This group can edit only standard user profiles. Every region in the country may have one or more user admins, and they change a user profile when necessary;
- **Grade admin**: This group can edit only user grade cells in the profile tables. User admin and grade admin roles are expected to resolve exceptions which may occur at the beginning of a semester. Both of these roles, for example, can be assigned to officers at schools to fix such problems caused by generalization of the rule based management. Figure 2 shows a user record structure designed for EDU-DRM;
- **License admin**: License admin is a role that can be assigned to a group of users formed by experts. They have the power to accept or decline draft contents in the pool. Accepted contents are published by moving them to the respective grade class by resetting the pool bit in the record (see Figure 3);
- **Teacher**: A teacher may access multiple grade contents and can create a report from log data. For example, a teacher can create various reports that show access frequency of the students to a specific content for a period of time;
- **Student**: Different from teachers, students can access contents only for viewing and they can only

	R1	R2	R3	R4	R5	R6	R7	R8	K1	K2	K3	K4	K5	K6	K7	K8	K9	K10	K11	K12
Standard content profile data	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0

Figure 2. A user profile record structure created in EDU-DRM system for a student user in grade K-1.

	K1	K2	K3	K4	K5	K6	K7	K8	K9	K10	K11	K12	PL	PB	T1	T2
Standard content profile data	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0

(a)

	K1	K2	K3	K4	K5	K6	K7	K8	K9	K10	K11	K12	PL	PB	T1	T2
Standard content profile data	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

(b)

Figure 3. Content record structure in the system: (a) A draft of K-11 content in the pool and (b) the reviewed and licensed content by an authorized user through simply resetting the pool bit (PL) in the record; it is now accessible by grade K-11 classes.

reach their grade contents; downloading is not allowed;

- **Producer**: A user can be a producer by role bit **R7** in the user profile record (see Figure 2). Producers can upload draft contents only to the pool. The encryption keys are created at the beginning of an uploading process; then, the content is immediately encrypted and stored in the system disk. A producer can also view and print own documents uploaded to the system;
- **Guest**: Guest users do not have passwords. If a user wants to log in as "guest", they must enter "guest" as user ID. The system immediately lets that user log in to the system without asking a valid password. Then, the system automatically creates a user record for that user with empty profile data; only **R8** bit is set. A guest user can only reach public documents in the system. Public documents are marked by setting public bit (see **PB** in Figure 3).

3.3. Licensing

Licenses are usually distributed dynamically in general DRM systems, whereas in EDU-DRM, licensing process is a static task. Assigning a student user to a class authorizes them to access a sub-class of contents. At another point, assigning a content to a grade makes it available to the respective classes. Student users' authorization does not change throughout the academic year, so they can reach their contents. In the next year, students' grades automatically change and their authorized sub classes also change. Similarly, a license of a content is set after evaluation process ends and the content is accepted. The content's grade does not change at all.

This is totally different form, e.g., iTunes by Apple. A user has to pay for licensing a movie in iTunes. Limited number of different devices registered to that account are authorized to download the content as many times as the user wishes. In another way, a user can rent a movie; in this case, the license ends after a period of time. That means the authorization to access that content ends.

In EDU-DRM, licensing is managed by the license admins. A license admin should be an expert in the area, because they evaluate the internals of a content only uploaded by a producer. Hence, a license admin can reject a content if the content is not suitable. If the content is appropriate, then licensing of a content is quite easy and accomplished by resetting only the respective pool bit (**PL**) of the content profile data. After giving license to a content, it becomes available to all authorized users.

The system automatically creates log data for all of these activities and the logs can be traced by the super admin. The system provides software tools to examine log data and creates various reports on the content and user activities.

3.4. Modules of the EDU-DRM system

The EDU-DRM has been developed in modular fashion. The system consists of several servers of which the responsibilities are defined clearly at the beginning. These servers are authentication server, license server, content server, and monitoring server, which are implemented as separate units in the system.

- **Authentication server**: This server controls the login when a user wants to retrieve a content in the list. A profile data must be created for a user to pass this step. All user profile data are stored at another site by the government. If a user does not pass the authentication, they can log in to the system as a guest and retrieve only public contents;
- **License server (authorization server)**: An authenticated user must also pass the license server to see if the user is authorized for the requested action. The requests coming from authorized users are fulfilled immediately; otherwise, a negative answer is turned back. Figure 4 depicts the activities coordinated by the license server;
- **Content server**: The content server is responsible for the following items:
  1. Creating a new record with a unique ID for newly uploaded content based on MPEG-21 Digital Item Declaration (DID) standard [23];

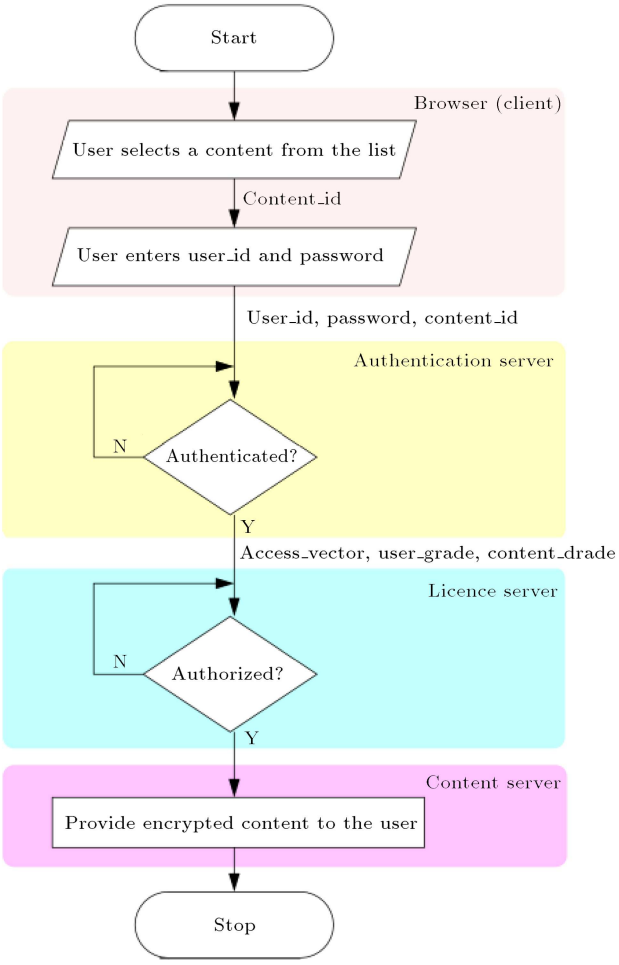

Figure 4. Flow diagram of a content requested by a user.

2. Encrypting and decrypting a content whenever requested;
3. Managing and storing encryption keys for the contents and supplying them when needed;

- **Monitoring server**: A monitoring server generates different kinds of reports and logs of the system, like authentication information, user activities, content information, etc. It also creates an activity list of users of specific grade contents. For example, a teacher can create a report that shows how many times each student accessed some specific contents;
- **Encryption**: The EDU-DRM system creates a pair of 4096-bit public/private keys with RSA algorithm at the initial setup phase. The private key is stored in the disk and only the root can access it. Then, a 256-bit symmetric key is generated by random number generator to encrypt the new content. The symmetric key is encrypted using asymmetric keys for security and stored into the database table. Before encryption, a 160-bit summary information is taken by "SHA1" algorithm from the original content to prove originality and integrity. This 160-bit characteristic data is also stored into the database;
- **Upload**: After passing authentication and authorization steps, a producer selects the upload option from the default menu configured for this group of users. The producer must enter definitive data for the new content and then press the upload button from the user interface shown in Figure 5. The metadata is formed in MPEG-21 DID standard, and saved into the database [23]. There is no constraint on file format for uploading; however, currently, the viewer (web browser) supports only PDF files. A content is encrypted on the fly during uploading, and the encrypted form of the content is saved into the disk;
- **View**: After passing authentication and authorization steps, a root process in the server fetches the encrypted key from the database. The symmetric key is resolved and sent along with the content and some Java-Script control script to the client side. Sending the content and the key to the client is accomplished by HTTPS protocol. The Java-Script disables some control buttons on client browser and decrypts the content. Copying and printing of the content at the client side are prevented in this way. Resolved content key is never stored into the disk on the client side and once the session ends, the key is deleted.

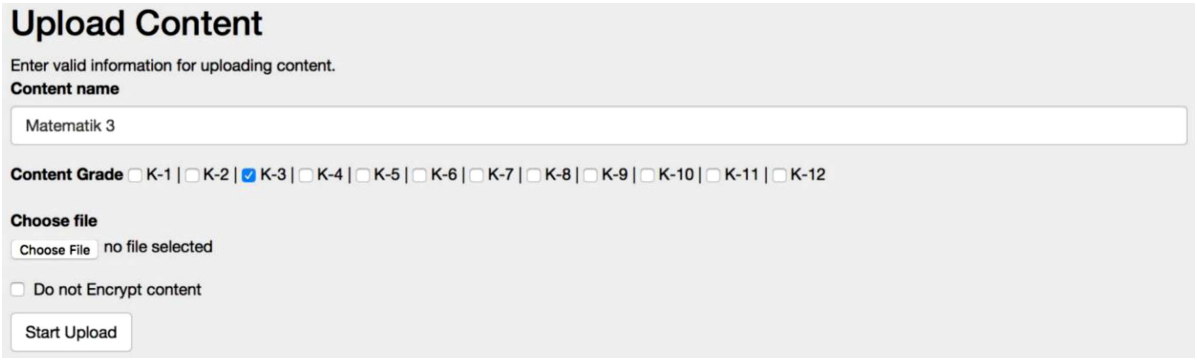

Figure 5. A screenshot of content uploading process. The producer must select a proper grade for the content before uploading.

4. Experimental

A fully functional prototype EDU-DRM system was implemented in Sakarya University and installed on the test servers for performance analysis. An alternative XML based authorization module was also implemented for EDU-DRM as a web service in the local domain. The XML file format was taken from the study by Polo at al. [24]. The size of the XML files used in the tests was approximately 1 KB or, usually, much larger depending on interoperability conditions.

Apache Benchmark was used for creating bursty requests and timing the events during the experiments [25].

Figure 6 shows the test bed used in the performance comparison studies. The web server, the license server, and the database server are created from a physical blade server system using a VMware, namely, vSphere v6.0 virtualization software. The blades inside the chassis contain the following blade boards:

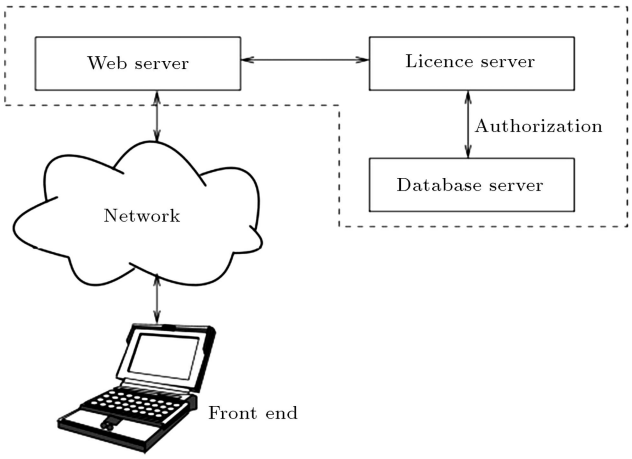

Figure 6. The test bed block diagram used for bit based and XML based authorization comparative studies.

- Blade 1: 6 Intel Xeon E5-2630 CPU v4 2.20 GHz; 10 cores.
- Blade 2: 2 Intel Xeon E5-2640 CPU v4 2.40 GHz; 10 cores.
- Blade 3: 2 Intel Xeon E5-2650 CPU v4 2.40 GHz; 10 cores.
- Blade 4: 1 Intel Xeon X5650 CPU v4 2.66 GHz; 6 cores.

The chassis contains 2.47 TB global memory of RAM shared by all blades. The computer is attached to a storage system (Hitachi HUS130 Storage) with a total of 100.78 TB disk space obtained by SSD, SAS, NL-SAS, and SATA technologies. The chassis has 10 Gbps bandwidth in its backbone, and the local network infrastructure from servers to main switches is made of fiber optical cables with 40 Gbps bandwidth.

Figure 7 shows the events between the timestamps that happen during both types of authorization. As it can be seen in the figure, XML based authorization includes more events to satisfy interoperability.

Graphs in Figure 8 show performance results of various tests conducted with 100, 1000, 5000, and 10000 concurrent requests. The request tests are continued up to 10000 requests, and the response time of each request is measured in milliseconds.

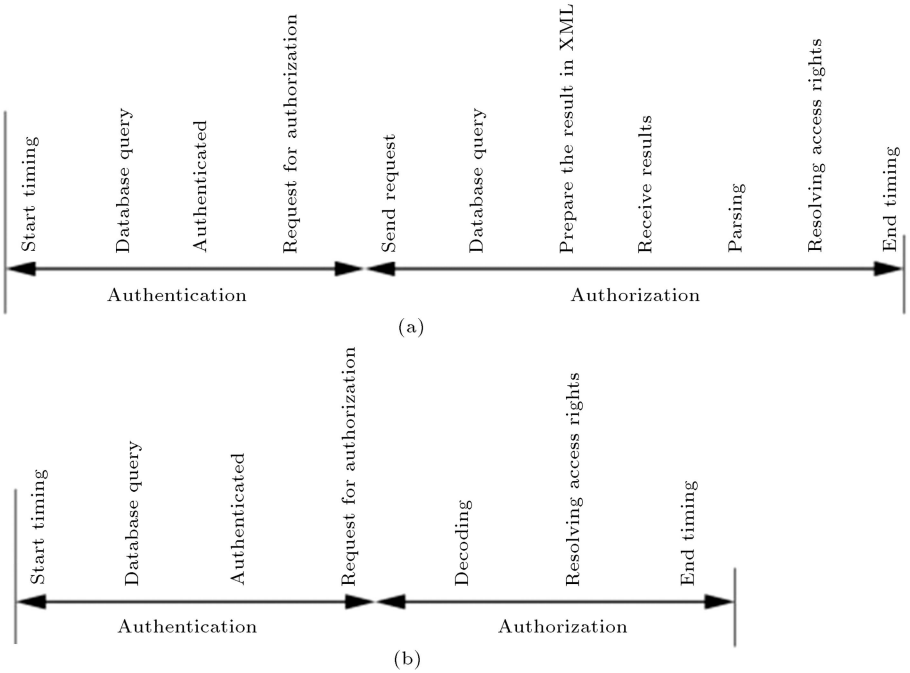

Figure 7. Authentication and authorization events: (a) XML-based and (b) bit-based licensing.

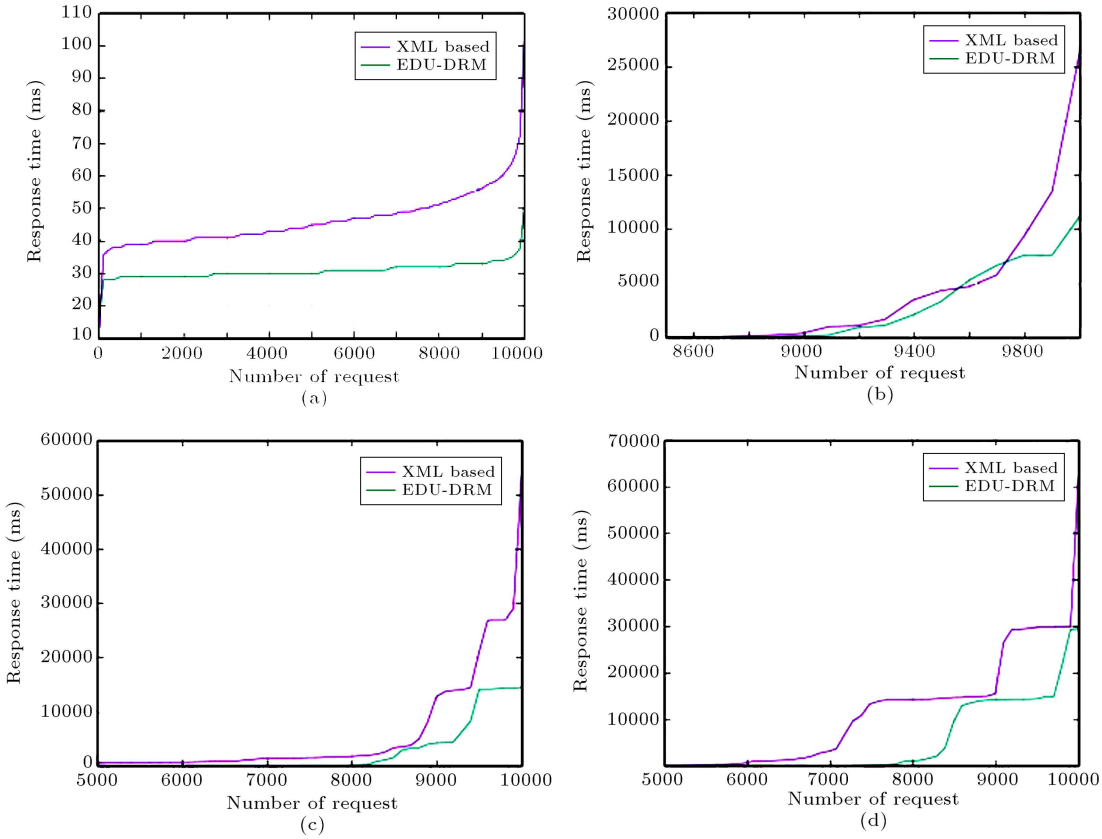

Figure 8. Comparative performance results are shown in milliseconds between EDU-DRM and XML based authorizations for a total of 10000 requests, which are sent in concurrent bundles: (a) 100 concurrent requests in a bundle, (b) 1000 concurrent requests in a bundle, (c) 5000 concurrent requests in a bundle, and (d) 10000 concurrent requests in a bundle.

5. Results and discussions

In the first part of the experiments, bundles of 100 concurrent requests are sent to the web server, and the response times are measured (see Figure 8(a)). Even though 100 concurrent requests are far less than the expected load, bit-based approach leads to better results than XML based approach does. As it can be seen in the figure, the delay difference is about 10 ms and it steadily increases up to 9000 requests. It becomes exponential for both approaches at the end of the experiment with a delay difference of 50 ms.

In the second part of the experiments, bundles of 1000 concurrent requests are sent to the system. As it can be seen in Figure 8(b), after 9700 requests, the delay time dramatically increases in XML based approach. When 10000 requests are reached, the delay difference between the two methods is about 15 seconds. After this point, each individual authorization request waits for about 26 seconds in the XML-based approach, while in the bit-based approach, the time is about 11 seconds.

In the third part of the experiments, bundles of 5000 concurrent requests are sent to the web server (see Figure 8(c)). This time, the difference starts from 5000 requests and increases steadily until the end of the experiment. The final delay difference is about 35 seconds, which means that every individual request that arrives after this point will wait at least 35 seconds for authorization process.

In the last part of the experiment, 10000 requests are sent all at once and the processing delays are observed (see Figure 8(d)). The XML based authorization causes significant delays after 7000 requests are reached, and a similar pattern is repeated in bit-based approach after 8500 requests are processed. Interestingly, the delay stays horizontal for some time and then, a sudden increase occurs when physical limits such as network bandwidth or data retrieval bandwidth in the database server are saturated.

As a result, bit-based authorization approach makes EDU-DRM system more robust against bursty demands. Since XML-based approaches take interoperability more seriously, the time taken to process the detailed XML description files includes more delays in response to sudden requests. The number of clients is very large (18 million students), and the access pattern is quite bursty for DRM system in K-12 education; hence, reducing the license processing time becomes more critical than interoperability of the system.

The system also has a rule based authorization adjustment module that automatically assigns the new licenses to the students at the end of each educational season. This approach significantly reduces the workload of license distribution that must be done at the end of each year. There is no license selling mechanism in the system as in classical DRMs.

The main disadvantage of bit-based role definition is that the encoded roles reduce interoperability with other DRM systems. However, smaller message size and shorter processing time significantly reduce the traffic in the network and possibility of a contention in the central server. Since EDU-DRM is only developed for education, interoperability has not been considered in the architecture.

6. Conclusions

The main problems of educational DRM systems are the delays that are incurred in response to bursty requests coming from the students and creating/managing license conditions for millions of end-users every year. Hence, the licensing module of the educational DRM systems must be different.

In this study, a DRM system architecture, called EDU-DRM, was developed, specifically for K-12 schools, to manage proprietary digital material at educational institutions from primary schools up to colleges. In the system, a novel bit-based role and access right definition schemes were developed instead of a Right Expression Language (REL) to process the requests within the lowest amount of time. A rule based authorization adjustment module was also developed in the system that automatically assigned the new licenses to the students at the end of each educational season. EDU-DRM was developed in a platform-independent manner and neither an operating system nor a specific client application was required. Only a web browser was sufficient to retrieve the private contents. Since all the interfaces were developed using web based techniques, the system could easily be managed independently from any location. Many comparative experiments were conducted to test the system authorization module performance for bursty requests, and promising results were obtained.

As a future work, we are planning to distribute the central content server dynamically to locations closer to the end-users based on the request frequency. It is expected that this enhancement will improve the system response time further by hiding network latencies.

References

1. Amoozegar, M. and Nezamabadi-pour, H. "A multi-objective approach to model-driven performance bottlenecks mitigation", *Scientia Iranica*, **22**(3), pp. 1018-1030 (2015).
2. Serrao, C. "Open secure infrastructure to control user access to multimedia content", *Proceedings of the Fourth International Conference on Web Delivering of Music*, Barcelona, Spain, pp. 62-69 (2004).
3. Delgado, J., Dias, M. S., and Serrão, C. "Using web-services to manage and control access to multimedia content", *Proceedings of The International Symposium on Web Services and Applications*, Las Vegas, Nevada, USA, pp. 23-28 (2005).
4. Koenen, R. "Intellectual property management and protection in mpeg standards", *Workshop on Digital Rights Management for the Web*, Sophia Antipolis, France, pp. 22-23 (2001).
5. Delgado, J., Torres, V., Llorente, S., and Rodríguez, E. "Rights management in architectures for distributed multimedia content applications", *Trustworthy Internet*, pp. 335-347 (2011).
6. Karpouzis, K., Maglogiannis, I., Papaioannou, E., Vergados, D., and Rouskas, A. "Mpeg-21 digital items to support integration of heterogeneous multimedia content", *Computer Communications*, **30**(3), pp. 592-607 (2007).
7. Ng, K., Ong, B., Neagle, R., Ebinger, P., Schmucker, M., Bruno, I., and Nesi, P. "Axmedis framework for programme and publication and on-demand production", *Proceedings of the First International Conference on Automated Production of CrossMedia Content for Multi-Channel Distribution*, Washington, DC, USA, pp. 247-250 (2005).
8. Kalker, T., Carey, K., Lacy, J., and Rosner, M. "The coral DRM interoperability framework", *Consumer Communications and Networking Conference*, Las Vegas, NV, USA, pp. 930-934 (2007).
9. Irwin, J. "Digital rights management: The open mobile alliance DRM specifications", *Information Security Technical Report*, **9**(4), pp. 22-31 (2004).
10. Austerberry, D., *Digital Asset Management*, Taylor & Francis, Oxford, UK 2nd Edn., pp. 283-306 (2012).
11. Becker, E., Buhse, W., Günnewig, D., and Rump, N. "Digital rights management: Technological, economic, legal and political aspects", *Lecture Notes in Computer Science*, 1st Edn., Springer, Berlin Heidelberg, Germany (2003).
12. Michiels, S., Joosen, W., Truyen, E., and Verslype, K., *Digital Rights Management - A Survey of Existing Technologies*, Report CW 428, K. U. Leuven, Department of Computer Science (2005).
13. Bhatt, S., Sion, R., and Carbunar, B. "A personal mobile DRM manager for smartphones", *Computers & Security*, **28**(6), pp. 327-340 (2009).
14. Das, A.K., Mishra, D., and Mukhopadhyay, S. "An anonymous and secure biometric-based enterprise digital rights management system for mobile environment", *Security and Communication Networks*, **8**(18), pp. 3383-3404 (2015).

15. Chang, C.C., Chang, S.C., and Yang, J.H. "A practical secure and efficient enterprise digital rights management mechanism suitable for mobile environment", *Security and Communication Networks*, **6**(8), pp. 972-984 (2013).

16. Chen, C.L. "A secure and traceable e-drm system based on mobile device", *Expert Systems with Applications*, **35**(3), pp. 878-886 (2008).

17. Chang, C.C., Yang, J.H., and Wang, D.W. "An efficient and reliable e-drm scheme for mobile environments", *Expert Systems with Applications*, **37**(9), pp. 6176-6181 (2010).

18. Li, J.S., Hsieh, C.J., and Hung, C.F. "A novel DRM framework for peer-to-peer music content delivery", *Journal of Systems and Software*, **83**(10), pp. 1689-1700 (2010).

19. Ke, C.K. and Lin, Z.H. "An approach for secure data exchange: Experiments on android-based mobile device", *Scientia Iranica*, **22**(4), pp. 1586-1593 (2015).

20. Iannella, R. "The open digital rights language: XML for digital rights management", *Information Security Technical Report*, **9**(3), pp. 47-55 (2004).

21. ISO "Information technology-multimedia framework (mpeg-21) part 5: Rights expression language", Standart 21000:5, International Organization for Standardization, Geneva, Switzerland (2004).

22. Tosuntas, S.B., Karadag, E., and Orhan, S. "The factors affecting acceptance and use of interactive whiteboard within the scope of FATIH project: A structural equation model based on the unified theory of acceptance and use of technology", *Computers & Education*, **81**(3) pp. 169-178 (2015).

23. Burnett, I., Davis, S., and Drury, G. "MPEG-21 digital item declaration and identification principles and compression", *IEEE Transactions on Multimedia*, **7**(3), pp. 400-407 (2005).

24. Polo, J., Prados, J., and Delgado, J. "Interoperability between ODRL and MPEG-21 REL", *ODRL Workshop*, Vienna, Austria, pp. 65-76 (2004).

25. Prokofyeva, N. and Boltunova, V. "Analysis and practical application of PHP frame-works in development of web information systems", *Procedia Computer Science*, **104**(1), pp. 51-56 (2017).

Biographies

Ahmet Özmen received the BS degree from Electronics and Communication Engineering Department of Istanbul Technical University, Istanbul, Turkey, in 1987, and MS and PhD degrees from Electrical Engineering Department of University of Kentucky, Lexington, USA, in 1996 and 2000, respectively. He is currently with the Department of Computer Engineering at Sakarya University. His main research activity is related to distributed parallel systems.

Ahmet Şanslı received his BS and MS degrees from Computer Engineering Department of Sakarya University in 2005 and 2007, respectively. Currently, he works as a Software Development Specialist in Computer Research Center at Sakarya University, Sakarya, Turkey. His particular areas of interest are software systems and DRM systems.

Veysel Harun Şahin received MS degree from Electrical and Electronics Engineering Department, and PhD degree from Computer Engineering Department of Sakarya University, Sakarya, Turkey. He is currently with the Department of Software Engineering at Sakarya University. His particular areas of interest are software systems and DRM systems.