Protecting Digital Content Utilizing Standards

DRM Server

Version 2.0.1
21 June 2006

Prepared By:
Mutable, Inc.
info@mutablemedia.com

Table of Contents

info@mutablemedia.com

1 Introduction

DRM server provides the central entity which handles all DRM related information, like content protection keys, users and associated rights. It communicates with DRM clients using messaging systems.

2 Overview of the Server

DRM server implements two messaging systems for communication with DRM clients:

- *OpenIPMP* system.
- OMA system as defined in OMA DRM documentation.

Communication is carried out via web services.
DRM server stores all DRM data using databases, thus providing means for later retrieval.

3 Building and Installing

DRM server has been built, installed and tested on Windows and Linux platforms. On both platforms there are three prerequisites:

- MySQL database server.
- Java Software Development Kit.
- JBoss application server.

The server was tested using MySQL Server 4.1, Java 1.4.2.10 SDK and JBoss 4.0.2 application server.

Except for installation of the three prerequisites, one must set up following environment variables:

- JAVA_HOME, set to root Java SDK directory.
- JBOSS_HOME, set to root JBoss application server directory.
- The PATH variable must contain the path JAVA_HOME/bin and the path to mysql command line client executable (used to access MySQL server).

After the prerequisites are installed and working and environment variables are set, go to src/server directory. If MySQL server is not installed on default address and port (localhost

info@mutablemedia.com

and 3306), please edit files OMADRMWS/server_config.xml and osms/ serverConfigData.xml to proper data for MySQL address and port.
Since OMA DRM clients depend on the exact URL string from which OMA DRM server module communicates with them, please set the value of <ri><url> tag in OMADRMWS/server_config.xml file to IP address and port from which you want OMA DRM server module to communicate with OMA DRM clients. Typically, this could be localhost:8080 or <IP address>:<IP port> where <IP address> and <IP port> are the values appropriate for the host machine where DRM server is installed.

Building and installation consists in running the install.bat script on the Windows platform or the install.sh script on the Linux platform. It is assumed that MySQL server is up and running. On the Linux platform, one needs to set execution permission of the install.sh script (chmod +x install.sh) and log in as a user with root privileges. Both scripts will ask for MySQL user name and password, to be able to connect to MySQL server and set up databases. Please enter the necessary information. We recommend passing the root user name and password.

After the installation is done, DRM server is ready to be started.

4 Starting and Stopping the Server

After the code is built and deployed, one starts and stops the DRM server by starting and stopping the JBoss application server, respectively. It is assumed that MySQL server is up and running.

To start the JBoss application server, go to JBOSS_HOME/bin directory and run run.bat script on the Windows platform or run.sh script on the Linux platform. On the Linux platform one needs to set execution permission of run.sh script (chmod +x run.sh) and log in as a user with root privileges. This will start the JBoss application server and thus DRM server as well.

To stop the JBoss application server, go to JBOSS_HOME/bin directory and run shutdown.bat -S on the Windows platform and shutdown.sh -S on the Linux platform. On the Linux platform one needs to set execution permission of shutdown.sh script (chmod +x shutdown.sh) and log in as a user with root privileges. This will stop the JBoss application server and thus DRM server as well.

5 Usage

DRM server consists of two modules: one handling an *OpenIPMP* system and another handling an OMA DRM system.

info@mutablemedia.com

To access the *OpenIPMP* system, go to http://localhost:8080/openipmp/jsp/login.jsp. The *OpenIPMP* system authenticates users via user name and password. To create a new user, follow the "REGISTER New User" link. If already registered, write in the user name and password to access the system. Following "Keystore" link, one can download a p12 file which is used by DRM clients for saving associated rights and protection keys. See DRM Plugin SDK documentation for more details.

The OMA DRM system currently cannot be accessed via a web page. It can only be accessed by DRM clients via a web service. For OMA DRM, it is necessary to have a server certificate file, a server CA certificate file and a server private key file. Optionally, one can have trusted CA certificate files, which are used as a CA for device certificates. Certificate files are expected to contain DER encoded X509 certificates. Public keys in certificates are expected to be RSA with MD5 keys. The private key file is expected to contain a DER encoded unencrypted PKCS8 private key. The private key is expected to be an RSA with MD5 key. These files can be created using for example openssl. Assuming that openssl and Perl are installed and the path to its executable is contained in the global path, one can use src/Demo/data/OMADRM_create.bat to create certificates and keys. Also note that depending on where CA.pl resides, it may be necessary to modify the relative path in in OMADRM_create.bat Look at the comments in the script for details. You must then run the src/server/OMADRMWS/admin_OMADRM.bat / .sh script in order to re-administer the OMA DRM device (client) data with the new server CA certificate or trusted device CA certificates. Note that this step is optional for Demo and testing, but is REQUIRED for a production deployment.

If trusted CA certificates are used, then a device must identify itself with a certificate chain which roots in any of the trusted CA certificates. Otherwise, a device can identify itself with any. certificate. The situation is symmetric on the device side. If a device uses trusted CA certificates, then the server CA certificate, which signs the server certificate, must be one of the CA certificates trusted by the device. Otherwise, the server can send any certificate to identify itself.

Default certificate and key files were added to MySQL database on installation. A script to update these files was copied to JBOSS_HOME/openIPMP/OMADRM directory. On the Windows platform it is the admin_OMADRM.bat script. On the Linux platform it is the admin_OMADRM.sh script. The respective script can be used later to update these files in theMySQL database. Please look at the comments in the scripts to perform these operations.

info@mutablemedia.com