Implementation of Digital Rights Management on the Android Mobile Terminal

Dongdong Dou, Shaobing Liu, Longzhen Jia

Department of Driver Training and Service, Bengbu Automobile NCO Academy, Bengbu, China

Department of Equipment Technology, Bengbu Automobile NCO Academy, Bengbu, China

Keywords: Android; digital rights management; platform; authentication; segmentation decryption.

Abstract. With the development of internet and Smartphone, the access to multimedia content (eBook, music, movie, video etc.) has become easier and easier on the mobile phone. Thanks to its good performance, Android mobile phone has attracted more and more people since the introduction of Android OS by Google .It’s necessary to pay much attention to the security of digital content on the Android mobile phone. In this paper, an enhanced DRM system based on OMA 2.1 was proposed, which can efficiently protect the digital content on the mobile phone.

Introduction

In the past few years, Android mobile terminal (Smartphone and tablet computer) has attracted more and more people because of its good performance. Meanwhile, with the rapid development of the third generation mobile communication technology, the network transmission rate on the mobile terminal has a great improvement. The digital content, which only can be transmitted among the computers before, has been widely circulated on the mobile terminals. More and more people consume digital content on the mobile terminals. So it’s necessary to pay more attention to the security of digital content on the mobile terminals. Digital right management (DRM) has becoming a pressing concern for the digital content business.

DRM is developed to protect the copyright of the digital content, which integrates encryption, decryption and signature. Nowadays, there are several DRM standards all over the world. The OMA DRM is the most popular DRM standard in mobile communication which is established by Open Mobile Alliance (OMA).The OMA DRM is the most perfect so far with the largest number of members.

However, there are also several problems in the application of OMA DRM on the mobile terminal .Firstly, to ensure the security of the whole DRM system; the OMA DRM needs a more complex production environment, for example, the build of the Certificate Authentication (CA), the management of the certificates. So it is rarely used in China. Secondly, because of the rapid update of the mobile terminal and the increased competition, the price of the mobile terminal has declined sharply. More and more people change their terminals even the current terminal was bought a few months ago. New problem appears, how can one who consumes digital content on an old terminal access his digital content on the new mobile terminal. Finally, as is known to all, the multimedia content may have a large size. However, due to the limit of the poor computation capability and battery power of the mobile terminals, it may be undesirable to encrypt a whole large multimedia content which may be more than 10M bytes.

In this paper, we put forward a DRM system model based on OMA 2.1, which can effectively solve the above problems. In this model, from the delivery of certificate to the business and the display of digital content, we all make a detailed description.

System Architecture

As Fig.1 shows, the DRM system consists of 3 parts, DRM Server, Platform, and Terminal Client.

Now we describe the three parts in detail.

DRM Server the DRM Server consists of five parts which integrates content encryption, the delivery and management of certificates, the generation of rights object (RO), the authentication and so on.

1). Certificate Database It is used to store certificates which have been authenticated by CA. All the certificates which have bound with the mobile terminals are stored in this database.

2). Content Server It is a web service which can be called to encrypt the digital content to the DRM Content Format (DCF). DRM Content is encrypted with a symmetric content encryption key (CEK). Content will be pre-packaged, content packaging does not have to happen on the fly.

3).Rights Server It is also a web service which can be called to generate the rights object (RO).

Rights Objects associated with DRM Content have to be enforced at the point of consumption. This is modeled in the OMA DRM specifications by the introduction of a DRM Agent. The DRM Agent embodies a trusted component of a Device, responsible for enforcing permissions and constraints for DRM Content on the Device, controlling access to DRM Content on the Device, and so on.

4).Certificate Server It is used to deliver the certificate which is bound with the mobile terminals .One terminal can be bound with the only one certificate.

5).Authentication Server When there has a business between terminal and platform, the platform will call this server to authenticate the validity and security of the business.

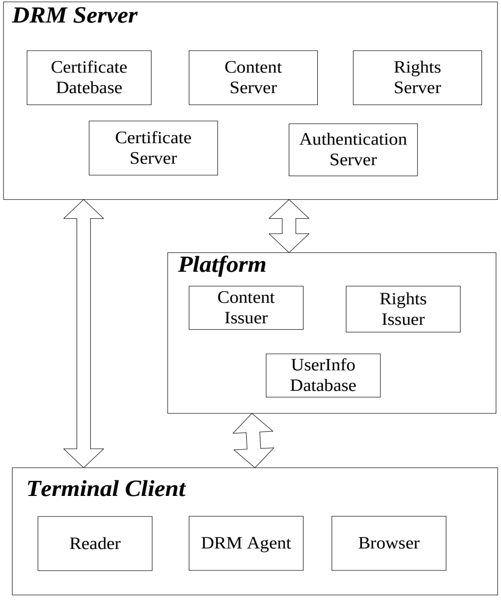

Figure 1. Function Architecture

Platform actually, it is an online mall. All the businesses on the mobile terminals are with the help of platform. It’s consist of three parts, Content Issuer (CI), Rights Issuer (RI) and User Info Database.

1). Content Issuer The content issuer is an entity that delivers the DRM Content. OMA DRM defines the format of DRM Content which is delivered to DRM Agents, and the way DRM Content can be transported from a content issuer to a DRM Agent using different transport mechanisms, such as HTTP. In this DRM system, the content issuer may do the actual packaging of DRM Content by the call of the Content Server in the DRM Server.

2). Rights Issuer the Rights Issuer is an entity that assigns permissions and constraints to the DRM Content, and generates Rights Objects by the call of the Rights Server. A Rights Object is an XML document expressing permissions and constraints associated with a piece of DRM Content. Rights Objects govern how DRM Content can be used, DRM Content cannot be used without an associated Rights Object, and may only be used as specified by the Rights Object.

3). User Info Database the User Info Database is an entity that stores the specific information associated with one user. Terminal Client A client is the user of DRM Content. It is consist of three parts, Reader, DRM Agent and Browser. The client can only buy the DRM Content through the

Browser and access DRM Content through a DRM Agent. 1). Reader A Reader is to display the DRM Content according to the permissions and constraints specified in a Right Object with the help of DRM Agent. 2).DRM Agent The entity in the Device that manages permissions for Rights Object on the Device. A DRM Agent embodies a trusted entity in a mobile terminal. This trusted entity is responsible for enforcing permissions and constraints associated with DRM Content, controlling access to DRM Content, and so on. DRM Content cannot be used without an associated Rights Object, and may only be used according to the permissions and constraints specified in a Rights Object. 3). Browser A browser is an entity that connects with the platform. With the help of the browser, we can consume the DRM Content on the platform.

Implementation

An integrated, commercial DRM system must be consist of the implementation of the Public Key Infrastructure (PKI), DRM Server, Platform, Terminal Client, and so on. Due to the space limitation, detailed analysis will be not allowed.

In this part, we will focus mainly on the implementation of the terminal client in the Android mobile terminal, which consists of the terminal initialization, the consumption of the DRM Content and the access of DRM Content.

Terminal Initialization. The process of the terminal initialization is actually the process of the DRM Agent authentication. Every DRM Agents has a unique private/public key pair and a certificate. The certificate includes additional information, such as maker, device type, software version, serial numbers, etc. This allows the content and rights issuers to securely authenticate a DRM Agent. Any privacy aspects with releasing such information are addressed in the technical specifications.

It seems that the certificate plays a very important role in the DRM Agent. So how to distribute a unique certificate to the DRM Agent becomes a problem to be solved. It will waste lots of work and may be caused security problems to provision a certificate in the DRM Agent before its delivery. Automatic certificate distribution can deal with this problem more efficiently. In order to implement the automatic certificate distribution, we provision a same certificate in the DRM Agent, which can be called Common Certificate. The Common Certificate is used to identify the validation of the DRM Agent according to the standard PKI procedures.

Then, as Fig.2 shows, we use International Mobile Equipment Identity (IMEI) to uniquely identify the DRM Agent. As is known to all, the IMEI is a number, usually unique, to identify the valid mobile Terminal .Then the unique certificate can be delivered to the DRM Agent by the Certificate Server according to the IMEI of the terminal.

The DRM Agent can get the IMEI of the terminal as its unique identify. Then DRM Agent sends the IMEI to the Certificate Server. When the Server gets the IMEI, it will query if the DRM Agent associated with the IMEI has registered the certificate. If DRM Agent has registered the certificate before, the Certificate Server will push the certificate which associated with the DRM Agent from the Certificate Database. Otherwise, the Certificate Server will push a new certificate which has never used before.

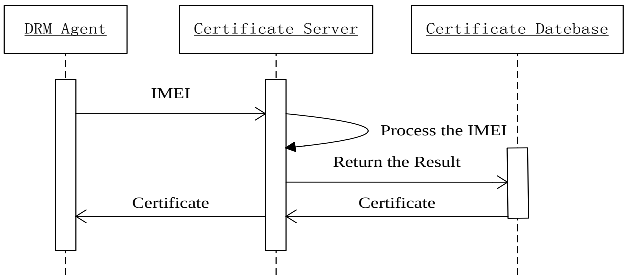

Figure 2. Terminal Initialization

Consumption. After the terminal initialization, we can register an account in order to access the website and consume the digital content in the platform. The information associated with this account, such as username, password, the IMEI of mobile terminal, the consumption record, will be put into the User Info Database.

When we login in our account, we can access the website of platform, and can buy everything we need. There will be a procedure of authentication at the point of consumption to ensure the validity of the trade.

As Fig.3 shows, firstly, when consumption happens, the browser needs an authcode which is generated by the DRM Agent. The browser can connect with the DRM Agent by the JavaScript (JS).Then the DRM Agent packages the business request and authcode into the Digital Envelop.

The browser sends the Digital Envelop to the platform, then the platform calls the Authentication Server to unpackage the Digital Envelop and get the authinfo. The Authentication Server will verify the authinfo and return the verification result. The platform will process the business request according to the verification result .If everything works, the Rights Server will be called to generate the Rights Object associated with the DRM Content and this user. And the business information will be stored into the User Info Database. The process result will be sent to the terminal browser. Then the terminal client will download the Content Object and Rights Object. So far, a complete business procedure has implemented.

The application of the Authentication has greatly improved the safety of the business, which efficiently ensures the validity of the business.

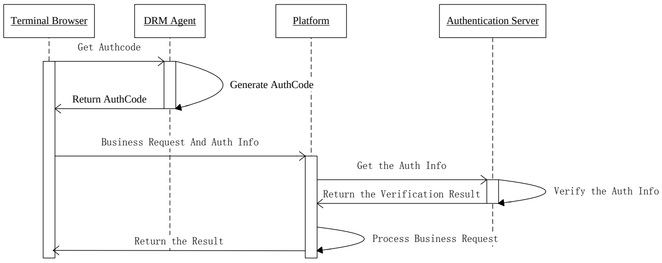

Figure 3. Consumption

Access the DRM Content

It‘s assumed that there has been Content Object file and Rights Object file in the mobile terminal.

The Reader can only display the DRM Content with the help of DRM Agent. The DRM Agent embodies a trusted component of the mobile terminal, responsible for enforcing permissions and constraints for DRM Content on the terminal, controlling access to DRM Content

Full-text decryption is mainly used to decrypt the DRM Content by DRM Agent in the mobile terminals. In this model, which will take much computation cost and energy consumption when the DRM Content has a large size. In this paper, we propose a more efficient scheme which can be called segmentation decryption.

Segmentation decryption can support random access to the DRM Content and only decrypt a small part of the DRM Content. Compared to the full-text decryption, the segmentation decryption reduces the computation time and energy, which can achieve much better performance.

Conclusion

This paper has described an efficient and helpful DRM system on an Android mobile terminal. Firstly, Terminal initialization can support the delivery and management of the certificates on the mobile terminal. Secondly, The User info Database on the Platform can store the consumption information associated with a user. With the help of User info Database, even we replace our mobile terminal, we can know what we have bought before and can download them freely on the new terminal. In this case, it’s unnecessary to worry about the loss of content caused by the replacement of the mobile terminal. Finally, by adopting the segmentation decryption to decrypt a small part of the content, the proposed scheme has a good performance, which can greatly reduce the computation time and the energy consumption.

The terminal client has been put into operation for almost one year, which has been proved to have good performance. After this paper, we will research more about the DRM system on the mobile terminal and make gradual improvement.

References

[1] Lan, X., Xue, J., Tian, L., Hu, W., Xu, T., and Zheng, N. 2009, “A peer-to-peer architecture for live streaming with DRM,” In Proceedings of the 6th IEEE Conference on Consumer Communications and Networking Conference (Las Vegas, NV, USA, January 11 - 13, 2009) IEEE Press, Piscataway, NJ,pp. 1222-1226.

[2]Li, L. Zhao, C. Wang, and F. Ma, “DRM system for multiple cascaded business operators,” 2010 IEEE International Conference on Multimedia& Expo (ICME2010), July 2010, pp.1651–1654.

[3]G. Li, Chejen Hsieh, Chengfu Hong, “A novel DRM framework for peer-to-peer music content delivery,” 2010, pp.1689-1700.

[4]Open Mobile Alliance “DRM Architecture OMA-AD-DRM-V2_2-20110419-C,” 2011 Open Mobile Alliance Ltd.

[5] Jiang Zhang, Bin Li, K. Zhao, Shiqiang Yang, “License Management Scheme with Anonymous Trust for Digital Rights Management,” IEEE International Conference on Multimedia and Expo, 2005, p.257–260.

[6]Q. Zhang, Y. Wang, “A Centralized KeyManagement Scheme for Hierarchical Access Control:” Proceedings, IEEE GLOBECOM, 2009, pp. 2067-2071.

[7]Yoon, Jun Hee Cheon, Yongdae Kim, “Batch Verifications with ID-based Signatures”, Proceedings of ICISC 2004, LNCS 3506, p.233-248, 2005.