

Extending DRM Features to Distributed Environments

SangGyoo SIM, YunSang OH, SukBong LEE
 Software Laboratories, Samsung Electronics, R.O.Korea
 Email: {sanggyoo.sim, yunsang.oh, sukbong.lee}@samsung.com

Abstract—Recent electronic devices are connected with each other by various connectivity techniques. This trend makes computing resources distributed over many devices. However, since the general DRM models assume contents and the corresponding rights should be stored in the same space or storage, they cannot fully support distributed environments. To fully utilize the distributed resource and to provide more user-friendly scenario, this paper assumes the contents and the rights can be stored in different storage. Based on the altered assumption, this paper proposes new models: rights movement model and rights consumption model. In addition, remote playback model is proposed, which is a specialized version of rights consumption model for rendering devices. By enabling free flow of the contents and rights between devices, the usage of DRM systems can be widened and more activated.

Index Terms—DRM, Distributed Computing Environment

I. INTRODUCTION

DRM (Digital Rights Management) has been attracted as an application which can broaden the market of digital contents and multimedia devices. Hence, three different groups of companies have concentrated on DRM: contents providers, service providers, and device manufacturers. Although there have been many DRM schemes after the fundamental concept of DRM was established, they have a common characteristic assumption: the Contents Object (CO) and Rights Object (RO) should be stored in the same space or storage.

In general models including OMA DRM v2.0 [1], Microsoft Windows Media DRM 10 [2] and broadcast-supported DRM systems of DVB TM-CBMS [3], a device stores CO and RO in its internal storage. If a user wants to use the content contained in a CO, the DRM agent of the device checks the validity of the RO referred by the CO and then determines from the context of the RO whether it allows the user to use the CO. Usually state information of the RO traces the CO usage and resides in the device in a secure manner. This common user scenario assumes that the CO and RO are stored in the same storage space. However, this assumption is so restrictive that the diverse usage and scenario of contents cannot be supported in the rapidly changing circumstances.

Commercial equipments evolve with very high speed and too many types of equipment are connected with each other in various ways. We call it a distributed environment. This paradigm shift in environments requires new features which are not supported by the conventional

models of information services including DRM. In aspect of DRM model, it is not restricted that the CO and its corresponding RO should be in the same storage space. If the CO and the RO are allowed to be stored in distributed storage, the general DRM model based on the aforementioned assumption should be altered into a new model.

This paper tries to comply with the need of a new DRM model. The proposed model allows free movement of ROs in distributed environments and rendering CO by referring ROs in remote devices. In addition, this paper proposes an adoption of the proposed model for OMA DRM v2.0.

II. TERMS

To propose our new model, some terminological terms are defined in accordance with OMA DRM v2.0 [1].

- CO (Content Object): Container of Media Objects that are consumed according to a set of Permissions in a Rights Object.
- RO (Rights Object): A collection of Permissions and other attributes such as content encryption key which are linked to Content Object.
- RI (Rights Issuer): An entity that issues Rights Objects to authenticated devices.
- Rights Usage: Rights Usage represents all activities that use RO issued by RI.
- Rights Consumption: Rights Consumption is a kind of Rights Usage that represents a sequential operation for rendering. To render a CO, a DRM agent updates the context of RO and strengthens the constraints for the RO.
- State Information: State Information is used to enforce the stateful RO according to the constraints and permissions expressed in the RO. State Information contains temporary information for the duration from when Rights Usage is activated to when the RO is updated or re-issued.

III. DISTRIBUTED ENVIRONMENT

In distributed environments, there are many cases which conventional DRM systems do not consider.

A. RO Movement

RO movement is a very useful and natural Rights Usage. For example, a user can enjoy convenient portability

or better sound quality by moving ROs to a portable device or a Hi-Fi audio device. General DRM systems cryptographically bind RO to a specific device by encrypting the content key using the public key of the device. To move RO between devices, the source device should decrypt the encrypted content key and then rebind the content key with the public key of the destination device. Although conventional DRM systems define RO sharing and RO export rather than RO movement, they are neither concrete nor efficient. In distributed environments, there happen many RO movements because many devices of various kinds are connected with each other. Therefore, an efficient RO movement model should be designed.

B. RO Consumption

RO Consumption Let assume that a user have many devices which are connected with each other. He may have multiple ROs over devices for a given CO. For example, an RO allows free preview and is stored in Device 1. Another RO allows 5 playbacks and is stored in Device 2. In distributed environments, there are multiple ROs in different devices for a given CO. Therefore, if RI policy permits it then an RO in other device can be consumed to play a CO. Moreover, a special kind of device with only rendering function (without storage) does not have either RO or CO, but must be able to play CO by consuming RO in other devices.

IV. PROPOSED MODEL

This clause proposes three conceptual models: RO movement model, RO consumption model and remote rendering model. First, RO movement model is the underlying model of the proposed models. It enables the movement of RO(s) between two arbitrary devices. Second, by utilizing the RO movement model RO consumption model enables a device to use the RO of other devices as well as its own RO. Last, remote rendering model enables a rendering device without RO and CO to playback contents contained in the CO of other device.

A. RO Movement Model

RO movement model in Figure 1 describes the free movement of ROs and their State Information between arbitrary devices in a very efficient method. The movement consists of the following steps.

- **Step 1. Authentication between devices.** The two devices authenticate each other. Explicit authentication scheme is recommended rather than implicit one for further security. In some authentication schemes, the exchange of each own certificate for public key may be accompanied.
- **Step 2. Building secure state.** The two devices build secure state, by which all communication between the two can be done giving confidentiality and integrity. In general, building secure state is achieved by sharing secure information in common.

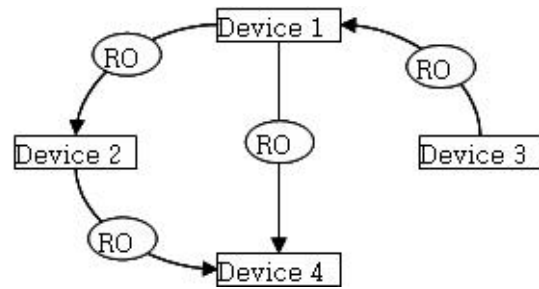


Figure 1. RO Movement

The shared secure information can be used to induce a session key and a integrity check key.

If an authenticated key agreement scheme is used in Step 1, it covers session key agreement and entity authentication, simultaneously. Therefore, Step 1 and Step 2 can be replaced with an authenticated key agreement scheme.

- **Step 3. Moving RO and State Information.** While the two devices remain in secure state, ROs of the source device are moved to the destination device. If RO has State Information, it should be moved with the RO to the target device in secure state giving integrity and confidentiality. There are two methods to move an RO. For easy understanding, the source device and the destination device are denoted as Device 1 and Device 2, respectively. (See Figure 1.)

- Method 1. Device 1 decrypts the encrypted content key, and then rebinds the content key by encrypting it with the public key of Device 2. Device 1 sends the RO and the content key encrypted with the public key of Device 2.
- Method 2. Device 1 decrypts the encrypted content key, and then moves the decrypted content key and the RO to Device 2.

In Method 1, Device 1 can use another key instead of the public key of Device 2 to rebind the content key. The rebinding key can be possibly replaced with another key which is shared between Device 1 and Device 2.

In Method 2, Device 2 receives the ‘plain’ content key rather than the encrypted content key. Therefore, Device 2 should store the content key securely. Device 2 can encrypt the content key so that others can not decrypt the key. Or Device 2 can use its special secure storage.

In RO movement, the RI signature of RO loses the meaning of authenticity after movement. RI generates the signature on the RO which contains the encrypted content key with the public key of Device 1. However, Device 1 may not send the encrypted key with the public key of Device 1 but with the public key of Device 2. By this reason, the authenticity from RI is replaced with the integrity and confidentiality from secure state of devices. To allow RO movement, RI should issue ROs according to its security policies.

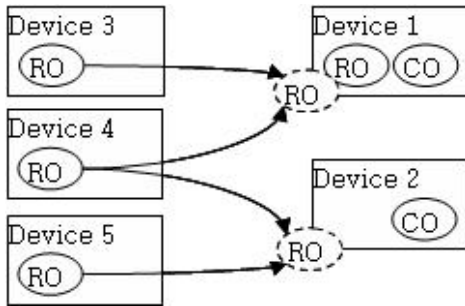


Figure 2. RO Consumption

B. RO Consumption

RO consumption model in Figure 2 describes how a device consumes the RO stored in its own storage or other device's storage.

Assumption: Device 3, Device 4 and Device 5 have the RO bound to themselves. Device 1 has both of the CO and its corresponding RO. Device 2 has only the CO.

RO Consumption: Device 1 wants to use the CO whose usage requires consuming its corresponding RO. Hence, Device 1 tries to search the RO corresponding to the CO. Firstly, it explores over its own storage. If it succeeds to find out the RO in its own storage, it determines whether to keep going on exploring other device(s) which are 'possibly connected'. If it succeeds to find the RO additionally in other device(s) connected (for example, Device 3 and Device 4 in Figure 2) to it, it determines the RO to be used between in its own storage and in other connected device(s). The number of the founded RO(s) in other connected device(s) can be more than one. Although Device 1 succeeds to find the RO in other device, it can use the RO in its own storage rather than one(s) in other connected device(s). Moreover, it can use both the RO in its own storage and the RO(s) in other device(s), simultaneously.

Similarly as the case of Device 1, Device 2 wants to use the CO which may be probably different from the CO of Device 1. However, Device 2 has no RO corresponding to the CO inside of its own storage. Device 2 explores other device(s) which are 'possibly connected'. If it succeeds to find out the RO in other device(s) connected (for example, Device 4 and 5 in Figure 2) to it, it determines the RO to be used among the ROs of other connected device(s).

The term 'possibly connected' means arbitrary two devices are already connected or can connect successfully with the each other. Some devices make a connection and keep on the connected state. Some other devices are devices which can be in touch with each other and reconnected when needed. Thus, 'possibly connected' denotes the available or possible connection.

C. Remote Rendering Model

Assumption: Device 1 has neither the CO nor the RO inside its own storage. Device 2 and Device 3 have both the CO and the RO in their storage.

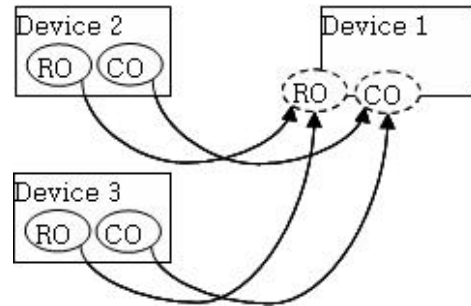


Figure 3. Remote Rendering

Remote Rendering: Device 1 may be a rendering device which has only playback functionality and can not store any RO and CO in its own memory. Device 1 explores the 'possibly connected' devices to get the list of the COs. After selecting a CO in the list of COs stored inside other devices, Device 1 requests the RO corresponding to the selected CO to the 'probably connected' device. By consuming the RO, Device 1 can playback the contents contained in the CO.

The device giving the CO to Device 1 would be different from the device giving the RO to Device 1. In addition, Device 1 can consume the multiple ROs stored in other devices to playback the one CO.

V. SECURITY CONSIDERATIONS

This section defines security issues needed for the adoption of the proposed model.

A. Mutual Authentication

Devices can authenticate each other (i.e. mutual authentication) based on credentials that are securely provided to each. The result of this mutual authentication allows the devices to establish a secure channel for the exchange and sharing of secret elements.

B. Message Transaction

Based on the mutual authentication, an RO and its state information or any necessary messages can be securely delivered between devices regardless of lower layer communication. The secret elements are used to guarantee the confidentiality and integrity of the message transaction. A symmetric encryption or keyed hash function may be used. Replay of message transactions will not result in any action being taken by the receiver that was unintended by the original transmitter of those messages.

C. Rights Protection

An RO stored in a device is cryptographically bound to the DRM agent in the device. While moving the Rights Object and its state information, the result of the mutual authentication can be used to protect the confidentiality of sensitive parts (e.g. CEK; Content Encryption Key) and, integrity of the RO itself and its state information. After

the move operation, the RO is still securely bound to a trusted entity: the DRM agent of the source device or DRM agent of the target device.

D. Protection of Rights Consumption

To consume the RO, the source DRM agent sends CEK to the target DRM agent. The device decrypts a DRM content with the CEK. The result of the mutual authentication can be used to protect the confidentiality of the CEK. If the mutual authentication becomes invalid, the transferred CEK has to be invalidated for the target DRM agent.

E. Trust Model

The DRM agent of a device has to be trusted by the DRM agent of other devices, in terms of authorization, data protection, and root of trust. Only an authorized DRM agent can access and utilize data stored in other devices and the DRM agent has to guarantee the integrity and the confidentiality of the data. The DRM agent is also trusted enough to hide security elements (e.g. private key) from other entities.

Each DRM agent is provisioned with a unique key pair (a private key and a public key) and an associated certificate. The certificate identifies the DRM agent and certifies the binding between the SRM Agent and the key pair. This allows DRM agents to securely authenticate other DRM agents. This allows DRM agents to securely authenticate DRM agent of other devices.

The information in the certificates of the DRM agents enables to trust each other SRM Agent and send the sensitive data of the RO and its state information.

F. Aborted Transaction Recovery

If transaction fails during Rights consumption, there is possibility for Rights not to be updated properly. If transaction between the device and the SRM is failed during RO movement operation, there is also possibility for users to lose their RO. There are two examples of abnormal movement failure:

In case of RO movement from source to target, RO is removed from the source device immediately after the RO has been moved to the target device. If the transaction is failed before the moved RO is reached to the target device, users lose the RO.

If the RO is removed from the source device after the RO have reached the target side successfully, the transaction failure may cause another security problem - unexpected duplication of the Rights.

The transaction failure may occur by unstable communication between the devices which happens rarely or by unexpected physical disconnection. To prevent the above problems, OMA DRM agent has to provide with ways to recover the transaction failure.

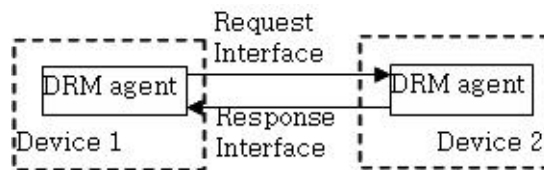


Figure 4. Architectural Model

G. Exclusive Rights Consumption

If multiple DRM agents try to consume a single RO in one device, only one DRM agent consumes the Rights exclusively. OMA SRM achieves renewed availability of the RO for consumption by other DRM agents while guaranteeing the exclusivity.

VI. ADOPTION MODEL

A. Architectural Model

Figure 4 shows the architectural diagram of the proposed model.

A DRM agent embodies a trusted entity in a device. This trusted entity is responsible for enforcing permissions and constraints associated with DRM Content, controlling access to DRM Content, etc. The DRM agent of Device 1 securely communicates with the DRM agents of Device 2 to control and manage the Rights stored in Device 2.

The Request Interface carries transactions and data between the DRM agents including:

- Rights Objects and associated state information delivered from the DRM agent of Device 1;
- Requests from the DRM agent of Device 1 to read the list of Rights in Device 2; and
- Requests from the DRM agent of Device 1 to update the state information of RO stored in Device 2.

The Response Interface carries transactions and data between the SRM Agents including

- Rights Objects and associated state information delivered from the SRM Agent in Device 2;
- List of Rights stored in Device 2; and
- Acknowledgement of requests from the DRM agent of Device 1.

All transactions and data of the Request Interface and Response Interface should be secured using mechanisms for mutual authentication and securing communication between the DRM agents.

B. Functional Model

Figure 5 shows the functional diagram of the proposed model. The functional model consists of 5 functions to support the proposed model. In the functional model, there are two devices: the source device and the target device. The target device tries to read the ROs stored in the source device and the RO are moved from the source device to the target device. In Figure 4, the target device is Device 1 which generates the Request Interface,



Figure 5. Functional Model

and the source device is Device 2 which generates the Response Interface corresponding to the Request Interface from Device 1.

Move Rights

This function moves a Rights Object and its associated state information (i.e. Rights) from a DRM agent to the other DRM agent (or vice versa). After the move, the Rights Object and its

associated state information is present in the destination and deleted from the source and it is guaranteed that duplication or loss of Rights is not possible.

Read Rights

The DRM agent reads the Rights Object and its associated state information from source device (e.g. Device 2 in Figure 4) and can use it (i.e. locally consume) if and only if the same DRM agent is securely connected to the DRM agent. That is, the source device should keep the connection with the target device while the target device reads the RO and its state information stored in the source device.

Update Rights State Information

The DRM agent updates the state information of RO stored in the source device that is locally consumed by the DRM agent of the target device.

Get Rights Information

The target DRM agent reads the detail information of arbitrary RO and its associated state information from the source device. This operation should be under the mutual authentication.

Get Rights List

The DRM agent of the target device retrieves a list of RO identifiers from the source SRM Agent. The source DRM agent can provide this list identifying ROs that are associated with a specific Content Object.

VII. ADOPTION FOR OMA DRM V2.0

The current OMA DRM 2.0 system allows RO to reside in a device and to be consumed without exposure out

of the device. The 'export' function has been defined in OMA DRM 2.0 to allow the exposure of not only stateless RO but also stateful RO. However, the 'export' function converts the RO format to an RO of other DRM systems causing that no original DRM system (OMA DRM 2.0) characteristics are remaining. Besides the 'export' function, there is 'domain' function as well. The 'domain' function enables RO shared among devices under the control of RI. Devices can join to and leave from a domain. If a device is joined in a domain, the device receives a domain key. For the domain, RI issues a domain RO which can be consumed by devices having the domain key. If the device leaves the domain, the domain key in the other devices are updated and the leaving device cannot access RO which is issued after it left the domain. OMA DRM 2.0 concerns the security of stateful RO in the 'domain' function, because consuming state of the RO must be protected and shared between devices in a secure manner. Therefore, the 'domain' function targets only stateless RO.

If it is assumed that the OMA DRM 2.0 is extended to support the RO movement of ROs which is defined in this paper, the following features must be supported in the OMA DRM 2.0.

A. Authorized Transmission

Devices must establish communication channels between them and protect the channels. OMA DRM devices have their own certificates in the format of X.509 [4]. The certificates can be used to establish the secure channel. In case of establishing authorized secure channel between two devices, both exchange their certificates and verify them. By using the public key in the certificates, the integrity and the confidentiality of the secure channel can be guaranteed.

B. State Information Format

State information of RO traces all usage of CO and records them. For RO movement, state information must be also moved. OMA DRM 2.0 doesn't define a state information format of RO. State information format must be readable in any devices by defining standard format. The aforementioned authorized transmission must be used for the movement of state information.

C. RO Rebinding

RO in OMA DRM 2.0 is cryptographically bound to a specific device. RO is bound to a device by encrypting CO encryption key in the RO with the public key of the device. For the RO movement, RO(s) must be able to rebind to other devices than the originally bound device. For an instance, if a RO bound to device X is moved to device Y, CO encryption key of the RO is decrypted by the public key of device X and encrypted by the public key of device Y.

D. Extension of Rights Expression Language

Most of legacy DRM systems including OMA DRM 2.0 use Rights Expression Language such as ODRL and XrML which is used to describe RO permissions, constraints and CO protection scheme. The Rights Expression Language may optionally be extended to be able to describe the RO movement. Aforementioned state information must be able to trace the RO movement and record it.

In the OMA DRM 2.0, the RO consumption in remote devices which is proposed in this paper is not conceptually supported. To support the functionality in the OMA DRM 2.0, the following features must be supported.

E. Key Transmission

Devices must be capable of passing CO encryption key in a secure manner. The aforementioned Authorized Transmission must be used for the Key Transmission.

VIII. CONCLUSION

Recent general DRM models such as OMA DRM assume a CO and the corresponding RO to be stored in the same device. However, the assumption is too restrictive in a distributed environment. Since computing resource exists distributively in distributed environments, to fully utilize the distributed resource it needs to be supported to store the CO and the RO in different storage. Hence, this paper proposes the new model to comply with the need.

The proposed model consists of three detailed models: the RO movement model, the RO consumption model and remote rendering model. The RO movement model enables the movement of RO(s) between two arbitrary devices freely. The RO consumption model enables a device to use the RO of other devices as well as its own RO by utilizing the RO movement model. The remote rendering model is a specialized version to the RO consumption model and enables a rendering device with no RO and CO to playback contents contained in the CO of other device.

The proposed model supports DRM schemes to work even in distributed environments where the COs and the ROs may not exist in the same storage. Since more user-friendly scenarios are possible using the proposed model, the usage of DRM systems can be more widened and activated.

REFERENCES

- [1] "OMA DRM Specification V 2.0, Open Mobile Alliance," *OMA-DRM-DRM- V2.0-20041210-C*, 2004.
- [2] "Microsoft Windows Media DRM 10," Microsoft, available at MSDN of Microsoft, 2005.
- [3] "Technical Drafts of DVB CBMS - Convergence of Broadcast and Mobile Services," Digital Video Broadcasting, *tm-cbms 1221, tm-cbms 1222, tm-cbms, 1223 and tm-cbms 1224*, 2005.
- [4] R. Housley, W. Ford, W. Polk, D. Solo, "Internet X.509 Public Key Infrastructure - Certificate and CRL Profile," *RFC3280*, April 2002.

SangGyoo SIM was born in Masan, South Korea. He received his PhD degree in Department of Electronics of Electrical Engineering from the Pohang University of Science and Technology (POSTECH), Pohang in 2004, his MS degree in the same department from POSTECH in 1998, and his BS degree in the same department from POSTECH in 1996.

He is currently a Senior Engineer of Software Laboratories from Samsung Electronics, Suwon, Korea. His current interests include the Digital Rights Management (DRM) systems and security issues of consumer electronics and portable devices.

Dr. Sim is a member of KIISC (Korea Institute of Information Security and Cryptography).

YunSang OH was born in Daegu, South Korea. He received his MS degree in Department of Computer Science from Sogang University in Seoul, Korea in 2002 and his BS degree in the same department from Sogang University in 1997.

He served his Internship in Siemens VDO Automotive, Regensburg, Germany. He was also an Instructor of Computer Class in Korea-Sri Lanka Technical and Vocational Training Institute, Sapugaskanda, Sri Lanka. He is currently an Engineer of Software Laboratories from Samsung Electronics, Suwon, Korea.

His current interests include the Digital Rights Management (DRM) systems and multimedia contents protection.

SukBong LEE was born in Kwangju, South Korea. He received his MS degree in Interdisciplinary Program of Information Security from Chonnam National University in Kwangju, Korea in 2004 and his BS degree in Department of Mechanical Engineering from the same university in 2001.

He is currently an Engineer of Software Laboratories from Samsung Electronics, Suwon, Korea.

His current interests include the Digital Rights Management (DRM) systems and computer forensics.