An Enterprise Digital Right Management Scheme with Anonymous Trust for Mobile Devices

Jen-Ho Yang
Department of Multimedia and Mobile Commerce, Kainan University,
No. 1, Kannan Road, Luzhu, Taoyuan County, 33857, Taiwan
E-mail: jenhoyang@mail.knu.edu.tw

Hung-Ming Sun
Department of Information Management, Kainan University,
No. 1, Kannan Road, Luzhu, Taoyuan County, 33857, Taiwan
E-mail: sunhm@mail.knu.edu.tw

Ping-Liang Chen
Department of Multimedia and Mobile Commerce, Kainan University,
No. 1, Kannan Road, Luzhu, Taoyuan County, 33857, Taiwan
E-mail: pingliang@mail.knu.edu.tw

Keywords: enterprise digital right management, authentication, user privacy, anonymity, mobile device

Received: August 3, 2012

In recent years, various enterprise digital right management (E-DRM) schemes have been proposed to protect and manage access rights of digital contents for the enterprise applications. However, we find that the previous E-DRM schemes do not protect the user privacy while mobile users access digital contents. In addition, the previous E-DRM schemes have high computation and communication loads. Besides, these schemes do not provide usage tracking for the digital content, and thus the digital right may be abused by malicious users. To solve the above problems, we propose a new E-DRM scheme with anonymous trust for mobile devices in this paper. The proposed scheme has low computation and communication loads, and it provides the user anonymity and usage tracking. Therefore, the proposed scheme is more efficient and practical than the related works for E-DRM applications.

Povzetek: Razvita je nova shema E-DRM za določanje zaupanja v mobilnih napravah.

1 Introduction

The Digital Right Management (DRM) scheme is a digital technique that protects and manages the access rights of digital contents. It can prevent the confidential information of a digital content from unauthorized usages by illegal users. Generally, there are four roles in the DRM scheme: a content provider (author), a consumer (client), a clearing house, and a distributor [1].

The content provider creates the digital content and encrypts it using some proper cryptosystems, such as RSA [2], ElGamal [3], and ECC [4]. Then, the content provider sends the encrypted content to the distributor (e.g., web server or online shop) for online distribution. Note that some researches combine the distributor with the content provider. Next, the content provider sends the usage rules to the clearing house, such as the copy permit, the pay-per-view, and the usage fee, to specify how to use the digital content. Note that the clearing house is responsible for issuing the digital license and handling the financial transactions for the content provider, the distributor, and the consumer.

Assume that the consumer downloads the digital content from the web server. To access the encrypted content, the consumer requests the clearing house to issue a valid license, which contains the decryption key, usage rules, and descriptions of the digital content. Then, the clearing house performs the user authentication mechanisms [5-11] to verify the identity of the consumer. Then, the clearing house can charge the consumer account for the digital content. After the consumer has paid the money, the clearing house sends the license to the consumer. Finally, the consumer has the access rights to use the digital content. Figure 1 illustrates the above DRM infrastructure as follows.

Due to the digital content can be easily obtained and distributed via the Internet, the DRM scheme becomes a popular research topic in recent years. Therefore, various DRM schemes [12-16] have been proposed to protect and manage the access right of a digital content. In 2005, Zhang et al. [12] proposed a license management scheme with anonymous trust for digital rights management (LMSAT) based on Elliptic Curve Cryptosystem (ECC). Zhang et al.'s scheme provides the user privacy protection, and their scheme allows the client access the content by using any permissive device. Thus, their

scheme provides a flexible license acquisition and usage tracking for the digital right management. However, Zhang et al.'s scheme has large computation loads because it utilizes the public-key cryptosystem [2-4]. Thus, their scheme is not suitable for the mobile device with low computation ability.

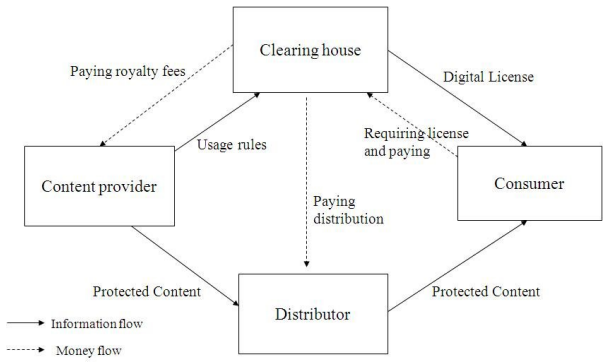

Figure 1: The DRM infrastructure [1].

On the other hand, the DRM scheme can be also applied to digital right protection for enterprise applications. In 2008, Chen [14] proposed a traceable enterprise digital right management (E-DRM) scheme based on one-way hash functions for mobile devices. However, Chang et al. [16] found that Chen's scheme is insecure, and thus they proposed an improved E-DRM scheme in 2010. Chang et al.'s scheme solved the security problem, and it has lower computation cost.

Unfortunately, we find that Chang et al.'s scheme has some flaws shown as follows. First, their scheme does not provide the anonymity so that the user's privacy may be revealed. Second, their scheme does not provide the usage trace of the digital content. Thus, the digital right may be ruined by any malicious user. Third, their scheme uses the certificate to authenticate the mobile user. This increases the communication time because the mobile user must apply the certificate from the certification authority (CA). Fourth, their scheme still has large computation costs because it has to compute the digital signature for authenticating messages.

To solve the above-mentioned flaws, we propose a new E-DRM authentication scheme with user anonymity for mobile devices in this paper. In the proposed scheme, the mobile user can be authenticated by using an anonymity identity so that the user privacy can be well-protected. Moreover, the proposed scheme provides the usage tracking of a digital content, and thus the digital right would not be abused by any malicious user. Instead of using the certificate and digital signature, we only use the one-way hash function and exclusive-or (XOR) operations to accomplish the user authentication so the computation costs can be greatly reduced. According to the above descriptions, the proposed scheme is more efficient and practical than the previously proposed E-DRM schemes, and it is more suitable for the digital right management in mobile environments.

2 Review of Chang et al.'s E-DRM scheme

In this section, we review Chang et al.'s scheme [16] and point out some drawbacks of their scheme. There are six roles in their scheme: the author of the digital content, the package server (PS), the content server (CS), the license server (LS), the authorization authority (AA), and the mobile user (MU). Table 1 shows the notations used in Chang et al.'s scheme.

T, τ	A timestamp and a time constant
$Cert$	The digital certificate of a mobile user
P_i	The *i*-th one-time password
$SEED$	The initial random seed number generated by the authorization authority
N_i	The *i*-th request random number generated by authorization authority
$IMEI$	A unique International Mobile Equipment Identification number of each mobile terminal
msg_{req}	The authorization request message of the mobile user
CID	The identity of a digital content
$DRM-AP_{type}$	The type of the DRM-Enable application
$V_X(\cdot)/S_X(\cdot)$	The verifying/signing function using *X*'s public/secret key
KEY_{CID}	The symmetric key for the digital content with *CID*
$E_{KEY_X}(\cdot)$ / $D_{KEY_X}(\cdot)$	The symmetric encryption / decryption function using a symmetric key KEY_X
$H(\cdot), F(\cdot)$	Two collision free one-way hash functions

Table 1: The notations used in Chang's E-DRM scheme

The package phase:

Step 1. The author produces the digital content and uploads it to PS.

Step 2. PS chooses a symmetric key KEY_{CID} to encrypt M by $C = E_{KEY_{CID}}(M)$. Then, PS generates the content header CH and uses its private key to produce two signatures $Sig_C = S_{PS}(C)$ and $Sig_{CH} = S_{PS}(CH)$. In addition, PS generates the E-DRM formatted file and sends it to CS.

Step 3. PS generates a signature $Sig_{KEY_{CID}} = S_{PS}(CID, KEY_{CID})$ and sends the messages $\{CID, KEY_{CID}, Sig_{KEY_{CID}}\}$ to LS.

Step 4. After receiving the E-DRM formatted file, CS uses PS's public key to verify whether $Sig_C = S_{PS}(C)$ and $Sig_{CH} = S_{PS}(CH)$ are valid or not. If they are valid, then CS stores the E-

DRM formatted file to its database and publishes the file to its public directory.

Step 5. After receiving {*CID*, KEY_{CID} , $Sig_{KEY_{CID}}$ }, LS uses PS's public key to verify $Sig_{KEY_{CID}} = S_{PS}(CID, KEY_{CID})$. If the signature is valid, then LS stores (*CID*, KEY_{CID}) to its database.

Step 6. To access the digital content, MU downloads the E-DRM file from the public directory of CS. According to *URL* of the content header, MU sends the registration request to AA for asking the access right.

Content Identity (*CID*)	Type of the DRM-enable Application ($DRM-AP_{type}$)	Identity of the Decryption Key (KEY_{CID})
Attributes	Signature of the Encrypted Content (Sig_C)	Authorization Authority (*URL*)
The Encrypted Digital Content(*C*)		

Figure 2: The E-DRM formatted file.

The registration phase:

Step 1. MU sends its *IMEI* and *Cert* to AA via a secure channel.

Step 2. AA checks MU by verifying *IMEI* and *Cert*. If MU is valid, then AA generates an initial random number N_1 = *SEED* and sends it to MU through a secure channel.

Step 3. AA stores *IMEI*, *Cert*, and *SEED* in its database and sends them to the LS through a secure channel.

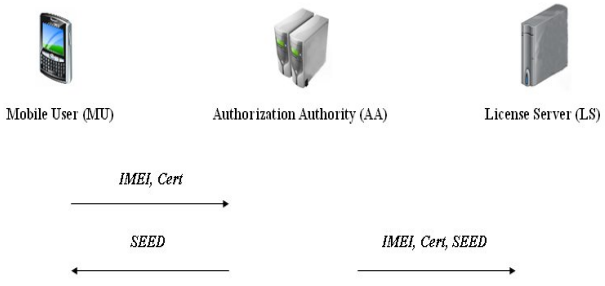

Figure 3: The registration Phase of Chang et al.'s scheme.

The authentication phase:

Step 1. When MU asks for *i*-th authentication from AA, it uses N_i to generate *i*-th one-time password $P_i = H^i(N_i \oplus IMEI \oplus Cert \| T \| CID)$, where $H^i(\cdot)$ denotes the message *x* performs one-way hash function for *i* times, and *T* is a timestamp. Then, MU sends the messages {*i*, msg_{req} , *T*, *CID*, P_i , *Cert*} to AA.

Step 2. AA checks whether *T* is correct or not. If *T* is smaller than τ, AA loads N_i and *IMEI* from its database and computes $H^i(N_i \oplus IMEI \oplus Cert \| T \| CID)$. If P_i and $H^i(N_i \oplus IMEI \oplus Cert \| T \| CID)$ are equal, then AA computes $H^{F(N_i)}(IMEI \| T \| N_i) \oplus N_{i+1}$ and $H(IMEI \| N_i \| N_{i+1})$. Then, AA sends the above messages to MU.

Step 3. AA generates $Sig_{AA} = S_{AA}(IMEI, CID, i, T, Cert, H^{F(N_i)}(IMEI \| T \| N_i) \oplus N_{i+1})$ and sends {*IMEI*, *CID*, *i*, *T*, *Cert*, $H^{F(N_i)}(IMEI \| T \| N_i) \oplus N_{i+1}$, Sig_{AA}} to MU.

Step 4. After MU obtains the messages sent from AA, it computes $N'_{i+1} = H^{F(N_i)}(IMEI \| T \| N_i) \oplus H^{F(N_i)}(IMEI \| T \| N_i) \oplus N_{i+1}$ and checks whether $H(IMEI \| N_i \| N'_{i+1})$ is equal to $H(IMEI \| N_i \| N_{i+1})$ or not. If they are equal, MU accepts N'_{i+1} and keeps it.

Step 5. LS uses the public key of AA to verify the digital signature by $V_{AA}(Sig_{AA}) = IMEI, CID, i, T, Cert, H^{F(N_i)}(IMEI \| T \| N_i) \oplus N_{i+1}$. If the equation holds, LS uses KEY_{CID} stored in its database to generate $H^{F(N_i)}(IMEI \oplus Cert \oplus T \oplus N_i) \oplus KEY_{CID}$ and $H(IMEI \| N_i \| KEY_{CID})$. Then, LS sends the above messages to MU. Finally, MU computes the equation $N'_{i+1} = H^{F(N_i)}(IMEI \| T \| N_i) \oplus H^{F(N_i)}(IMEI \| T \| N_i) \oplus N_{i+1}$ and stores it.

Step 6. To obtain the symmetric key, MU firstly computes the equation: = ( )(⊕ | |) ⊕ . Next, MU checks whether $H(IMEI \| N_i \| KEY'_{CID})$ is equal to $H(IMEI \| N_i \| KEY_{CID})$ or not. If they are equal, MU computes $M = D_{KEY'_{CID}}(C)$ to obtain the digital content.

According to the above description, we point out some flaws of Chang et al.'s scheme as follows. First, their scheme does not provide the user anonymity so that the user privacy may be revealed during the message transactions. Second, their scheme adopts the certificate and digital signature for user and message authentications. This causes large computation and communication loads. Third, their scheme does not consider the usage tracking of the digital content. Thus, the digital right may be abused by malicious users. To overcome the above flaws, we propose a new DRM scheme in next section.

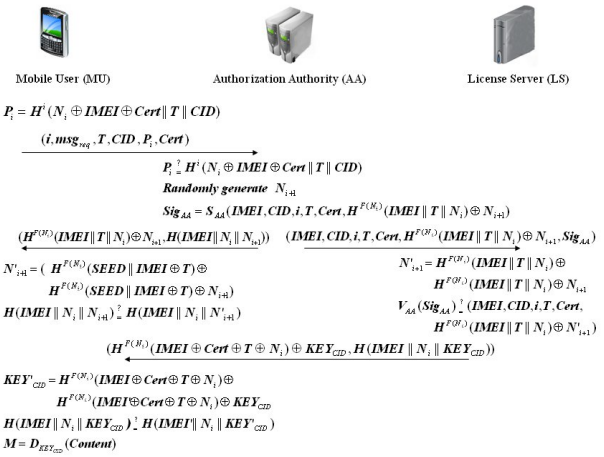

Figure 4: The authentication phase of Chang et al.'s scheme.

3 The proposed E-DRM scheme

There are three roles in the proposed scheme: the content provider (CP), the clearing house (CH), and the client device with the DRM agent (DA). The E-DRM model of the proposed scheme is similar as Figure 1. The difference of the proposed scheme and Chang et al.'s scheme is that the proposed scheme combines the functions of authorization authority and license server into a clearing house. This combination can greatly reduce the communication time, and that is why we do not adopt the model of Chang et al.'s scheme.

In the proposed scheme, DA is loaded in the Client's mobile device. And, DA is responsible for paying the digital content, acquiring the digital license from CH, authenticating the license and the content, decrypting the encrypted content, and reporting the usage to CH. The notations used in the proposed scheme are shown in Table 2.

Notations	Explanations
$H(.)$	A secure one-way hash function
SK	The session key
$E_{SK}(\cdot) / D_{SK}(\cdot)$	The symmetric encryption / decryption with the session key SK
X	The secret key of the content provider
$License$	License of a digital content
$Anonymity\_ID$	The anonymity identity of a client
$Content\_ID$	The identity of a digital content
$Usage\_Rules$	The usage rules of a digital content
$Usage\_Data$	The usage data of the digital content
$\|\|$	The concatenation of strings
SN	The sequence number of the license
$\oplus$	Exclusive-or operation
$Decryption\_Key$	The decryption key for the encrypted digital content
$Other\_Data$	The other information of the license

Table 2. The notations used in the proposed scheme

The registration phase:

In this phase, the client sends the registration information to the CP. Then, CP sends an anonymity identity and the authentication information to the client. Finally, the above information is stored in DA for the authentication. The steps of this phase are shown as follows.

Step 1. For the registration, the client sends his/her identity (ID) and password (PW) to CP in a secure channel.

Step 2. CP generates $Anonymity\_ID$ to compute the authentication information $Auth\_Info = H(Anonymity\_ID \oplus X) \oplus PW$ for the client. Then, CP sends $Anonymity\_ID$ and $Auth\_Info$ to the client.

Step 3. The client stores $Anonymity\_ID$ and $Auth\_Info$ in his/her device for later authentications.

The authentication and license acquisition phase:

In this phase, DA downloads the encrypted digital content from CP. To access the digital content, DA has to get the digital license from CH. In addition, CH is responsible for authenticating the client and sending the license to DA. Note that the client has to pay the money to CH for buying the license. However, we omit the payment steps because it is another research topic for DRM. The steps of this phase are shown as follows.

Step 1. The client computes $Auth\_Info \oplus PW = H(Anonymity\_ID \oplus X)$ by using PW. Then, the client (DA) chooses a random number R_{DA} to compute $S_{DA} = H(H(Anonymity\_ID \oplus X)) \oplus R_{DA}$ and $C_{DA} = H(R_{DA})$. Finally, DA sends $Anonymity\_ID$, S_{DA}, and C_{DA} to CH.

Step 2. CH computes $R_{DA}^{'} = S_{DA} \oplus H(H(Anonymity\_ID \oplus X))$ to check if C_{DA} is equal to $C_{DA}^{'} = H(R_{DA}^{'})$. If they are equal, then CH authenticates that DA is valid. Next, CH chooses random number R_{CH} to compute $S_{CH} = H(H(Anonymity\_ID \oplus X) \| R_{DA}) \oplus R_{CH}$ and the session key $SK = H(H(Anonymity\_ID \oplus X) \| R_{DA} \| R_{CH})$. Finally, CH computes $C_{CH} = H(R_{DA} \| R_{CH} \| SK)$ and sends S_{CH} and C_{CH} to DA.

Step 3. DA computes $R_{CH}^{'} = S_{CH} \oplus H(H(Anonymity\_ID \oplus X) \| R_{DA})$ and the session key $SK^{'} = H(H(Anonymity\_ID \oplus X) \| R_{DA} \| R_{CH}^{'})$. Then, DA computes $C_{CH}^{'} = H(R_{DA} \| R_{CH}^{'} \| SK^{'})$

to check if C_{CH} is equal to $C_{CH}^{'}$. If they are equal, then DA authenticates that CH and *SK* are both valid. Next, DA computes $E_{SK}(Anonymity\_ID \| Content\_ID \| Usage\_Rules \| SK)$ and sends it to CH.

Step 4. CH computes the digital license which contains the decryption key by the equation: $License = \{SN \| Content\_ID \| Usage\_Rules \| Decryption\_Key \| OtherData\}$. Then, CH computes $E_{SK}(License \| SK)$ and sends it to DA. Finally, DA can use *SK* to decrypt $E_{SK}(License \| SK)$ and obtain *License*. Thus, the client gets *Decryption_Key* from *License* and uses it to decrypt the encrypted digital content.

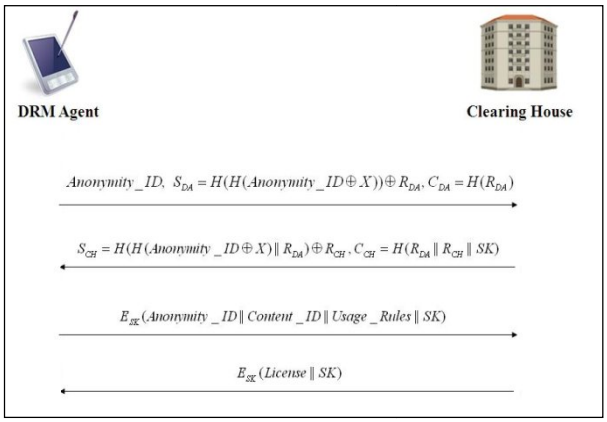

Figure 5: The authentication and license acquisition phase.

The usage tracking phase:

In this phase, CH receives the report of the content usage from DA. For fraud prevention, CH needs to authenticate the validity of DA. In addition, the usage information needs to be encrypted for protecting the user privacy. The steps of this phase are shown as follows.

Step 1. First, CH chooses a random number $\overline{R}_{CH}$ to compute $\overline{S}_{CH} = H(H(Anonymity\_ID \oplus X)) \oplus \overline{R}_{CH}$ and $\overline{C}_{CH} = H(\overline{R}_{CH})$. Then, CH sends $Anonymity\_ID$, $\overline{S}_{CH}$, and $\overline{C}_{CH}$ to DA.

Step 2. DA computes $\overline{R}_{CH}^{'} = \overline{S}_{CH} \oplus H(H(Anonymity\_ID \oplus X))$ to check if $\overline{C}_{DA}$ is equal to $\overline{C}_{CH}^{'} = H(\overline{R}_{CH}^{'})$. If they are equal, then DA authenticates that CH is valid. Next, DA chooses random number $\overline{R}_{DA}$ to compute $\overline{S}_{DA} = H(H(Anonymity\_ID \oplus X) \| \overline{R}_{CH}) \oplus \overline{R}_{DA}$ and the session key $\overline{SK} = H(H(Anonymity\_ID \oplus X) \| \overline{R}_{CH} \| \overline{R}_{DA})$. Next, DA computes $\overline{C}_{DA} = H(\overline{R}_{CH} \| \overline{R}_{DA} \| \overline{SK})$ and sends $\overline{S}_{DA}$ and $\overline{C}_{DA}$ to CH.

Step 3. CH computes $\overline{R}_{DA}^{'} = \overline{S}_{DA} \oplus H(H(Anonymity\_ID \oplus X) \| \overline{R}_{CH})$ and the session key $\overline{SK}^{'} = H(H(Anonymity\_ID \oplus X) \| \overline{R}_{CH} \| \overline{R}_{DA}^{'})$. Then, CH computes $\overline{C}_{DA}^{'} = H(\overline{R}_{CH} \| \overline{R}_{DA}^{'} \| \overline{SK}^{'})$ to check if $\overline{C}_{DA}$ is equal to $\overline{C}_{DA}^{'}$. If they are equal, then CH ensures that DA and $\overline{SK}$ are both valid. Next, CH computes $E_{\overline{SK}}(Anonymity\_ID \| Content\_ID \| SN \| \overline{SK})$ and sends it to DA.

Step 4. DA computes $E_{\overline{SK}}(Usage\_Data \| \overline{SK})$ and sends it to CH. Finally, CH can decrypt $E_{\overline{SK}}(Usage\_Data \| \overline{SK})$ to trace the content usage.

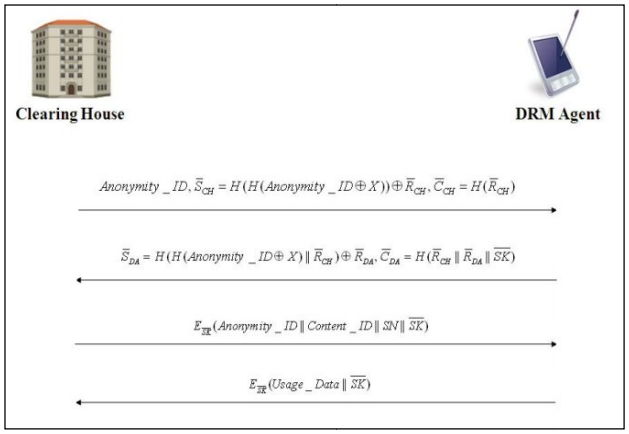

Figure 6: The usage tracking phase.

According to the above descriptions, the proposed scheme is designed by one-way hash functions and XOR operations. Thus, the computation cost of the mobile device can be greatly reduced. In addition, the proposed scheme provides the usage tracking and user anonymity so the digital right and the user privacy can be well-protected. Besides, we simplify the communication model and eliminate the use of the certificate. Therefore, the communication cost of the proposed scheme is much lower than that of Chang et al.'s scheme.

4 Discussions

In this section, we perform some possible attacks on the proposed scheme to show the security analyses as follows.

Man-in-the-middle attack:

Assume that an attacker wants to get the *Decryption_Key* from CH, and he/she impersonates DA to share the session key *SK* with CH. First, the attacker intercepts all messages sent form DA to CH. Then, the attacker impersonates DA to generate the forged

$S_{DA}^{"} = H(H(Anonymity\_ID \oplus X^{"})) \oplus R_{DA}^{"}$ and $C_{DA}^{"} = H(R_{DA}^{"})$. Next, the attacker sends $Anonymity\_ID$, $S_{DA}^{"}$, and $C_{DA}^{"}$ to CH. However, the attacker cannot pass the authentication because $S_{DA}^{"} \neq S_{DA}$. Thus, CH can find that $Anonymity\_ID$, $S_{DA}^{"}$, and $C_{DA}^{"}$ are sent from an attacker. Similarly, an attacker cannot impersonate CH because he does not know the real $H(Anonymity\_ID \oplus X)$. Therefore, the man-in-the-middle attack is infeasible for the proposed scheme.

Outsider attack:

Assume that an attacker wants to get *SK*, and then he/she wiretaps the communications between DA and CH. Thus, the attacker can obtain S_{DA}, C_{DA}, S_{CH}, and C_{CH}. Then, the attacker wants to use the information to compute $SK = H(H(Anonymity\_ID \oplus X) \| R_{DA} \| R_{CH})$. Unfortunately, the attacker needs to know $H(Anonymity\_ID \oplus X)$, R_{DA}, and R_{CH} from S_{DA}, C_{DA}, S_{CH}, and C_{CH} to obtain *SK*. Thus, the attack is impossible because $H(Anonymity\_ID \oplus X)$, R_{DA}, and R_{CH} are protected by the one-way hash function. Therefore, the outsider attack is infeasible for the proposed scheme.

Replay attack:

Assume that an attacker collects the messages once being transferred between DA and CH. To get the digital license or the session key, the attacker may pretend to be DA or CH by resending the pre-collected messages between DA and CH. However, this attack cannot work because all messages are generated and changed according to the random numbers in the proposed scheme. Thus, the messages are different in each time so that the replay attack is infeasible for the proposed scheme.

Stolen-verifier attack:

Assume that the client's device is lost or stolen by an attacker. The attacker may try to use this device to access the digital content as a legal client. However, this attack is impossible because the authentication information $H(Anonymity\_ID \oplus X)$ is protected by user's password *PW*. And, the attacker does not know the correct *PW* to compute $Auth\_Info \oplus PW$. Therefore, the proposed scheme is still secure even if the client's device is lost or stolen.

Impersonating attack:

Assume that an attacker wants to impersonate a legal user to access the digital content. Then, he/she may generate a forged $H(Anonymity\_ID \oplus X)$" and R_{DA}" to compute $S_{DA}" = H(H(Anonymity\_ID \oplus X)") \oplus R_{DA}"$ and sends $Anonymity\_ID$, $S_{DA}"$, and $C_{DA}"$ to CH, where $C_{DA}" = H(R_{DA}")$. However, this attack does not work because the CH will use $H(Anonymity\_ID \oplus X)$ to compute $R_{DA}^{'} = S_{DA}" \oplus H(H(Anonymity\_ID \oplus X))$. Finally, CH will find that $C_{DA}"$ is not equal to $C_{DA}^{'} = H(R_{DA}^{'})$. Therefore, the impersonating attack is impossible for the proposed scheme.

Performance analyses:

In Table 3, we show some comparisons among the proposed scheme, Zhang et al.'s scheme [12], and Chang et al.'s scheme [16]. According to Table 3, only the proposed scheme has no stolen-verifier attack. Unlike the other schemes, the proposed scheme can solve the security problem when the mobile devise is lost or stolen. Moreover, the proposed scheme provides the user anonymity and usage tracking.

Table 3 also shows the computation costs in the user's sides of these three schemes. The computation costs of the asymmetric encryption, symmetric encryption, one-way hash function and exclusive-or operation are denoted as A, S, H, and X, respectively. According to [18], the measurement of the above computation costs can be denoted as A>>>S>>>H>>>X, where "A>>>S" means that A is much larger than S. According to Table 3, the computation cost of the proposed scheme is lower than those of the other schemes. In conclusion, the proposed scheme is more efficient and practical than the other schemes.

Meth. / Comp.	[12]	[16]	Ours
User anonymity	Yes	No	Yes
Usage tracking	Yes	No	Yes
Stolen-verifier attack	Yes	Yes	No
Computation cost	3A+2S +3H	1S+(2|$F(N_i)$| +i+2)H+7X	1S+4H+3 X

Table 3. Comparisons of the related works

5 Conclusion

In this paper, we propose an efficient and practical E-DRM scheme with anonymity trust for mobile devices. According to our analysis, the proposed scheme has low computation cost so it is suitable for mobile devices. In addition, it protects the user's privacy by using the anonymous identity for the user authentication. Thus, the proposed scheme allows users access their digital contents by any permissive mobile devices. Besides, the proposed scheme provides usage tracking to make sure that the access is not abused by malicious users. Compared with the related works, the proposed scheme is more efficient and practical in the E-DRM applications for mobile devices. Based on the proposed scheme, we will investigate the usage charge for E-DRM applications to make our research more complete in the future.

Acknowledgement

This work was supported in part by National Science Council under the grants NSC 100-2221-E-424-006.

References

[1] Q. Liu, S. N. Reihaneh, and N. P. Sheppard, "Digital rights management for content distribution," *Proceedings of Australasian Information Security Workshop 2003 (AISW2003), Conferences in Research and Practice in Information Technology*, Adelaide, Australia., Vol. 21, 2003.

[2] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM*, Vol. 21, pp. 120-126, 1978.

[3] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Transactions on Information Theory*, Vol. 31, pp. 469-472, 1985.

[4] N. Koblitz, "Elliptic curve cryptosystem", *Mathematics of Computation*, Vol. 48, pp. 203-209, 1987.

[5] T. Kwon and J. Song, "Efficient key exchange and authentication protocols protecting weak secrets," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, Vol. E81-A, pp. 156-163, 1998.

[6] T. Kwon and J. Song, "Authenticated key exchange protocols resistant to password guessing attacks," *IEEE Proceedings Communications*, Vol. 145, pp. 304-308, 1998.

[7] M. Sandirigama, A. Shimizu, and M. T. Noda, "Simple and secure password authentication protocol (SAS)," *IEICE Transactions on Communications*, Vol. E83-B, pp. 1363-1365, 2000.

[8] M. S. Hwang, C. C. Lee, and Y. L. Tang, "A simple remote user authentication protocol," *Mathematical and Computer Modelling*, Vol. 36, pp. 103-107, 2002.

[9] H. Y. Chien and J. K. Jan, "Robust and simple authentication protocol," *Computer Journal*, Vol. 46, pp. 193-201, 2003.

[10] E. Bresson, O. Chevassut, A. Essiari, and D. Pointcheval, "Mutual authentication and group key agreement for low-power mobile devices," *Computer Communications*, Vol. 27, pp. 1730-1737, 2004.

[11] H. M. Sun and H. T. Yeh, "Password-based authentication and key distribution protocols with perfect forward secrecy," *Journal of Computer and System Sciences*, Vol. 72, pp. 1002-1011, 2006.

[12] J. Zhang, B. Li, L. Zhao, and S. Q. Yang, "License management scheme with anonymous trust for digital rights management," *Proceedings of* 2005 IEEE International Conference on Multimedia and Expo, Amsterdam, Netherlands, pp. 257-260, July 2005.

[13] C. C. Lin and P. H. Chiang, "A mobile trading scheme for digital content based on digital rights", *Proceedings of the Eighth International Conference on Intelligent Systems Design and Applications (ISDA 2008)*, Kaohsiung City, Taiwan, Vol. 3, pp. 451-456, Nov. 2008.

[14] C. L. Chen, "A secure and traceable E-DRM system based on mobile device," *Expert Systems with Applications*, Vol. 35, No. 3, pp. 878-886, Oct. 2008.

[15] C. C. Lin, S. C. Wu, P. H. Chiang, and C. C. Chen, "Enterprise-oriented digital rights management mechanism: eDRM," *Proceedings of International Conference on Availability, Reliability and Security*, ares, pp. 923-928, 2009.

[16] C. C. Chang, J. H. Yang, and, D. W. Wang, "An efficient and reliable E-DRM scheme for mobile environments," *Expert Systems with Applications*, Vol. 37, No. 9, pp. 6168-6176, Sep. 2010.

[17] D. Hankerson, A. Menezes, and S. Vanstone, Guide to elliptic curve cryptography, Springer-Verlag, New York, USA, 2004.

[18] B. Schneier, Applied Cryptography: Protocols, Algorithms, and Source Code in C, Second Edition, John Wiley & Sons, Inc., New York, USA, 1996.