Eland: An Efficient Lightweight Anonymous Authentication Protocol Applied to Digital Rights Management System

Qing Fan[1,2], Jianhua Chen[1*], Yihong Wen[3], Min Luo[4]

[1] School of Mathematics and Statistics, Wuhan University, China
[2] Guangxi Key Laboratory of Trusted Software, Guilin University of Electronic Technology, China
[3] The 54th Research Institute of China Electronics Technology Group Corporation, China
[4] School of Cyber Science and Engineering, Wuhan University, China
qingfan_Cassie@whu.edu.cn, chenjh_ecc@163.com, yihwen@139.com, mluo@whu.edn.cn

Abstract

A digital rights management (DRM) system provides the function of packaging raw content into an appropriate distribution form, protection for content temper-proof transmission and circumvention of unauthorized use. The digital rights contents in DRM system improve people's spiritual quality and create huge market value. However, in the sharing of digital contents, malicious adversaries could bring security and privacy issues even cause infringement of copyright. To solve these problems, valid user identity authentication and secret key distribution to legal users is an efficient method. Nevertheless, the existing authentication schemes for DRM system have problems of security weakness or efficiency to be enhanced. Thereby, we cryptanalyze these protocols' vulnerabilities and propose a new lightweight anonymous authentication protocol called Eland that can be efficiently applied to DRM system. We compare our protocol with six DRM-related authentication protocols from the aspects of security properties, computation expense and communication cost. Comparisons results indicate that our protocol achieves a well trade-off between security and performance. Specifically, our protocol reduces 37.5% communication cost and satisfies more security requirements than Yu et al.'s scheme.

Keywords: Digital rights management (DRM), Authentication, Key distribution, Anonymity

1 Introduction

According to explanation from World Intellectual Property Organization (WIPO), the digital rights management (DRM) refers to identification and description of property, rights involved in creation and enforcement of usage restrictions [1]. In a narrow sense, it includes techniques, tools and measures of digital rights protection during the digital content usage. Digital rights contents generally contain rights-related digital consumption products such as e-books, digital music, video, games and so on. The massive quality digital content has met the people's growing spiritual and cultural needs [2]. Value of DRM system is increasingly prominent and the DRM industry becomes one of the most promising formats. Mordor Intelligence report on DRM market indicates that the DRM market was valued at USD 3215.4 million in 2020, expected to exceed USD 6000 million by 2026 and increase at a compound annual growth rate (CAGR) of 12.18% over the forecast 2021-2026 [3].

In view of DRM operations, the digital rights content is encrypted using a symmetric key and stored in a content server. Meanwhile, the symmetric key and the rights object (i.e. state of permitted ways for digital content consumption) are packaged into a license held by the license server [4]. Generating process of ciphertexts and license is depicted in Figure 1. The user sends a license request information to the license server for developing the digital content's value. Not all the request could be accepted since the user may be illegal or malicious. Moreover, adversaries may also trace the user's behavior to steal user privacy. Furthermore, piracy statistics for 2021 shows that digital video piracy is costing US content and distribution sectors between USD 29.2 and USD 71.0 billion each year [5]. Therefore, security and privacy protection of content delivery is particularly significant.

Authentication, as a prominent cryptographic technique, is undoubtedly an efficient way to authenticate the user's legality. There exist some authenticated protocols applying in digital rights management system [6-9], some of which has improved previous schemes for better security or efficiency. However, these schemes whether the new proposed or enhanced schemes still have some vulnerabilities. For example, Rana and Mishra's [6] and Rana and Mishra's [7] are not equipped with three-factor security. Lee et al's [8] is resistless to denial of service attack (DoS) and does not provide secure authentication and three-factor security. The scheme [9] does not hold mobile device theft attack resistance, three-factor security and secure authentication. Yu et al.'s [10] further remedy scheme [9]'s limitation and proposed a lightweight three-factor authentication protocol.

Although the above schemes attempt to provide security and privacy protection in DRM system application, they cannot satisfy as many security properties as possible to the best of our knowledge. Moreover, the limited computing power and memory capacity of DRM mobile device [11] asks higher demand for protocol's computation efficiency and storage expense. Consequently, we present an efficient anonymous authentication protocol applied to DRM system which only uses hash function and XOR operations to realize lightweight.

*Corresponding Author: Jianhua Chen; E-mail: chenjh_ecc@163.com

DOI: 10.53106/160792642022032302007

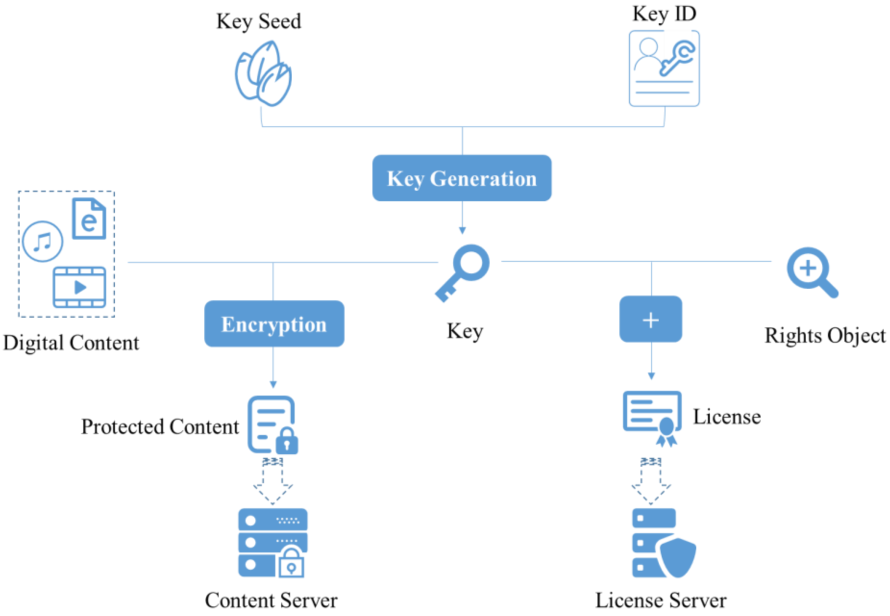

Figure 1. Generating process of license and protected content

Our Contributions.

- We propose a new authentication protocol named Eland, which only employs hash function and XOR operations to achieve lightweight anonymous authentication. We also apply it to DRM system and meet the system's security requirements.
- We compare the proposed protocol with six recent DRM-related authentication protocols. The security and performance comparison results show that our protocol realizes a better trade-off between efficiency and security. Thereby, our protocol is more efficient for digital contents protection.
- We prove our protocol achieves secure mutual authentication under the ROR model, that is we give theory proof of impersonation attack resistance and key distribution security.

Organization. Section 2 introduces DRM system model, fuzzy extractor, threat model and security requirements of DRM authentication protocol. Section 3 reviews Yu et al.'s protocol and analyzes weakness of this protocol. To realize a well trade-off between security and efficiency, we propose a new authentication and key distribution protocol in section 4. Its formal security proof and informal security analysis are provided in section 5. Section 6 compares the performance of existing six authentication schemes. We introduce related works in section 7. Finally, we conclude this paper in section 8.

2 Preliminaries

In this section, we introduce the digital rights management (DRM) system model including compositions and workflow, the biometrics information extraction tool i.e. fuzzy extractor, threat model of authentication protocol as well as security requirements of DRM system. Finally, we give notations used in this paper.

2.1 DRM System Model

According to [8-9, 12-13], DRM system is composed of four parties: the content provider, the content server, the license server and the user. The workflow of DRM can be seen in Figure 2.

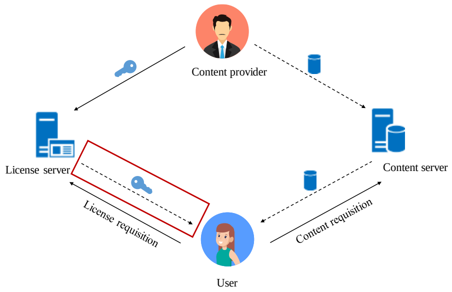

Figure 2. System model of DRM

- Content provider. Content provider is the creator and owner of the digital content who generates a symmetric encryption key K_{DC} corresponding to digital content. He will use K_{DC} to encrypt digital content and transmit the ciphertext to content server by a public channel. Meanwhile, the content provider sends K_{DC} with digital content identity ID_{DC} to the license server via a secure channel.
- Content server. The content server is responsible for encrypted digital content storage and publishes an abstract of digital content with the specific identity ID_{DC}.
- License server. The license server is in charge of secretly preserving the encrypted key K_{DC} and performing authentication to the user. Upon receiving (ID_{DC}, K_{DC}) secretly, the license server stores it in a secure database.
- User. The user is a demander of digital content and determines the access content ID_{DC} according to the showed abstract from content server. The user firstly sends an access requisition i.e. ID_{DC} to a license server through a secure channel. Having received the acknowledge message from license server, the user transmits the authentication messages for legality

verification and the corresponding key K_{DC} is finally distributed to the user. In the end, the user decodes the ciphertexts from content server and gets the intact digital content.

- Mobile device. The mobile device is held by the user which stores some authentication-related information such as auxiliary biometrics string, the hiding password information, authorized secret value from license server and so on. Information stored in mobile device will be used for user identity verification.

In this paper, we mainly consider the critical authentication and key distribution process between the user and license server.

2.2 Fuzzy Extractor

In this paper, we use the biometric tool fuzzy extractor to extract a certain secret value σ from biometric data BIO, where σ can be seen as a biometric key for user authentication and could be obtained with input an error-tolerant biometrics and an auxiliary string. Fuzzy extractor consists of **Gen** algorithm and **Rep** algorithm. **Gen** algorithm extracts two strings σ, θ and **Rep** algorithm recovers σ using an error-tolerant biometrics BIO' and θ. Details of fuzzy extractor are described as follows.

- **Gen** is a probabilistic algorithm and takes biometrics BIO as input. It is responsible for extracting a uniformly random string σ and an auxiliary string θ. This algorithm can be written as $(\sigma, \theta) \leftarrow Gen(BIO)$.
- **Rep** is a deterministic algorithm to reproduce the string σ with inputting biometrics BIO' and auxiliary string θ, where the distance between BIO' and BIO must keep within an allowable range. This algorithm can be written as $\sigma \leftarrow Rep(BIO', \theta)$.

2.3 Threat Model

Abdalla et al. [14] proposed the Real-Or-Random (ROR) model for two-party authenticated key exchange protocols. It demonstrates that ROR model provides more stronger security than BPR model and applied in many authenticated schemes [15-18]. According to [14, 17], the adversary could execute Send query, Execute query, Capture query, Reveal query and many-time $Test$ query.

- Hash(·): This query holds a hash list L_h with the form of (x, h_x). When $\mathcal{A}$ executes this query with message x, the oracle first checks whether x is in L_h and returns the corresponding h_x if it is. Otherwise, the oracle computes the hash value $h(\cdot, x)$ with x, adds $h(\cdot, x)$ into L_h and returns h_x to $\mathcal{A}$.
- $Execute(U_i, S_j, sid)$: This query models passive attacks where the adversary $\mathcal{A}$ could eavesdrop on all messages transmitted between U and S as protocol Π with session identifier sid. On receiving this query, oracle $\Pi_{U,S}^{sid}$ and $\Pi_{S,U}^{sid}$, respectively simulating U and S, will launch protocol Π. Then the messages exchanged between them will be recorded and transmitted to $\mathcal{A}$.
- $SendServer(U_i, S_j, sid, M')$: This query models active attacks against a server where $\mathcal{A}$ sends messages M' to server oracle $\Pi_{S,U}^{sid}$. Then the oracle will calculate a response honestly as Π and sends the response to $\mathcal{A}$.
- $SendUser(S_j, U_i, sid, M')$: This query models active attacks against a user where $\mathcal{A}$ sends messages M' to user oracle $\Pi_{U,S}^{sid}$. If $M' = \lambda$, the oracle will launch a new session and transmit authenticated request messages to $\mathcal{A}$. Otherwise, the oracle will execute Π to generate corresponding messages and send them to $\mathcal{A}$.
- $Capture(U_i)$: This query models user's mobile device theft attack in which all the values stored in the user mobile device will be captured by $\mathcal{A}$ through executing this query.
- $Reveal(\Pi, U_i, S_j, sid)$: This query models KEY exposure attack between user U_i and license server S_j. This query could only be executed when mutual authentication is realized. On receiving this query, the oracle will return KEY to $\mathcal{A}$.
- $Test(U_i, S_j, sid)$: This query can be executed by $\mathcal{A}$ after secure mutual authentication and the KEY has been shared with U_i. Then the oracle $\Pi_{U,S}^{sid}$ tosses a coin b. The oracle returns either a random one if $b = 0$ or the real session key if $b = 1$. The adversary could ask as many $Test$ queries as he wants.

In the end of above game, $\mathcal{A}$ outputs a guess bit b, Assuming the probability of $\mathcal{A}$ success is $Pr[Succ]$, we define advantage of $\mathcal{A}$ breaking the protocol Π is $Adv_{\Pi,\mathcal{A}}^{MA} = |2\,Pr[Succ] - 1|$.

Definition 1 (*Secure KEY distribution*): If there is no probabilistic polynomial time (PPT) adversary could correctly guesses b in the above game with non-negligible advantage $Adv_{\Pi,\mathcal{A}}^{MA}$, we say the proposed protocol is MA-secure.

Definition 2 (*Hash function*): A hash function $h: \{0,1\}^* \rightarrow \{0,1\}^n$ is a deterministic function that takes a variable length bit string as input and outputs a fixed length string of n bits. Collision-resistance and one-wayness of hash function are described as follows.

- One-wayness. Given a hash value $h(x)$, it is uncomputable for PPT adversary to find the x.
- Collision-resistance. Let $Adv_{\mathcal{A}}^{HASH}$ denote the advantage of PPT $\mathcal{A}$ finding a hash collision, i.e. $Adv_{\mathcal{A}}^{HASH} = Pr\,[(x_1, x_2): x_1 \neq x_2, h(x_1) \neq h(x_2)]$, where x_1, x_2 are randomly selected with arbitrary length. Then, $Adv_{\mathcal{A}}^{HASH}$ is negligible.

2.4 Security Requirements

According to [8-10, 19], an authentication protocol for DRM system has the following security requirements.

- **Impersonation attack resilience**: In the DRM system, it requires that no illegal user could impersonate the legitimate client to obtain digital rights content, which is an essential demand for digital rights content protection. Thus, the authenticated and key distribution should hold impersonation attack resilience.
- **Mobile device theft attack resilience**: Each user of DRM system has a mobile device to store secret information and execute necessary operations. Even if the private data in the mobile device is theft, the adversary still cannot acquire other information such as password and secret string from biometrics to obtain digital rights content access eventually.

- **Offline password guessing attack resilience**: This attack is that the adversary or malicious server attempts to guess the user password in the offline manner. Password exposure may result in other private information leakage even exposing digital rights content. Thereby, offline password guessing attack resilience is important for authentication protocol of DRM system.
- **Replay attack resilience**: This attack is that the adversary could take previous authentication messages in the public channel for current protocol execution, which cannot be found by the server. Replay attack may bring resource waste of license server and should be resisted in the authenticated and key distribution for DRM.
- **Stolen verifier attack resilience**: This attack could happen in the protocol registration phase. If the server holds a verifier table, the adversary may acquire user authenticated information by intruding into the server database. An efficient authentication scheme needs to keep stolen verifier attack resilience.
- **Privileged insider attack resilience**: This attack mainly takes place in the user registration phase. Privileged insider in the server side may acquire more information to break the protocol. Privilege insider attack resilience requires this breaking will not happen.
- **Server spoofing attack resilience**: Server spoofing attack means the adversary impersonates server to spoof users. This attack could bring perplex to the user's regular work. Thereby, server spoofing attack resilience is a necessary security requirement for DRM system.
- **Denial of service attack resilience**: This attack is that an adversary tries to make service resource unavailable to intended users by disrupting services of a client connected to the Internet. It is usually achieved by flooding the targeted server with overloaded authentication requests to prevent some or all legitimate requests from being fulfilled. To guarantee regular digital rights content access, DRM system should have denial of service attack resilience.
- **Three-factor security**: In the biometrics-based authenticated scheme, the smart device, biometrics and password consist of three factors to guarantee the scheme's security. Three-factor security requires that any two of factors are exposed cannot lead to the other one factor's leakage. For example, the adversary could not guess the user's password even if theft of smart device and biometrics.
- **Secure mutual authentication**: In the DRM system, we assume that the license server is honest and trustworthy. Therefore, authentication to the user who tries to obtain KEY of digital rights content is important. Secure authentication in this paper demands any possible attacks from the user side should be prevented.
- **Anonymity**: In the digital rights content access process, the user always wants to hide his true identity and the requested digital rights content in case the adversary maliciously track the user's behavior. For protecting the user's identity and behavior privacy, anonymity of authentication protocol needs to be held.
- **Un-traceability**: To hide the digital rights access footprint, the user may expect each authentication and digital rights content access is independent, that is the adversary cannot find relationship between any two authentication phases. Un-traceability could protect the user's privacy from statistical analysis and is another security requirement for DRM system.

Notations used in this paper is explained in Table 1.

Table 1. Notations

Notations	Interpretation
U_i	The *i*th mobile user
LS_j	License Server
ID_i	Identity of U_i
ID_{DC}	Identity of digital content
PW_i	U_i's password
BIO_i	Biometrics of U_i
r_1	Random nonce of U_i
r_2	Random nonce of LS_j
x_{LS}	Master key of LS_j
KEY_{DC}	Secret key of digital content
$h(\cdot)$	One-way hash function
$\oplus$	Bitwise XOR operation
$\|\|$	Concatenation operation

3 Review of Yu et al.'s Protocol

In this section, Yu et al.'s three-factor authentication protocol will be reviewed briefly. Their protocol consists of user registration, login and authentication, and password change. Each phase of Yu et al.'s protocol is presented in the following subsection.

3.1 Yu et al.'s Protocol

• *Registration phase:* A mobile user U_i registers its identity with license server LS_j as the following steps.

- U_i chooses ID_i, PW_i and biometrics BIO_i. Then U_i calculates $Gen(BIO_i) = (\sigma_i, \theta_i), RPW_i = h(PW_i||\sigma_i)$ and transmits $\{ID_i, RPW_i\}$ to LS_j through a secure channel.
- On receiving the request information from U_i, LS_j calculates $X_i = h(ID_i||x_{LS}), d_i = X_i \oplus h(ID_i|||RPW_i)$ and $f_i = h(RPW_i||X_i)$. Then LS_j stores ID_i, X_i in a database and transmits $\{d_i, f_i\}$ to U_i via a secure channel.
- Having received messages from LS_j, U_i stores $\{d_i, f_i\}$ in a device.

• *Authentication and key distribution phase:* The goal of this phase is to realize mutual authentication between U_i and LS_j, and distribute decryption key KEY_{DC} to a legitimate user, which can be seen in Figure 3.

- U_i inputs ID_i, PW_i, BIO_i, computes $\sigma_i = Rep(BIO_i, \theta_i), RPW_i = h(PW_i||\sigma_i), X_i = d_i \oplus h(ID_i||RPW_i)$ and verifies whether $h(RPW_i||X_i)$ is equal to f_i in the device. Then U_i randomly selects a nonce r_1 and calculates $M_1 = X_i \oplus r_1$, $M_2 = ID_i \oplus r_1$, $M_3 = ID_{DC} \oplus r_1$, $M_{US} = h(ID_i||ID_{DC}||X_i||r_1)$. Finally, U_i transmits $\{M_1, M_2, M_3, M_{US}\}$ to the LS_j.
- On receiving $\{M_1, M_2, M_3, M_{US}\}$, LS_j calculates $r_1 = M_1 \oplus X_i, ID_i = M_2 \oplus r_1, ID_{DC} = M_3 \oplus r_1$ and verifies $M_{US}^* = h(ID_i||ID_{DC}||X_i||r_1)$. Then LS_j retrieves secret key KEY_{DC} according to ID_{DC} and calculates $M_4 = r_2 \oplus X_i, M_5 = KEY_{DC} \oplus X_i, M_{SU} =$

$h(ID_i||X_i||KEY_{DC}||r_2)$. Eventually, LS_j transmits $\{M_4, M_5, M_{SU}\}$ to U_i.

- On receiving $\{M_4, M_5, M_{SU}\}$, U_i recovers $r_2 = M_4 \oplus X_i, KEY_{DC} = M_5 \oplus X_i$ and verifies $h(ID_i||X_i||KEY_{DC}||r_2) = M_{SU}$. If the equation holds, U_i stores the secret key KEY_{DC} and has the access right to the digital content.

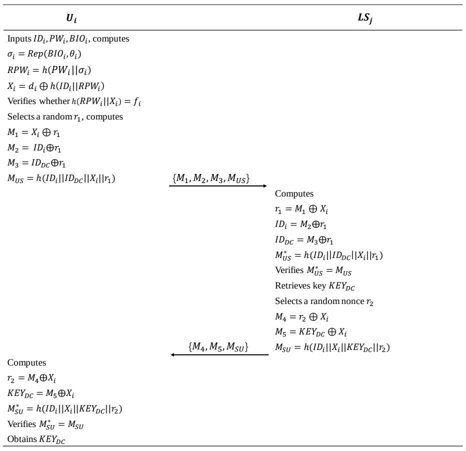
U_i
LS_j
Inputs ID_i, PW_i, BIO_i, computes
$\sigma_i = Rep(BIO_i, \theta_i)$
$RPW_i = h(PW_i||\sigma_i)$
$X_i = d_i \oplus h(ID_i||RPW_i)$
Verifies whether $h(RPW_i||X_i) = f_i$
Selects a random r_1, computes
$M_1 = X_i \oplus r_1$
$M_2 = ID_i \oplus r_1$
$M_3 = ID_{DC} \oplus r_1$
$M_{US} = h(ID_i||ID_{DC}||X_i||r_1)$
$\{M_1, M_2, M_3, M_{US}\}$
Computes
$r_1 = M_1 \oplus X_i$
$ID_i = M_2 \oplus r_1$
$ID_{DC} = M_3 \oplus r_1$
$M_{US}^* = h(ID_i||ID_{DC}||X_i||r_1)$
Verifies $M_{US}^* = M_{US}$
Retrieves key KEY_{DC}
Selects a random nonce r_2
$M_4 = r_2 \oplus X_i$
$M_5 = KEY_{DC} \oplus X_i$
$M_{SU} = h(ID_i||X_i||KEY_{DC}||r_2)$
$\{M_4, M_5, M_{SU}\}$
Computes
$r_2 = M_4 \oplus X_i$
$KEY_{DC} = M_5 \oplus X_i$
$M_{SU}^* = h(ID_i||X_i||KEY_{DC}||r_2)$
Verifies $M_{SU}^* = M_{SU}$
Obtains KEY_{DC}

Figure 3. Authentication and key distribution phase of Yu et al.'s protocol

- *Private information change phase:* This phase is performed offline. A legitimate user U_i could change password and imprinted biometrics as follows in any cases.

- U_i takes ID_i and the present BIO_i, PW_i as inputs and calculates $Gen(BIO_i) = (\sigma_i, \theta_i)$, $RPW_i = h(PW_i||\sigma_i)$. Then $\{ID_i, RPW_i\}$ are sent to the device.

- The device calculates $X_i = d_i \oplus h(ID_i||RPW_i)$, checks $f_i \overset{?}{=} h(RPW_i||X_i)$ and aborts if the check fails. Otherwise, the mobile device transmits a permitting message to U_i.

- On receiving the permitting message, U_i chooses a new password PW_i^{new} and a new biological information BIO_i^{new} to compute $Gen(BIO_i^{new}) = (\sigma_i^{new}, \theta_i^{new})$, $RPW_i^{new} = h(PW_i^{new}||\sigma_i^{new})$. Then the RPW_i^{new} is sent to the mobile device.

- Eventually, the mobile device computes $d_i^{new} = X_i \oplus h(ID_i||RPW_i^{new})$, $f_i^{new} = h(RPW_i^{new}||X_i)$ and replaces $\{d_i, f_i\}$ with $\{d_i^{new}, f_i^{new}\}$.

3.2 Security Analysis of Yu et al.'s Protocol

In this section, we analyze that a malicious attacker $\mathcal{A}$ could impersonate a legitimate user U_i by computing the login request message of U_i. $\mathcal{A}$ could intercept the valid messages $\{M_1, M_2, M_3, M_{US}\}$ from the public channel. $\mathcal{A}$ could guess the correct random nonce r_1 by verifying the equation $h(M_2 \oplus r_1||M_3 \oplus r_1||M_1 \oplus r_1||r_1) \overset{?}{=} M_{US}$. Having gotten r_1, $\mathcal{A}$ can get $X_i = M_1 \oplus r_1, ID_i = M_2 \oplus r_1, ID_{DC} = M_3 \oplus r_1$. Then $\mathcal{A}$ could achieve impersonation attack even break properties of secure mutual authentication and user anonymity.

- **Impersonation attack:** $\mathcal{A}$ could impersonate U_i and obtains secret key KEY_{DC} as follows.

- Step 1: $\mathcal{A}$ randomly chooses a nonce r_a and calculates $M_1 = X_i \oplus r_a, M_2 = ID_i \oplus r_a, M_3 = ID_{DC} \oplus r_a$ and $M_{US} = h(ID_i||ID_{DC}||X_i||r_a)$. Then $\mathcal{A}$ transmits $\{M_1, M_2, M_3, M_{US}\}$ to LS_j.

- Step 2: Upon receiving $\{M_1, M_2, M_3, M_{US}\}$, LS_j first recovers $r_a = M_1 \oplus X_i, ID_i = M_2 \oplus r_a$, and $ID_{DC} = M_3 \oplus r_a$. Then LS_j computes $M_{US}^* = h(ID_i||ID_{DC}||X_i||r_a)$ and verifies the equation $M_{US}^* \overset{?}{=} M_{US}$. If it doesn't hold, aborts; otherwise, LS_j produces a nonce r_s and calculates $M_4 = r_s \oplus X_i, M_5 = KEY_{DC} \oplus X_i$ and $M_{SU} = h(ID_i||X_i||KEY_{DC}||r_s)$. Eventually, LS_j transmits $\{M_4, M_5, M_{SU}\}$ to $\mathcal{A}$.

- Step 3: On receiving $\{M_4, M_5, M_{SU}\}$, $\mathcal{A}$ computes $r_s = M_4 \oplus X_i$ and gets $KEY_{DC} = M_5 \oplus X_i$. Then $\mathcal{A}$ verifies the equation $h(ID_i||X_i||KEY_{DC}||r_s) \overset{?}{=} M_{SU}$. If it holds, $\mathcal{A}$ obtains the correct secret key of digital content.

- **Secure mutual authentication:** Yu et al.'s protocol cannot hold secure mutual authentication since the adversary could forge valid authenticated messages $\{M_1, M_2, M_3, M_{US}\}$ while the license server cannot find. The detailed attack process is the same as that of impersonation attack.

- **User anonymity:** As analyzed in the first paragraph of this section, $\mathcal{A}$ can require the random number r_1 used in the authentication. Having obtained r_1, $\mathcal{A}$ can find the user's real identity ID_i by computing $M_2 \oplus r_1$. Therefore, user anonymity of Yu et al.'s protocol does not hold.

- **Stolen verifier attack:** In the registration phase of Yu et al.'s protocol, the license server stores identity ID_i with corresponding secret value X_i into a database. Once $\mathcal{A}$ steals the verifier, he can use X_i to generate valid authenticated messages for U_i, i.e. randomly chooses a nonce r' to compute $M_1' = X_i \oplus r', M_2 = ID_i \oplus r', M_3 = ID_{DC} \oplus r', M_{US} = h(ID_i||ID_{DC}||X_i||r')$. Thus, their protocol does not have stolen verifier attack resilience.

- **Replay attack:** The adversary $\mathcal{A}$ could intercept messages $\{M_1, M_2, M_3, M_{US}\}$ in the public channel and retransmit them to the license server which can be successfully verified by the server. Therefore, Yu et al.'s protocol cannot resist replay attack.

- **Vulnerability to three-factor security:** If the adversary $\mathcal{A}$ gets the user's device and biometrics that is d_i, f_i and σ_i are obtained, an equation relationship could be established by $\mathcal{A}$ to guess the user's low-entropy information. In other words, $\mathcal{A}$ could verify whether f_i equals to $h(h(PW_i'||\sigma_i)||d_i \oplus h(ID_i'||h(PW_i'||\sigma_i)))$ holds to guess the user's password PW_i' and identity ID_i'.

4 The Proposed Protocol

In this section, a more secure lightweight authentication protocol for the DRM system is proposed to resist the impersonation attack. The proposed protocol comprises three phases: registration phase, login and authentication phase, and password change phase.

4.1 Registration Phase

Each user holds a device to execute necessary computation and store secret information. Through the device, a user could complete the authentication protocol for DRM system. A new user U_i who tries to access to digital content will first register the identity ID_i through the license server LS_j. Registration phase of the proposed protocol is shown in Figure 4. Detailed descriptions are as follows.

- U_i inputs the identity ID_i, password PW_i and imprints the biometrics BIO_i into the device. The device could obtain σ_i and θ_i through $Gen(BIO_i) = (\sigma_i, \theta_i)$. Then the device selects a random nonce r_i and computes $d_i = h(PW_i||\sigma_i||ID_i), RPW_i = d_i \oplus r_i$. Finally, U_i transmits $\{ID_i, r_i, d_i\}$ to LS_j through a secure channel.
- On receiving $\{ID_i, r_i, d_i\}$, LS_j calculates $X_i = h(ID_i||d_i||x_{LS}||r_i), f_i = h(d_i||r_i) \oplus X_i$ and sends f_i to U_i through a secure channel.
- Eventually, U_i stores $\{\theta_i, RPW_i, f_i\}$ in the device.

4.2 Authentication and Key Distribution Phase

Having received a confirmation message of ID_{DC} from LS_j over a secure channel, the user U_i will transmit a login request to the corresponding license server LS_j. Then LS_j authenticates U_i and sends the secret key KEY_{DC} to U_i under the premise of U_i legitimate. As Figure 5, the detailed login and authentication steps are described in the following.

- U_i inputs the identity ID_i, password PW_i^* and biometrics BIO_i^* into the device. Then the device uses BIO_i^* and the stored θ_i to compute $\sigma_i^* = Rep(BIO_i^*, \theta_i)$ and calculates $d_i^* = h(PW_i^*||\sigma_i^*||ID_i), r_i^* = d_i^* \oplus RPW_i, X_i^* = h(d_i^*||r_i^*) \oplus f_i$. Then it chooses a random r_1 and computes $M_1 = (d_i^*||r_i^*) \oplus h(ID_{DC}), M_2 = X_i \oplus (ID_i||r_1), M_3 = (ID_i||ID_{DC}) \oplus h(X_i||r_1||T_1)$ and $M_{US} = h(ID_i||ID_{DC}||d_i^*||r_i^*||X_i^*||r_1||T_1)$, where T_1 is the current timestamp. Eventually, U_i sends $\{T_1, M_1, M_2, M_3, M_{US}\}$ to LS_j.
- Upon receiving $\{T_1, M_1, M_2, M_3, M_{US}\}$ at the time T_2, LS_j first checks whether $|T_2 - T_1| \leq \Delta$ and aborts if the check fails. Otherwise, LS_j computes $(d_i||r_i) = M_1 \oplus h(ID_{DC}), X_i = h(d_i||x_{LS}||r_i)$ and gets $(ID_i||r_1) = M_2 \oplus X_i$, $(ID_i||ID_{DC}) = M_3 \oplus h(X_i||r_1||T_1)$. Then LS_j verifies whether the equation $M_{US} = h(ID_i||ID_{DC}||d_i||r_i||X_i||r_1||T_1)$ holds. It aborts if the equation does not hold. Otherwise, the secret key KEY_{DC} to the digital content is retrieved. Finally, LS_j chooses a random nonce r_2 to compute $M_4 = X_i \oplus r_2, M_5 = h(X_i||r_1||r_2) \oplus KEY_{DC}$, $M_{SU} = h(r_2||ID_i||X_i||ID_{DC}||KEY_{DC})$ and transmits $\{M_4, M_5, M_{SU}\}$ to U_i.
- On receiving $\{M_4, M_5, M_{SU}\}$, U_i calculates $r_2 = M_4 \oplus X_i$ and $KEY_{DC} = M_5 \oplus h(X_i||r_1||r_2)$. Then he checks the equation $M_{SU} \overset{?}{=} h(r_2||ID_i||X_i||ID_{DC}||KEY_{DC})$. If the equation holds, U_i obtains the correct secret key KEY_{DC} and has access authority to the digital content.

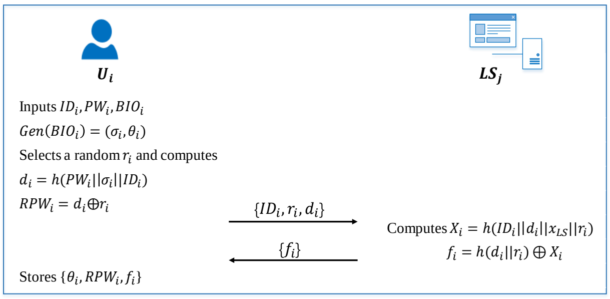
U_i
LS_j
Inputs ID_i, PW_i, BIO_i
$Gen(BIO_i) = (\sigma_i, \theta_i)$
Selects a random r_i and computes
$d_i = h(PW_i||\sigma_i||ID_i)$
$RPW_i = d_i \oplus r_i$
$\{ID_i, r_i, d_i\}$
Computes $X_i = h(ID_i||d_i||x_{LS}||r_i)$
$f_i = h(d_i||r_i) \oplus X_i$
$\{f_i\}$
Stores $\{\theta_i, RPW_i, f_i\}$

Figure 4. Registration phase

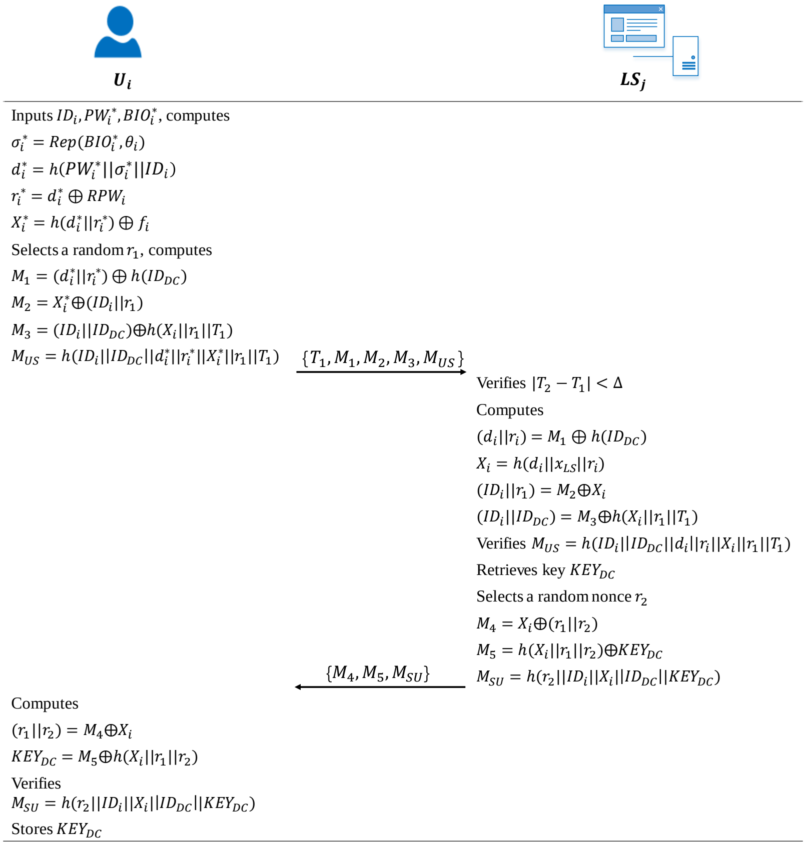
U_i
LS_j
Inputs ID_i, PW_i^*, BIO_i^*, computes
$\sigma_i^* = Rep(BIO_i^*, \theta_i)$
$d_i^* = h(PW_i^*||\sigma_i^*||ID_i)$
$r_i^* = d_i^* \oplus RPW_i$
$X_i^* = h(d_i^*||r_i^*) \oplus f_i$
Selects a random r_1, computes
$M_1 = (d_i^*||r_i^*) \oplus h(ID_{DC})$
$M_2 = X_i^* \oplus (ID_i||r_1)$
$M_3 = (ID_i||ID_{DC}) \oplus h(X_i||r_1||T_1)$
$M_{US} = h(ID_i||ID_{DC}||d_i^*||r_i^*||X_i^*||r_1||T_1)$
$\{T_1, M_1, M_2, M_3, M_{US}\}$
Verifies $|T_2 - T_1| < \Delta$
Computes
$(d_i||r_i) = M_1 \oplus h(ID_{DC})$
$X_i = h(d_i||x_{LS}||r_i)$
$(ID_i||r_1) = M_2 \oplus X_i$
$(ID_i||ID_{DC}) = M_3 \oplus h(X_i||r_1||T_1)$
Verifies $M_{US} = h(ID_i||ID_{DC}||d_i||r_i||X_i||r_1||T_1)$
Retrieves key KEY_{DC}
Selects a random nonce r_2
$M_4 = X_i \oplus (r_1||r_2)$
$M_5 = h(X_i||r_1||r_2) \oplus KEY_{DC}$
$\{M_4, M_5, M_{SU}\}$
$M_{SU} = h(r_2||ID_i||X_i||ID_{DC}||KEY_{DC})$
Computes
$(r_1||r_2) = M_4 \oplus X_i$
$KEY_{DC} = M_5 \oplus h(X_i||r_1||r_2)$
Verifies
$M_{SU} = h(r_2||ID_i||X_i||ID_{DC}||KEY_{DC})$
Stores KEY_{DC}

Figure 5. Authentication and key distribution phase

4.3 Private Information Change Phase

In this phase, we suppose that U_i requests to change the secret password PW_i and biological information BIO_i. This phase is executed by the user U_i and the mobile device with the aid of license server LS_j. The detailed steps are described in the following, which is depicted in Figure 6.

- U_i inputs identity ID_i, the present password PW_i^*, biometrics BIO_i^* and the new password PW_i^{new}, biometrics BIO_i^{new} into the device. Then the device computes $\sigma_i^* = Rep(BIO_i^*, \theta_i)$, $, d_i^* = h(PW_i^* || \sigma_i^* || ID_i)$, $r_i^* = RPW_i \oplus d_i^*$ and $X_i^* = h(d_i^* || r_i^*) \oplus f_i$. Then it selects a new random s_i and computes $Gen(BIO_i^{new}) = (\sigma_i^{new}, \theta_i^{new})$ $, d_i^{new} = h(PW_i^{new} || \sigma_i^{new} || ID_i)$, $RPW_i^{new} = d_i^{new} \oplus s_i$. Finally, the device transmits $\{ID_i, r_i^*, d_i^*, X_i^*, s_i, d_i^{new}\}$ to the LS_j via a secure channel.
- On receiving messages from U_i, LS_j computes $X_i = h(ID_i || d_i^* || x_{LS} || r_i^*)$ and verifies the equation $X_i = X_i^*$. If it does not hold, LS_j refuse the U_i's request. Otherwise, it computes $X_i^{new} = h(ID_i || d_i^{new} || x_{LS} || s_i)$ and $f_i^{new} = h(d_i^{new} || s_i) \oplus X_i^{new}$. Eventually, LS_j sends f_i^{new} to U_i. Upon receiving $\boldsymbol{f_i^{new}}$, the mobile device stores $\{\boldsymbol{\theta_i^{new}}, \boldsymbol{RPW_i^{new}}, \boldsymbol{f_i^{new}}\}$ securely.

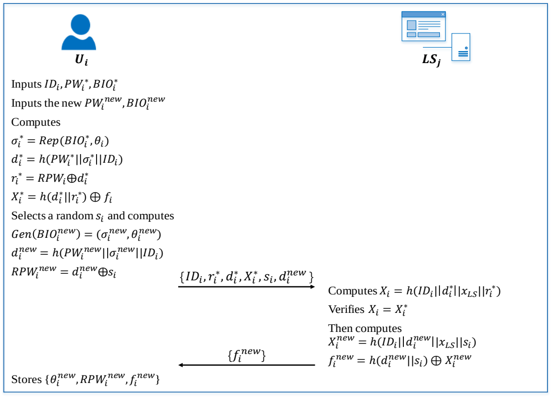
U_i
LS_j
Inputs ID_i, PW_i^*, BIO_i^*
Inputs the new PW_i^{new}, BIO_i^{new}
Computes
$\sigma_i^* = Rep(BIO_i^*, \theta_i)$
$d_i^* = h(PW_i^* || \sigma_i^* || ID_i)$
$r_i^* = RPW_i \oplus d_i^*$
$X_i^* = h(d_i^* || r_i^*) \oplus f_i$
Selects a random s_i and computes
$Gen(BIO_i^{new}) = (\sigma_i^{new}, \theta_i^{new})$
$d_i^{new} = h(PW_i^{new} || \sigma_i^{new} || ID_i)$
$RPW_i^{new} = d_i^{new} \oplus s_i$
$\{ID_i, r_i^*, d_i^*, X_i^*, s_i, d_i^{new}\}$
Computes $X_i = h(ID_i || d_i^* || x_{LS} || r_i^*)$
Verifies $X_i = X_i^*$
Then computes
$X_i^{new} = h(ID_i || d_i^{new} || x_{LS} || s_i)$
$f_i^{new} = h(d_i^{new} || s_i) \oplus X_i^{new}$
$\{f_i^{new}\}$
Stores $\{\theta_i^{new}, RPW_i^{new}, f_i^{new}\}$

Figure 6. Private information change phase

5 Security Analysis

In this section, we firstly give the proof of Eland's MA-security under the security model in section 2.3. Then we explain that Eland protocol satisfies the security requirements mentioned in section 2.4.

5.1 Formal Security Proof

Theorem 1. Assume that the PPT adversary $\mathcal{A}$ running in time t under the ROR model and PD is a proper password dictionary, ρ is the bit length of the biometrics key σ_i, the advantage of $\mathcal{A}$ in breaking the MA-secure of protocol Π is estimated as $Adv_{\Pi,\mathcal{A}}^{MA} \leq \frac{q_h^2}{2|HASH|} + \frac{q_{send}}{2^{\rho-1} \cdot |PD|}$ where $q_h, q_{send}, |PD|$ and $|HASH|$ denote the number of Hash queries, Send queries, the size of PD and the range of $h(\cdot)$ respectively.

Proof. We follow the similar proof as . In this proof, four games denoted by $G_j (j = 0,1,2,3)$ are considered where the success probability is $Pr[Succ_j]$ for guessing the right bit b. Descriptions of these games are given as follows.

G_0: G_0 denote a real attack executed by $\mathcal{A}$ against the proposed protocol Π under ROR model, where $\mathcal{A}$ tries to guess the correct bit b to break MA-secure. Suppose that probability of $\mathcal{A}$ succeeding in this game is $Pr[Succ_0]$. According to definition of advantage, there exists the following equation:

$$Adv_{\Pi,\mathcal{A}}^{MA} = |2Pr[Succ_0] - 1| \tag{1}$$

G_1: This game simulates an eavesdropping attack executed by eavesdropper, who can intercept $Msg_1 = \{T_1, M_1, M_2, M_3, M_{US}\}$ and $Msg_2 = \{M_4, M_5, M_{SU}\}$ in the public channel. In other words, $\mathcal{A}$ in ROR model could make *Execute* query. At the end of the game, $\mathcal{A}$ executes *Test* query which outputs the actual KEY or a random string according to bit b. Note that KEY is computed as $KEY_{DC} = M_5 \oplus h(X_i || r_1 || r_2)$ where $X_i = h(h(ID_i || PW_i || \sigma_i) || x_{LS} || r_i)$. To compute KEY, $\mathcal{A}$ must have M_5 and $h(X_i || r_1 || r_2)$, which further needs to know the master x_{LS}, password PW_i and biometrics key σ_i. $\mathcal{A}$ also requires ID_i, r_1, r_2 which are secret to $\mathcal{A}$. Therefore, probability of $\mathcal{A}$ succeeding in the game G_1 is not increased by eavesdropping attack, that is

$$Pr[Succ_1] = Pr[Succ_0] \tag{2}$$

G_2: Through adding *Send* query and *Hash(·)* query to G_1, it is transformed into G_2 which simulates the active attack. In this game, $\mathcal{A}$ tries to deceive an honest participant to accept modified messages. Specifically, $\mathcal{A}$ is allowed to execute several *Hash(·)* queries to check whether there exists hash collisions. Moreover, $\mathcal{A}$ may perform two kinds of *Send* queries i.e. $SendServer(U_i, S_j, sid, M)$ query to server oracle and $SendUser(\ S_j, U_i, sid, M\ )$ query to user oracle. Any changes on $\{T_1, M_1, M_2, M_3, M_{US}\}$ and $\{M_4, M_5, M_{SU}\}$ during authentication and key distribution phase contain random numbers, timestamp and user's identity. As a consequence, there does not exist hash collision when $\mathcal{A}$ queries *Send* oracle. According to birthday paradox, we have

$$|Pr[Succ_1] - Pr[Succ_2]| \leq q_h^2/(2|HASH|) \tag{3}$$

G_3: Through adding $Capture(\ U_i\ )$ query to G_2, it is transformed into G_3 which simulates the mobile device theft attack. In this game, $\mathcal{A}$ may perform dictionary attack through using the information stored in user device to guess the low-entropy password. In addition, $\mathcal{A}$ may attempt to get biometrics key σ_i according to the stored information in device. We use fuzzy extractor and biometrics BIO to uniformly extract random ρ-bit biometrics key $\sigma_i \in \{0,1\}^\rho$. Then the probability of $\mathcal{A}$ in guessing the correct σ_i is approximately $\frac{1}{2^\rho}$. Moreover, the system permits the limited number of wrong password entries. Thereby, we have

$$|Pr[Succ_2] - Pr[Succ_3]| \leq q_{send}/(2^\rho \cdot |PD|) \tag{4}$$

According to equation (2-4), we have

$$
\begin{aligned}
& |Pr[Succ_1] - Pr[Succ_3]| \\
\leq \ & |Pr[Succ_1] - Pr[Succ_2]| + |Pr[Succ_2] - Pr[Succ_3]| \\
\leq \ & q_h^2/(2|HASH|) + q_{send}/(2^{\rho} \cdot |PD|)
\end{aligned}
$$

Combining with equation (1) and $Pr[Succ_3] = \frac{1}{2}$, we have

$$Adv_{\Pi,\mathcal{A}}^{MA} \leq \frac{q_h^2}{2|HASH|} + \frac{q_{send}}{2^{\rho-1} \cdot |PD|} \quad \textbf{(5)}$$

5.2 Informal Security Analysis

Next, we show that the Eland protocol is equipped with the system's security requirements as section 2.4.

- **Impersonation attack resilience:** The adversary $\mathcal{A}$ attempting to impersonate a user U_i must produce valid authentication messages $\{M_1, M_2, M_3, M_{US}\}$ where natural secret values are $\{PW_i, \sigma_i, X_i, r_i, ID_i, ID_{DC}\}$. $\mathcal{A}$ could not recover any one of these secret value neither by establishing equation utilizing $\{M_1, M_2, M_3, M_{US}\}$ nor through breaking security of hash function $h(\cdot)$. Thus, $\mathcal{A}$ cannot forge valid authenticated messages and successfully deceive license server. Thereby, the proposed Eland protocol resists impersonation attack.
- **Mobile device theft attack resilience:** The user stores $\{\theta_i, RPW_i, f_i\}$ in the mobile device where θ_i is an auxiliary string, RPW_i is $d_i \oplus r_i$, r_i is a random number, f_i is $h(d_i||r_i) \oplus X_i$. Even if $\mathcal{A}$ acquires $\{\theta_i, RPW_i, f_i\}$, he still cannot get ID_i, PW_i, σ_i due to secrecy of master key x_{LS}. Thereby, the proposed protocol holds mobile device theft attack resilience.
- **Replay attack resilience:** Suppose that $\mathcal{A}$ intercepts messages $\{T_1, M_1, M_2, M_3, M_{US}\}$ and replaces T_1 with the new timestamp T_1'. Then he attempts to impersonate U_i by replaying $\{T_1', M_1, M_2, M_3, M_{US}\}$ to LS_j. LS_j could find the attack by checking $M_{US} = h(ID_i||ID_{DC}||d_i||r_i||X_i||r_1||T_1')$ since the M_{US} with the new timestamp is different for each authentication. Therefore, the proposed protocol withstands replay attack.
- **Denial of service attack resilience:** According to the denial of service, $\mathcal{A}$ generally produces superfluous authenticated messages to LS_j which results in overloaded computation. LS_j first verifies whether the timestamp is new enough which is the first step to hinder needless computation. Then he verifies M_{US} to avoid more useless calculated load. These two steps shows the proposed protocol does not have denial of service.
- **Stolen verifier attack resilience:** As shown in Figure 4, LS_j does not keep a verification table for the following authentication. Therefore, $\mathcal{A}$ cannot break our protocol through stealing verifier.
- **Privileged insider attack resilience:** In the registration phase, the user transmits ID_i, r_i and $d_i = h(PW_i||\sigma_i||ID_i)$ to the server. Then the privileged insider cannot require PW_i from d_i because he does not know the biometrics information σ_i. Thereby, the proposed protocol could resist privileged insider attack.
- **Server spoofing attack resilience:** To impersonate LS_j, $\mathcal{A}$ has to produce a correct authenticated message $M_{SU} = h(r_2||ID_i||X_i||ID_{DC}||KEY_{DC})$. Due to $\mathcal{A}$ cannot obtain server's master key x_{LS} and $h(\cdot)$ is a secure hash function, he could not acquire $X_i = h(d_i||x_{LS}||r_i)$. Therefore, $\mathcal{A}$ could not generate a valid M_{SU} and the proposed protocol resists server spoofing attack.
- **Anonymity:** As shown in Figure 5, U_i's identity is included in $M_2 = X_i \oplus (ID_i||r_1), M_3 = (ID_i||ID_{DC}) \oplus h(X_i||r_1||T_1), M_{US} = h(ID_i||ID_{DC}||d_i||r_i||X_i||r_1||T_1)$, where X_i is $h(h(ID_i||PW_i||\sigma_i)||x_{LS}||r_i)$. Due to secrecy of PW_i, σ_i, r_i, x_{LS}, $\mathcal{A}$ cannot compute X_i to get ID_i. Therefore, the Eland protocol holds user anonymity.
- **Offline password guessing attack resilience:** The adversary $\mathcal{A}$ may collect M_2, M_3, M_{US} to guess password PW_i which is included in $X_i = h(h(ID_i||PW_i||\sigma_i)||x_{LS}||r_i)$. In order to derive PW_i, $\mathcal{A}$ must know the master key x_{LS}, biometrics σ_i, the random number r_i and the user's real identity ID_i. Due to user anonymity and only legitimate user/server could have $(\sigma_i, r_i)/x_{LS}$, $\mathcal{A}$ cannot obtain these secret values to guess the password PW_i.
- **Secure mutual authentication:** The above analysis indicates that $\mathcal{A}$ could neither impersonate a legitimate user nor replace license server by any type of attacks. Therefore, this protocol provides secure mutual authentication.
- **Three-factor security:** As shown in Figure 4, the smart device stores $\{\theta_i, RPW_i, f_i\}$ where $f_i = h(d_i||RPW_i \oplus d_i) \oplus h(ID_i||d_i||x_{LS}||RPW_i \oplus d_i), d_i = h(PW_i||\sigma_i||ID_i)$. Even if user's biometrics and smart device are exposed to $\mathcal{A}$, he still cannot guess correct password due to secrecy of master key x_{LS}. Similarly, if $\mathcal{A}$ acquires password PW_i and biometrics information σ_i, he still cannot recover information stored in smart device due to unknown r_i and x_{LS}. If $\mathcal{A}$ gets smart device and password, he still cannot guess the right biometrics information by the equation $f_i = h(d_i||RPW_i \oplus d_i) \oplus h(ID_i||d_i||x_{LS}||RPW_i \oplus d_i)$ because r_i and x_{LS} are unknown.
- **Un-traceability:** As shown in Figure 5, messages in each authentication whether from the user side or the server side include random factors. On the one hand, M_1 is computed by d_i^*, r_i^* and ID_{DC} and we assume that the user will not request the same digital rights content ID_{DC}, thus ID_{DC} guarantees randomness of M_1. In addition, $\{M_2, M_3, M_{US}\}$ include the random r_1 which would not establish connection with other authentication messages. On the other hand, $\{M_4, M_5, M_{SU}\}$ contain the random factor r_2 which is different in each authentication.

6 Performance Analysis

In this section, we compare our protocol with six authenticated and authorized access schemes for DRM which are published in recent three years, i.e. [6-10] and [20]. We only consider the main body of scheme i.e. the authentication and key distribution phase, which plays a key role in the authenticated digital content distribution. Then we will perform comparisons in three aspects of security, computation cost and communication cost.

6.1 Security Comparison (Table 2)

According to analysis of section 3.2, Yu et al.'s protocol cannot withstand impersonation attack, replay attack, stolen verifier attack and cannot provide secure authentication and user anonymity. Rana and Mishra's [6], [7] do not provide three-factor security. Literature [13] pointed out that Lee et al.'s protocol [8] cannot resist denial of service attack and provide incorrect authentication phase. Lee et al.'s protocol [9] cannot resist mobile device theft attack and provides weak mutual authentication. It also can be inferred that Chen et al.'s [20] cannot withstand replay attack.

Table 2. Security comparison

Attack Resistance/Security	Ref. [10]	Ref. [6]	Ref. [7]	Ref. [8]	Ref. [9]	Ref. [20]	Ours
Impersonation attack	×	√	√	√	×	√	√
Mobile device theft	√	√	√	√	×	√	√
Offline password guessing	√	√	√	√	√	√	√
Replay attack	×	√	√	√	√	×	√
Stolen verifier	×	√	√	√	√	√	√
Privileged insider attack	√	√	√	√	√	√	√
Server spoofing attack	√	√	√	√	√	√	√
Denial of service	×	√	√	×	√	×	√
Three-factor security	×	×	×	×	×	-	√
Secure authentication	×	√	√	×	×	√	√
Anonymity	×	√	√	√	√	√	√
Un-traceability	×	√	√	√	√	√	√

6.2 Computational Cost Comparison

We adopt the running time in [13] to count computation cost of each scheme's main phase, where the experiment platform is Pentium IV computer with 512-MB RAM offering a maximum clock speed of 3 GHz. One hash operation (denoted as T_h) is 0.32ms, one fuzzy extractor (denoted as T_{fe}) is 17.1ms and one modular exponentiation (denoted as T_{me}) is 59.2ms. Due to the bit XOR operation cost is far from smaller than other operations, it is not counted in the computation cost comparison. According to Figure 5, authentication and key distribution of our protocol consumes $7T_h + 1T_{fe}$ i.e. 19.34ms in the user (U_i) side and $6T_h$ i.e. 1.92ms in the license server (LS_j) side.

Computation cost statistics of other four schemes can be seen in Table 3, that is Yu et al.'s [10] consumes 18.7ms and 0.64ms respectively in $\boldsymbol{U_i}$ side and $\boldsymbol{LS_j}$ side, Rana and Mishra's [6] consumes 19.34ms and 1.28ms respectively in $\boldsymbol{U_i}$ side and $\boldsymbol{LS_j}$ side, Mishra and Rana's [7] consumes 19.98ms and 2.56ms respectively in $\boldsymbol{U_i}$ side and $\boldsymbol{LS_j}$ side, Lee et al.'s [8] consumes 19.66ms and 2.88ms Lee et al.'s [9] consumes 19.66ms and 1.92ms respectively in $\boldsymbol{U_i}$ side and $\boldsymbol{LS_j}$ side, Chen et al.'s [20] consumes 76.94ms and 119.04ms respectively in the $\boldsymbol{U_i}$ side and $\boldsymbol{LS_j}$ side. It demonstrates that our protocol's authentication and key distribution has lower computation cost than [8-9] and [20] in both $\boldsymbol{U_i}$ side and $\boldsymbol{LS_j}$ side.

Table 3. Computation cost of authentication and key distribution phase: ms

Protocols	U_i Side	LS_j Side
Ref. [10]	$1T_{fe} + 5T_h(18.7)$	$2T_h(0.64)$
Ref. [6]	$1T_{fe} + 7T_h(19.34)$	$4T_h(1.28)$
Ref. [7]	$1T_{fe} + 9T_h(19.98)$	$8T_h(2.56)$
Ref. [8]	$1T_{fe} + 8T_h(19.66)$	$9T_h(2.88)$
Ref. [9]	$1T_{fe} + 8T_h(19.66)$	$6T_h(1.92)$
Ref. [20]	$1T_{me} + 2T_h + 1T_{fe}$ (76.94)	$2T_{me} + 2T_h$ (119.04)
Ours	$1T_{fe} + 7T_h(19.34)$	$6T_h(1.92)$

6.3 Communication Cost Comparison

Considering the 1024-bit RSA public key cryptosystem security level and calculation requirement, we take the size of hash function output (denoted as $|h|$), timestamp (denoted as $|T|$), identifier (denoted as $|ID|$), random number (denoted as $|r|$) and modular number (denoted as $|n|$) are respectively 256 bits, 32 bits, 128 bits, 128 bits and 1024 bits. As shown in figure [Fig: Login and authentication], authentication and key distribution phase of our protocol separately consumes $4|h| + 1|T|$ in the U_i side and $3|h|$ in the LS_j side.

Moreover, communication cost of other four schemes can be analyzed separately in the U_i side and the LS_j side. Yu et al.'s [10] respectively requires $2|h| + 2|ID|$ and $3|h|$. Rana and Mishra's [6] respectively requires $3|h| + 1|T|$ and $2|h| + 1|T|$. Mishra and Rana's [7] respectively requires $4|h| + 1|T|$ and $2|h|$. Lee et al.'s [8] respectively requires $3|h| + 1|T|$ and $3|h| + 1|T|$. Lee et al.'s [9] respectively requires $4|h| + 1|ID| + 1|T|$ and $3|h| + 1|r| + 1|T|$. Chen et al.'s [20] respectively requires $2|n| + 1|r| + 2|ID| + 1|T|$ and $1|n| + 1|r| + 2|ID| + 1|T|$. The comparison results are shown in Figure 7 and Figure 8, which indicates that our protocol needs narrower communication bandwidth than Lee et al.'s [9] and Chen et al.'s [20] in both user side and license server side.

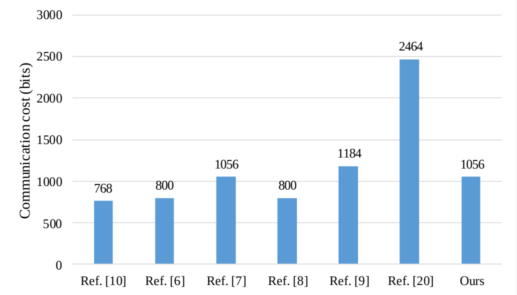

Figure 7. User side communication cost

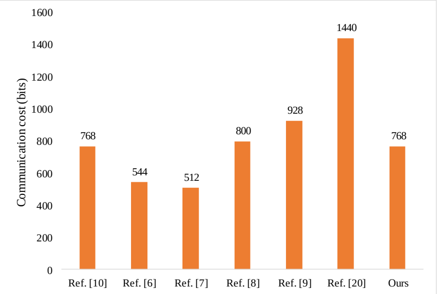

Figure 8. License server side communication cost

7 Other Related Works

With vast and various production of digital right contents, management and protection of them is increasingly becoming important recently. Rana and Mishra [21] proposed a provably secure authenticated content key distribution framework for IoT-enabled enterprise digital rights management systems. Xu [22] systematically formulated existential reasonability of digital rights management, protection measures, technology standards and usage models. Kapil [23] et al. pointed out that the increment of digital data is accompanied by data integrity, identity privacy information genuineness issues and how to achieve these security with acceptable computational and storage costs is a big challenge.

Hassan et al. [24] proposed a DRM framework utilizing robust Advanced Encryption Standard (AES) to encrypt copyrighted contents and Elliptic Curve Cryptography (ECC) cryptosystems to encrypt shared key. Specifically, this scheme uses Elliptic Curve Digital Signature Algorithm (ECDSA) to verify legality of user's license which does not take advantage of computation cost. Chen et al. [25] put forward an end-to-end collusion-secure fingerprinting framework called DeepMarks for digital management which could support authorship information retrieval and unique users identification. But they do not take into account the access process of digital right contents. Avila-Domenech et al. [26] proposed a fragile watermarking algorithm that could perceive tamper to protected image, whose disadvantage is that no clear step- by-step authentication and user access mechanism. Tabash et al. [27] reviewed and categorized several encryption works for copyright protection. But there is no specific integrity and authenticity technologies to address authentication issue.

DRM combined with blockchain technology is emerged to deal with digital rights confirmation and transaction issues. Finck and Moscon [28] researched the existed problems of DRM e.g. controversial outsourcing to private ordering in copyright low and provided idea to alleviate the controversies. Kurihara [29] discussed self-sovereign management of digital content and considered means that DRM integrated with blockchain technology. Zhang and Zhao [30] designed a DRM mechanism utilizing decentration of blockchain for reliable copyright transaction. However, these researches does not provide detailed cryptographic tools to guarantee security. In general, the blockchain-based DRM adopts smart contract to achieve automatic confirmation and transaction for distributed storage [28, 31-33]. Authentication technology [34-36] not only could be applied to wireless sensor networks and internet of drones, but also could be used for digital right contents protection. Garba et al. [37] proposed a DRM model based on scalable blockchain and discussed digital watermark encryption and authentication technologies. But these security schemes employ complex operations and cannot realize lightweight.

8 Conclusion

This paper proposed a new lightweight anonymous authentication protocol called Eland, which only adopts time-saving hash function and XOR operations. Eland provides protocol's security and user anonymity and is equipped with 13 security properties according to informal security analysis. We also introduced ROR threat model and proved Eland's mutual authentication security in the model which means the hash function security ensures Eland's security. Additionally, we compared Eland with six recent authentication protocols in aspects of security and performance. Comparisons result indicated that our protocol realizes a well trade-off between security and performance. Thereby, Eland is an efficient protocol for DRM system.

9 Acknowledgement

The work was supported by the National Natural Science Foundation of China (Nos. 62172307, 61972294, 61932016), the Blockchain Core Technology Strategic Research Program of Ministry of Education of China (No. 2020KJ010301), the Shandong Provincial Key Research and Development Program (No. 2020CXGC010107) , the Special Project on Science and Technology Program of Hubei Provience (No. 2020AEA013), the Natural Science Foundation of Hubei Province (No. 2020CFA052), the Wuhan Municipal Science and Technology Project (No. 2020010601012187) and the and Foundation of Guangxi Key Laboratory of Trusted Software (no. kx202001).

References

[1] World Intellectual Property Organization, *Standing committee on copyright and related rights*, SCCR/10/2 Rev., May, 2004, https://www.wipo.int/edocs/mdocs/copyright/en/sccr_10/sccr_10_2_rev.pdf.

[2] National Copyright Administration, http://www.ncac.gov.cn/chinacopyright/upload/files/2020/9/ 17105857106.pdf.

[3] Mordor Intelligence, *Digital Rights Management (DRM) Market - Growth, Trends, Covid-19 Impact, and Forecasts (2021-2026)*, https://www.mordorintelligence.com/industry-reports/digital-rights-management-drm-market.

[4] S. R. Subramanya, B. K. Yi, Digital rights management, *IEEE Potentials*, Vol. 25, No. 2, pp. 31-34, March-April, 2006.

[5] D J Spajić, Piracy is back: Piracy statistics for 2021, https://dataprot.net/statistics/piracy-statistics/.

[6] S. Rana, D. Mishra, Secure and Ubiquitous Authenticated Content Distribution Framework for IoT Enabled DRM System, *Multimedia Tools and Applications*, Vol. 79, No. 27-28, pp. 20319-20341, July, 2020.

[7] D. Mishra, S. Rana, Authenticated Content Distribution Framework for Digital Rights Management Systems with Smart Card Revocation, *International Journal of Communication Systems*, Vol. 33, No. 9, Article No. e4388, June, 2020.

[8] C.-C. Lee, C.-T. Li, Z.-W. Chen, Y. M. Lai, J. C. Shieh. An Improved E-DRM Scheme for Mobile Environments, *Journal of Information Security and Applications*, Vol. 39, pp. 19-30, April, 2018.

[9] C. C. Lee, C. T. Li, Z. W. Chen, Y. M. Lai, A Biometric-Based Authentication and Anonymity Scheme for Digital Rights Management System, *Information Technology and Control*, Vol. 47, No. 2, pp. 1-13, May 2018.

[10] S. J. Yu, K. S. Park, Y. H. Park, H. P. Kim, Y. H. Park, A Lightweight Three-factor Authentication Protocol for Digital Rights Management System, *Peer-to-Peer Networking and Applications*, Vol. 13, No. 5, pp. 1340-1356, September, 2020.

[11] 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, October, 2008.

[12] C.-C. Chang, J.-H. Yang, D.-W. Wang, An Efficient and Reliable E-DRM Scheme for Mobile Environments, *Expert Systems with Applications*, Vol. 37, No. 9, pp. 6176-6181, September, 2010.

[13] S. Rana, M. S. Obaidat, D. Mishra, S. Mukhopadhyay, B. Sadoun, Computational Efficient Authenticated Digital Content Distribution Frameworks for DRM Systems: Review and Outlook, *IEEE Systems Journal*, Vol. 15, No. 2, pp. 1586-1593, June, 2021.

[14] M. Abdalla, P.-A. Fouque, D. Pointcheval, Password-Based Authenticated Key Exchange in the Three-Party Setting, *Public Key Cryptography*, Les Diablerets, Switzerland, 2005, pp. 65-84.

[15] J. Srinivas, A. K. Das, N. Kumar, J. J. P. C. Rodrigues, Cloud Centric Authentication for Wearable Healthcare Monitoring System, *IEEE Transactions on Dependable and Secure Computing*, Vol. 17, No. 5, pp. 942-956, September- October, 2020.

[16] A. G. Reddy, A. K. Das, V. Odelu, A. Ahmad, J. S. Shin, A Privacy Preserving Three-Factor Authenticated Key Agreement Protocol for Client–Server Environment, *Journal of Ambient Intelligence and Humanized Computing*, Vol. 10, No. 2, pp. 661-680, February 2018.

[17] A. Gupta, M. Tripathi, A. Sharma, A Provably Secure and Efficient Anonymous Mutual Authentication and Key Agreement Protocol for Wearable Devices in Wban, *Computer Communications*, Vol. 160, pp. 311-325, July, 2020.

[18] M. Wazid, A. K. Das, V. Odelu, N. Kumar, W. Susilo, Secure Remote User Authenticated Key Establishment Protocol for Smart Home Environment, *IEEE Transactions on Dependable and Secure Computing*, Vol. 17, No. 2, pp. 391-406, March-April, 2020.

[19] D. He, D. Wang, Robust Biometrics-based Authentication Scheme for Multi-Server Environment, *IEEE Systems Journal*, Vol. 9, No. 3, pp. 816-823, September, 2015.

[20] H. B. Chen, W. B. Lee, T. H. Chen, A Novel DRM Scheme for Accommodating Expectations of Personal Use, *Multimedia Tools & Applications*, Vol. 77, No. 18, pp. 23099-23114, September, 2018.

[21] S. Rana, D. Mishra, Provably secure authenticated content key distribution framework for IoT-enabled enterprise digital rights management systems. *International Journal of Ad Hoc and Ubiquitous Computing*, Vol. 36, No. 3, pp. 131-140, March, 2021.

[22] C. Xu, Panorama of Digital Rights Management Systems, in: Regulatory Model for Digital Rights Management, *Singapore: Springer Singapore*, 2020, pp. 9-51.

[23] G. Kapil, Z. Ishrat, R. Kumar, A. Agrawal, R. A. Khan, Managing Multimedia Big Data: Security and Privacy Perspective, in: M. Tuba, S. Akashe, A. Joshi (Eds.), *ICT Systems and Sustainability*, Springer, Singapore, 2020, pp. 1-12.

[24] H. E.-R. Hassan, M. Tahoun, G. ElTaweel, A Robust Computational DRM Framework for Protecting Multimedia Contents Using AES and ECC, *Alexandria Engineering Journal*, Vol. 59, No. 3, pp. 1275-1286, June, 2020.

[25] H. Chen, B. D. Rouhani, C. Fu, J. Zhao, F. Koushanfar. Deepmarks: A Secure Fingerprinting Framework for Digital Rights Management of Deep Learning Models, *Proceedings of the 2019 on International Conference on Multimedia Retrieval*, Ottawa, ON, Canada, 2019, pp. 105-113.

[26] E. Avila-Domenech, A. Soria-Lorente, A. Taboada-Crispi, Dual Watermarking for Handwritten Document Image Authentication and Copyright Protection for JPEG Compression Attacks, *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, Havana, Cuba, 2019, pp. 656-666.

[27] F. K. Tabash, M. Izharuddin, M. I. Tabash, Encryption Techniques for H.264/AVC Videos: A Literature Review, *Journal of Information Security and Applications*, Vol. 45, pp. 20-34, April, 2019.

[28] C. Wang, J. Shen, S. Tan, PoSI: A New Consensus Protocol Based on Storage Age and Data Integrity Verification, *Journal of Internet Technology*, Vol. 22, No. 5, pp. 979-989, September, 2021.

[29] Y. Kurihara, Self-sovereign Identity and Blockchain-Based Content Management, in: D. Kreps, T. Komukai, T. V. Gopal, K. Ishii (Eds.), *Human-Centric Computing in a Data-Driven Society*, Springer International Publishing, 2020, pp. 130-140.

[30] Z. Zhang, L. Zhao, A Design of Digital Rights Management Mechanism Based on Blockchain Technology, *International Conference on Blockchain (ICBC),* Seattle, WA, USA, 2018, pp. 32-46.

[31] M. Kripa, A. N. Mahesh, R. Ramaguru, P. P. Amritha, Blockchain Framework for Social Media DRM Based on Secret Sharing, *Information and Communication Technology for Intelligent Systems*, Ahmedabad, India, 2020, pp. 451-458.

[32] D. Wang, J. Gao, H. Yu, X. Li, A Novel Digital Rights Management in p2p Networks Based on Bitcoin System, in: F. Li, T. Takagi, C. Xu, X. Zhang (Eds.), *Frontiers in Cyber Security*, Springer, 2018, pp. 227-240.

[33] Z. Xu, L. Wei, J. Wu, C. Long, A Blockchain-Based Digital Copyright Protection System with Security and Efficiency, in: K. Xu, J. Zhu, X. Song, Z. Lu (Eds.), *Blockchain Technology and Application*, Springer Singapore, 2021, pp. 34-49.

[34] D. Kumar, S. Chand, B. Kumar, Cryptanalysis and improvement of a user authentication scheme for wireless sensor networks using chaotic maps, *IET Networks*, Vol. 9, No. 6, pp. 315-325, November, 2020.

[35] S. Afsaneh, A. Sepideh, M. Ali, A. M. Salah, A two-layer attack-robust protocol for IoT healthcare security: Two-stage identification-authentication protocol for IoT, *IET Communications*, Vol. 15, No. 19, pp. 2390-2406, December, 2021.

[36] A. Huszti, N. Oláh, Security analysis of a cloud authentication protocol using applied pi calculus, *International Journal of Internet Protocol Technology*, Vol. 12, No. 1, pp. 16-25, March, 2019.

[37] A. Garba, A. D. Dwivedi, M. Kamal, G. Srivastava, M. Tariq, M. A. Hasan, Z. Chen, A Digital Rights Management System Based on a Scalable Blockchain, *Peer-to-Peer Networking and Applications*, Vol. 14, No. 5, pp. 2665-2680, September, 2021.

Biographies

Qing Fan received the B.S. degree in applied mathematics from School of Mathematics and Statistics, Shandong Normal University, Jinan, China, in 2017. She is currently taking a successive postgraduate and doctoral program at School of Mathematics and Statistics, Wuhan University, Wuhan, China. Her research interests include cryptography and information security.

Jianhua Chen received the Ph.D. degrees in applied mathematics from Wuhan University, Wuhan, China, in 1994. He is currently a Professor with School of Mathematics and Statistics, Wuhan University. His current research interests include number theory, cryptography, and information security.

Yihong Wen received his Ph.D. degree in computer application from BeiHang University in 2011. He is currently a senior engineer of CETC54. His research interests mainly include remote sensing image analysis and processing, aerospace blockchain, aerospace System simulation.

Min Luo received the Ph.D. degree in computer science from Wuhan University, Wuhan, China, in 2003. He is currently an Associate Professor with the School of Cyber Science and Engineering, Wuhan University. His research interests mainly include applied cryptography and blockchain technology.