

An Improvement to a Biometric-Based Multimedia Content Protection Scheme

Daniel Socek¹, Michal Sramka², Oge Marques³ and Dubravko Čulibrk⁴

^{1,3,4}Department of Computer Science and Engineering

²Department of Mathematical Sciences

^{1,2,4}Center for Cryptology and Information Security

Florida Atlantic University

777 Glades Road, Boca Raton, Florida 33431-0991, USA

{dsocek,msramka,omarques,dculibrk}@fau.edu

Abstract

A biometric-based method for multimedia content protection was recently proposed. The method relies on biometric data of the user and a layered encryption mechanism to achieve confidentiality of the multimedia content. Our analysis suggests that the method originally proposed has a few security related problems and several unnecessary performance bottlenecks. We point these out and propose corresponding security and performance improvements in order to make the scheme practical and efficient for the task it was originally designed for.

1 Introduction

Piracy and other copyright violations regarding digital multimedia content represent a significant problem for legal content owners and content distributors [4]. Hence, the protection of intellectual property rights for multimedia content, often referred to as the Digital Rights Management (DRM) for multimedia, recently started receiving a considerable amount of interest.

Most of the recently proposed DRM solutions make use of encryption and digital watermarking techniques [3]. In general, encryption techniques are used to prevent illegal copying and distribution, while digital watermarking techniques can be used to either prevent it or discourage it by allowing grounds for legal actions. However, most of the recently proposed DRM methods combine both encryption and digital watermarking to ensure better security against illegal copying and distribution [8]. In fact, digital watermarking techniques are often used to complement the encryption techniques within a DRM architecture.

Preventing illegal content copying and distribution is indeed a hard problem to solve. Over the course of many years, cryptographic techniques providing the standard goals of security (such as confidentiality, authentication, data

integrity and non-repudiation) have evolved. Unfortunately, these techniques do not provide straightforward solutions to the problem of content protection against illegal copying or illegal distribution. In classical cryptography, two communicating parties, say Alice and Bob, try to confidentially communicate the content through a non-trusted channel, while an enemy, say Eve, tries to obtain the content by listening to the channel and applying cryptanalysis to the acquired data. On the other hand, in the case of content protection, Alice tries to send the content securely to Bob's trusted device (software or hardware), while both Eve and Bob are considered enemies. In particular, Eve must not be able to utilize the content nor to copy or distribute it, while Bob is allowed to utilize the content, via the trusted device, but not allowed to copy or distribute it. Thus, for DRM it is crucial that the system is trusted and can not be tampered with. The basic assumption behind the current DRM technology is the existence of such trusted, tamper resistant environments.

One of the main goals for any DRM scheme is to allow the legal user to utilize the content, but at the same time, to prevent the user from content sharing. Thus, the content must not be shipped/transmitted in its clear form, but in a DRM packaged (wrapped) form [8]. The user, say Alice, must not be able to unwrap the packaged content directly, but through the use of the trusted, tamper resistant device owned by Alice. However, the same device owned by Bob must not be able to unwrap the content that Alice purchased. In addition, Alice's trusted device that is run by Bob must not be able to unwrap the content intended for Alice. Thus, an authentication method must be established between the user and user's trusted device.

In [10], different authentication methods were studied and their properties compared. The authentication methods that are based on a user's knowledge are limited to a relatively weak human memory. As such, they could be guessed or brute-forced. The methods based on a physical identification item in a user's possession can store much more information. However, the identification items could be lost, stolen, or duplicated. Such a method is often used in combination with a user's knowledge. This is done to prevent an unauthorized user from having the ability to immediately use the stolen item, thus allowing for a short period of time for the real user to obtain a new and different identification item. The methods based on a user's biometrics have several advantages over the aforementioned authentication methods. The greatest benefits of the biometric methods are the simplicity of use and the very limited risk of losing, stealing, or forging the user's biological identifier. The main weakness of the biometrics based methods is the inability to renew a compromised biological identifier [11]. This is a particularly significant issue since biometric-based authentication with the same biometrics is likely to be used in multiple applications and systems. For example, a fingerprint-based authentication could be used to gain access to a bank account, but also to gain access to a computer system or a highly secure laboratory. Therefore, storing or transmitting the user's biological identifier in its pure form should be avoided. Instead, a hashed version of the biometrics data should be used. Unfortunately, the imperfect biometric data acquisition process prevents the direct use of cryptographically secure hash functions (such as MD5

or SHA-1). Namely, since the biometric data of the same user varies at different acquisition times, it is not possible to utilize cryptographic hash functions which all have a strong avalanche property. Rather, a *robust hash function* must be used. A robust hash function here refers to a hash function that gives the same (or close) hashes of two close inputs and significantly different hashes for different inputs. A good overview of robust hash functions is given in [11].

Another important consideration related to biometrics-based security systems is the sensor’s ability to reject forgeries. A survey of different sensor based forgeries is presented in [13]. For example, it is reported by Putte and Keuning that a fake rubber finger gained access to several fingerprint based security systems since the sensors were not measuring perspiration and other more advanced features [9]. In this paper, we do not consider attacks based on deceiving the biometric sensor.

The rest of this paper is organized as follows. Section 2 reviews relevant details of a recently proposed biometrics based multimedia content protection scheme (Uludag-Jain scheme). In Section 3, we present a rigorous security and performance analysis of the originally proposed Uludag-Jain scheme, while in Section 4 we suggest several improvements. Finally, Section 5 holds our conclusions.

2 Uludag-Jain Scheme

In 2003, Uludag and Jain proposed a multimedia content protection system based on biometric data of the users and a layered encryption scheme. In this section we describe relevant details of the biometrics-based multimedia content protection scheme by Uludag and Jain from [12].

The Uludag-Jain scheme assumes two communicating parties, the server S and the user U . The multimedia content that is to be delivered to U for utilization is a file V that resides on S . The scheme uses both a symmetric-key cryptosystem and a public-key cryptosystem. While the authors did not specify or recommend which public-key cryptosystem to use, for the symmetric-key cryptosystem they recommended to use either *Data Encryption Standard* (DES) or *Advanced Encryption Standard* (AES).

Let K_S^+ and K_S^- denote the public and private keys of S , respectively, and let $K_S^+(\cdot)$ and $K_S^-(\cdot)$ denote their application in a given public-key cryptosystem. Furthermore, let $E(X, k_1, k_2, \dots, k_n)$ denotes the superencryption of file X with a given symmetric-key cryptosystem with keys k_1, k_2, \dots, k_n , respectively. Similarly, let $D(Y, k_n, k_{n-1}, \dots, k_1)$ denotes the superdecryption of file Y with keys k_n, k_{n-1}, \dots, k_1 , in that order. It is assumed that U has previously registered with S and selected the unique identifier I_U (e.g. username) and the password P_U , and that the registration was performed using a secure connection, so that I_U and P_U are known only to U and S . Let B_U^t , $t = 0, 1, 2, \dots$ denote the biometric data of U obtained at time t . As noted earlier, the biometric data of the same user obtained at different times may not be identical. That is, B_U^i is not necessarily equal to B_U^j for different times i and j .

At first, U encrypts I_U , P_U and B_U^0 using a public-key cryptosystem with key K_S^+ and sends the encrypted data to S . Upon receiving the encrypted data, S decrypts I_U , P_U and B_U^0 using the server's private key K_S^- . At this point, the server authenticates the user U by checking the given I_U and P_U against the user database. Here, it is assumed that the identity of a user is not stolen. Once the user is authenticated and content V approved for delivery, S creates a password P_{SUVT} as a function of server S , user U , content V , and a time stamp T . Next, S encrypts P_{SUVT} with its private key K_S^- and sends it to U , who easily recovers P_{SUVT} by using the server's public key K_S^+ . The server then calculates $V_1 = E(V, P_{SUVT}, P_U, I_U, B_U^0)$, appends B_U^0 to V_1 into V_2 , obtains $V_f = E(V_2, P_{SUVT}, P_U, I_U)$, and finally sends V_f to U . In order to utilize the content V , the user recovers V_2 and B_U^0 from V_f by superdecrypting it with keys I_U , P_U , and P_{SUVT} , respectively. Then, U compares B_U^0 with B_U^1 which is obtained from a user's biometric sensor at the time of utilization. If biometric data B_U^0 and B_U^1 match, the content V is recovered by superdecrypting V_1 with keys B_U^0 , I_U , P_U , and P_{SUVT} , in that order, after which user U can freely utilize content V .

3 Security and Performance Analysis

First, a performance analysis of the Uludag-Jain scheme is provided. The security analysis then follows.

The superencryption and superdecryption using symmetric key cryptosystem, namely $E(X, k_1, k_2, \dots, k_n)$ and $D(X, k_n, k_{n-1}, \dots, k_1)$, is used twice. Once with 4 different keys and then with 3 different keys. In practice, this means that an ordinary encryption (or decryption) is performed 7 times. Each time the whole multimedia content V (or its derivatives of comparable length) is processed. The multimedia content V can be of considerable size, starting from 5 MBytes for audio files to hundreds of MBytes for video files. Hence these 7 encryptions (and later 7 decryptions) create a computational bottleneck that does not increase security of the whole scheme. In addition, the biometric data B_U^0 is appended to the derived content V_1 , and then the whole result is encrypted again. Because of the sizes, this creates an unnecessary computational bottleneck, and can be avoided.

From a security point of view, the scheme has two major problems. The first problem is a consequence of the security model under which the scheme operates. As the original authors also argue [12], tamper resistance must be a requirement for this scheme so that the user is not able to violate the given utilization rights. Thus, the authors require a "closed application", where the decrypted file is not stored at the user's computer but decrypted just before it is played. The biometric sensor, matcher, decryption module, media player and playing medium (e.g., monitor, speaker, etc.) are assumed to be connected together securely, where no tampering is possible. However, the channel between

the server S and the user U is clearly not assumed to be tamper resistant, nor such assumption should be made due to practical reasons. Thus, both the user U and the attacker A can eavesdrop the encrypted content V_f . If the user gains access to V_f , he or she has all the information to recover the raw content V and then violate the security policy: I_U and P_U are initially known; P_{SUVT} is recovered from $K_S^-(P_{SUVT})$; V_1 and B_U^0 are recovered from V_2 that is recovered from V_f ; V is recovered from V_1 . If the attacker A steals I_U and P_U , he or she breaks the scheme in the same way. In doing so, the attacker also gets U 's biometric template B_U^0 , which may have even more serious consequences if the attacker can learn a single piece of information about U 's biometric identifier (e.g. location of the fingerprint minutiae). Such form of identity theft where the identifier is not renewable is the most dangerous one. A related consideration is that the server S must store the biometric templates of all system users, which in addition to storage issue, brings also a responsibility to protect non-renewable identities of their customers. On the other hand, a tamper resistant system, such as the one required by the original scheme, can safely store the values I_U , P_U , P_{SUVT} and B_U^0 in a secure domain.

The second security problem is the incorrect use of an asymmetric-key cryptosystem. At some point in the protocol, the server S creates a password P_{SUVT} and wants to securely deliver it to the user U . This is accomplished in the scheme by sending the value of $K_S^-(P_{SUVT})$ from S to U . However, since the public key K_S^+ is publicly known, anybody, including an attacker, can obtain the password by computing $K_S^+(K_S^-(P_{SUVT}))$, if the asymmetric scheme allows it. For example, these operations define the signing and verification equivalents in the case of the RSA scheme, but have no meaning in the ElGamal scheme.

Finally, we suggest that the obsolete DES cryptosystem should not be used with the proposed scheme. The scheme should be restricted to using AES cryptosystem, a change that has positive implications for both security and performance of the scheme.

4 An Improved Content Protection Scheme

We now propose a new scheme for content protection. The scheme consists of three phases: (1) *content request*, (2) *content delivery*, and (3) *user-device authentication*.

In our scheme we have three communicating parties, the user U , U 's tamper-proof device D_U and the server S . During the content request phase, the user purchases a right to utilize content V with device D_U , for which the user obtains an authorization number A that represents his or her proof of purchase. The user passes the number A to his or her device D_U . The same authorization number A is stored at the server S . As mentioned previously, it is a crucial assumption that user's device system is trusted and tamper resistant.

At the start of the content delivery phase, D_U has: (1) the purchase infor-

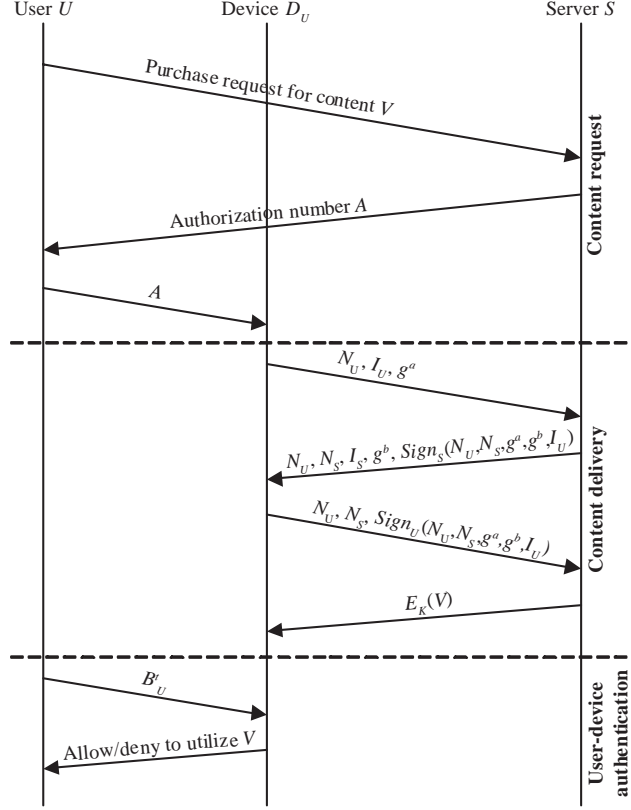


Figure 1: Sequence diagram of the proposed biometric-based multimedia content protection protocol.

mation confirmation number A , (2) U 's private key for signing messages, and (3) a public generator g of order n of a group. Let $Sign_x(y)$ denote user x 's digital signature of message y , and H a cryptographically secure hash function with a hash value size equal to the utilized AES keysize (128, 192, or 256 bits). The server S has its own private key for signing messages and the purchase confirmation number A that is somehow associated with the content V .

In the first step, D_U generates a random number (nonce) N_U , U 's private key a ($0 \leq a < n$), and sends N_U , U 's identity I_U , and g^a to S . The server S generates its own random number (nonce) N_S and its private key b ($0 \leq b < n$). S sends N_U , N_S , its identity I_S , and g^b together with $Sign_S(N_U, N_S, g^a, g^b, I_U)$ to D_U . D_U verifies the signature and replies by sending N_U , N_S , and signature $Sign_U(N_U, N_S, g^a, g^b, I_S)$ to S . The server S then verifies the signature, generates key $K = H(g^{ab})$, encrypts V as $V' = E_K(V)$ using AES block cipher, and sends it to D_U . The user's device D_U can also compute the key $K = H(g^{ab})$ and hence decrypt V' into the content V .

The final phase of the process involves user-device authentication. In particular, user U must be authenticated with his or her device D_U before the content can be utilized. As discussed earlier, the use of biometric-based authentication has several advantages over other authentication methods. The biometric sensor, which is itself connected and placed in the tamper-resistant trusted environment, is used to gather a biometric identifier B_U^t of user U at time t of the content utilization request. If the identifiers B_U^t and B_U^0 match, the device D_U allows the content utilization, otherwise it rejects the utilization request. Here, it is assumed that the biometric identifier B_U^0 of user U is initially stored at D_U . Figure 1 shows the sequence diagram of the proposed protocol.

4.1 Some implementation issues

The purchase confirmation number A either contains information about the content V directly, or the server S has to keep a track of what confirmation number refers to what content. The number A can contain some other information, too (e.g. expiration time). In any case, for security reasons, the random part of A should be at least 128 bits. After the confirmation number A is obtained for the first time, the server S should associate the number A with the identity of U or its hash value, and deny future attempts for the same information from different users. In addition, it is recommended that the nonces are each at least 32 bits long.

4.2 Security of the scheme

The first part of the proposed scheme exactly corresponds to ISO 9798-3 protocol [7] which is known to be provably secure (SK-secure) [2]. As such, it inherently possesses the resistance to man-in-the-middle attacks, resistance to known-key attacks, forward secrecy, and more. This is assuming that the generator g generates a group with a hard decision Diffie-Hellman assumption, and that the underlying signature scheme is secure. The use of nonces (N_U and N_S) prevents replay attacks, serves as a session key identifier, and also protects against parallel runs (so that the participants know which run a message belongs to). Furthermore, the security of the scheme is heavily based on the assumption that the value of A is being kept confidential in the trusted, tamper resistant system. The same applies for the key K . In addition, by adding only one more exchange step, it is easy to extend this protocol to withstand *Denial of Service* (DoS) attacks [1]. This introduces a negligible performance overhead for applications where DoS is of concern.

In difference to the original Uludag-Jain scheme, we do not associate the value B_U^0 with the key K or the protected (encrypted) content V' . We use the value of B_U^0 as a form of authentication to the device D_U . By making the assumption that the device is tamper resistant, we can securely store the value of B_U^0 , compare it with future values of B_U^t for $t \geq 1$, and if they match, provide the content utilization.

In some cases, it may be undesirable to store the raw biometric information (feature vector) of a user in the user’s device. Unless the tamper-resistant technology of the device provides level 4 security (e.g. IBM 4758 Model 002), or at least at level 3 security (e.g. IBM 4758 Model 023), as defined in FIPS PUB 140-1 specification, storing the raw biometric information in the device should be avoided due to non-renewability of biometric information and its potential use in many different personal transactions, applications and security protocols. Unfortunately, devices with high levels of tamper-resistance are expensive to produce, and as such, are unlikely to be used in DRM applications involving a relatively low-cost entertainment multimedia content and a fairly large number of potential users. When lower cost tamper-resistant devices (such as smart-cards) are used, usually providing only levels 1 and 2 security, the biometric identifier B_U^0 of user U should be transformed using a secure robust hash function before storing it.

4.3 Performance

Both user’s device D_U and server S need to generate some random numbers, compute exponentiation g^a and g^b , respectively, and sign one message each. The possible speedup can be achieved by letting the server use its private key b for a bounded period of time. In this case, the costly exponentiation g^b is computed only once in a while, not for every new request. The computation of the key K would have to be modified to depend on N_U and N_S for additional security as follows: $K = H(g^{ab} || N_U || N_S)$.

The symmetric encryption is applied to the original content V only once. In comparison to the original scheme which uses superencryption and applies the symmetric encryption 7 times, this is a reduction by a factor of 7. In addition, by replacing the obsolete DES cipher with the more secure AES cipher, we gain additional speedup. FPGA and ASIC implementations of AES are able to encrypt/decrypt with a speed of 20-70 Gbits/s [6, 5]. This is especially suitable for the server S that may have to process many requests at a time.

5 Conclusions and Future Work

Uludag-Jain presented an interesting idea of content protection based on biometric data. However, their proposed multimedia content protection scheme has several security flaws, and possesses several unnecessary performance bottlenecks. We analyzed the original Uludag-Jain scheme from a security and performance point of view, and showed that some of the proposed steps in the protocol are not secure and that the scheme includes too many encryptions. Based on the analysis, we proposed a new multimedia content protection scheme. We reused the idea of using the user’s biometric data, however, we simplified the protocol to use this data only for the user-device authentication. We included some implementation issues, showed performance analysis, and provided preliminary security analysis of our proposed scheme. Our contribution

results in an improved scheme that accomplishes the same security goals as the original proposal, but in a more secure, concise and efficient way.

Future work could include extending the scheme to support parental rating of the content, variable number of allowed utilizations, time limitations for the content utilization, and similar controlling mechanisms.

References

- [1] W. Aiello, S. M. Bellovin, M. Blaze, R. Canetti, J. Ioannidis, A. D. Keromytis, and O. Reingold. Just fast keying: Key agreement in a hostile internet. *ACM Transactions on Information and System Security*, 7(2):1–30, 2004.
- [2] R. Canetti and H. Krawczyk. Analysis of key-exchange protocols and their use for building secure channels. *In Proc. EUROCRYPT 2001, Lecture Notes in Computer Science*, 2045:453–474, 2001.
- [3] J. Dittmann, P. Wohlmacher, and K. Nahrstedt. Using cryptographic and watermarking algorithms. *IEEE Multimedia*, 8(4):54–65, 2001.
- [4] B. Furht, E. A. Muharemagic, and D. Socek. *Multimedia Security: Encryption and Watermarking*, volume 28 of *Multimedia Systems and Applications*. Springer, 2005.
- [5] A. Hodjat and I. Verbauwhede. Minimum area cost for a 30 to 70 gbits/s aes processor. *In Proc. 2004 IEEE Computer Society Annual Symposium on VLSI (ISVLSI 2004), Emerging Trends in VLSI Systems Design*, pages 83–88, 2004.
- [6] A. Hodjat and I. Verbauwhede. A 21.54 gbits/s fully pipelined aes processor on fpga. *In Proc. 12th IEEE Symposium on Field-Programmable Custom Computing Machines (FCCM 2004)*, pages 308–309, 2005.
- [7] IEEE. Entity authentication mechanisms - part 3: Entity authentication using assymmetric techniques. *Tech. Rep. ISO/EIC IS 9798-3, ISO*, 1993.
- [8] E. T. Lin, A. M. Eskicioglu, R. L. Lagendijk, and E. J. Delp. Advances in digital video content protection. *IEEE: Special Issue on Advances in Video Coding and Delivery*, 93(1):171–183, 2005.

- [9] T. Putte and J. Keuning. Biometrical fingerprint recognition: dont get your fingers burned. *In Proc. IFIP TC8/WG8.8, Fourth Working Conf. Smart Card Research and Adv. App.*, pages 289–303, 2000.
- [10] N. Ratha, J. Connell, and R. Bolle. Enhancing security and privacy in biometrics-based authentication systems. *IBM Syst. J.*, 40(3):614–634, 2001.
- [11] Y. Sutcu, H. T. Sencar, and N. Memon. A secure biometric authentication scheme based on robust hashing. *In Proc. ACM Multimedia and Security Workshop (MM&Sec'05), New York, NY*, pages 111–116, 2005.
- [12] U. Uludag and A. Jain. Multimedia content protection via biometrics-based encryption. *In Proc. IEEE International Conference on Multimedia and Expo (ICME2003), Baltimore, MD*, 3:237–240, 2003.
- [13] U. Uludag, S. Pankanti, S. Prabhakar, and A. Jain. Biometric cryptosystems: Issues and challenges. *IEEE Special Issue on Enabling Security Technologies for Digital Rights Management*, 92(6):948–960, 2004.