BIOMETRICS AND FUZZY IDENTITY-BASED ENCRYPTION

EMMANUEL TSUKERMAN

Abstract. Biometric data is attractive in the field of cryptography because it uniquely identifies and authenticates an individual. The biometric data itself can also be used as a key for encryption which can later be uniquely decrypted by the same person. We discuss some of the uses of biometric data in cryptography and, in particular, its use in the Fuzzy Identity-Based Encryption scheme, a scheme which improves upon standard IBE.

1. Introduction

The importance of strong cryptographic systems is hard to overemphasize. Content owners, such as authors and authorized distributors, lose billions of dollars each year due to illegal copying and sharing of digital media [6], [7]. The problem is usually addressed via Digital Rights Management (DRM) systems, which rely on user authentication to determine whether a user has access to the content. In a generic cryptographic system, however, the authentication is possession based, meaning that possession of the secret key is sufficient to establish identity. Because cryptographic keys are long and random (e.g., the advanced encryption standard (AES) uses 128 bit keys [8]), they are difficult to memorize. As a result, they are typically stored either on the computer or somewhere else, and then released based on an alternate authentication mechanism such as password input. Most passwords can be easily broken via social engineering methods or dictionary attacks. Complex passwords are usually placed in easily accessible locations, e.g., a post-it on a nearby board, and as a consequence passwords are often unsafe. Moreover, users often reuse the same password for different services, so that if one password is compromised, it may open many doors. The need for a more effective system is present, and a good substitute would be very valuable. Fortunately, many of the limitations of traditional passwords can be ameliorated by incorporating a better method of user authentication.

2. Biometrics

Biometric authentications is the authentication of users based on their biometrics. A biometric is data describing a person's physiological or behavioral characteristics. Examples of biometrics include fingerprints, hand geometry, iris, retina, signature, key stroke and voice. Below, we list a comparison chart of various biometrics as seen in Uludag et al [2].

Biometrics have several important advantages over traditional passwords:

(1) Biometrics uniquely identify individuals.
(2) They are more complex and random than ordinary human-generated passwords.
(3) They are always in the user's possession and cannot be lost.

Date: 12.15.2011.

Biometric identifier	Universality	Distinctiveness	Permanence	Collectability	Performance	Acceptability	Circumvention
Face	H	L	M	H	L	H	H
Fingerprint	M	H	H	M	H	M	M
Hand geometry	M	M	M	H	M	M	M
Iris	H	H	H	M	H	L	L
Keystroke	L	L	L	M	L	M	M
Signature	L	L	L	H	L	H	H
Voice	M	L	L	M	L	H	H

FIGURE 2.1. Comparison of various biometric technologies based on the perception of the authors in [2]. Universality: Do all people have it? Distinctiveness: Can people be distinguished based on this identifier? Permanence: Does the identifier change its features with time? Performance: How accurate is the technique and how fast can it be measured? Acceptability: Willingness of people to use it.

(4) They are more difficult to copy, share and distribute.
(5) They require the user to be present at the time of biometric verification

Thus biometrics-based authentication is a very possible replacement for password-based authentication.

One potential difficulty in utilizing biometrics for cryptography is that biometric measurements are unavoidably noisy. For instance, a dry skin finger print and a skin with a normal amount of moisture will produce different readings. As such, we cannot utilize biometric encryption via standard cryptography. Several approaches have been implemented to deal with this problem and are outlined in [3]. For the convenience of the reader, we will outline these below.

2.1. **Fuzzy commitment scheme.** This approach, which is given in detail in [4], addresses the fuzziness of biometric data through error-correcting codes. The scheme operates as follows:

- The user inputs their biometric B.
- The enrollment process takes B and encrypts it into a codeword C which is near to B (e.g., a lattice point in the same block as B).
- The system generates a hash H of C and calculates a Δ between B and C.

The function H of C is called the fuzzy commitment of C and does not reveal much about B. Authentication proceeds as follows:

- A user presents a biometric B'.
- The system computes $C' = B' + \Delta$ and then a hash H' of C'.

The user is authorized if $H' = H$.

Another, simpler alternative is to use the Euclidean or hamming distance of B and B' and have the system declare a match if the distance is below a given threshold.

2.2. **Fuzzy vault scheme.** This approach is a variant of the fuzzy commitment scheme presented by Juels and Sudan [5]. In this scheme, Alice can lock her secret using a set A and Bob will be able to unlock the secret with his own set B provided that B overlaps enough with A. For instance, A might be Alice's biometric and B might be Alice's biometric rescanned. To lock a secret using a set A, the scheme proceeds as follows:

- Alice selects a polynomial p with embedding of c in its coefficients.
- Alice evaluates p on the elements of A.
- Alice creates a random set of points that do not lie on p.

The randomly generated points allow Alice to conceal p. To authenticate, Bob uses the set B to

- Identify many points and hence recover a large number of correct points.
- Uses the Reed Solomon error-correcting code to remove noise.

If successful in decrypting, this outputs a polynomial intersecting a large number of input points.

3. Introduction to Fuzzy IBE

One cryptographic application that stands to gain greatly from the use of biometrics is identity-based encryption (IBE). In IBE, a central authority (PKG) creates a private master key sk_{PKG} and a public key pk_{PKG}, which is available to all interested parties **(setup stage)**. A user Bob authenticates himself to the PKG to obtain a private key sk_{IDBob} associated with his identity ID_{Bob} **(private key extraction stage)**. Using Bob's identity ID_{Bob} and the public key pk_{PKG}, another user Alice encrypts her message M to obtain a cipher-text C **(encryption stage)**. Upon receiving C from Alice, Bob decrypts C using his private key sk_{IDBob} and recovers the message M **(decryption stage)**. In practice, the identity could be an e-mail address or an ip address.

A particular step that would benefit greatly from the use of biometric encryption is the private key extraction step. In this step, Bob authenticates himself to the central authority. Typically, this will involve Bob presenting the central authority with supplementary documents and credentials. However, these documents themselves could be forged. As a consequence, there will be a trade off between having a system that is expensive in this step and a less secure system. The use of a biometric as an identity, however, could significantly better this step, and save resources. The user will have to demonstrate ownership of the biometric under supervision of a well trained operator. If the operator is able to detect imitation attacks (e.g., playback of a recording of a voice) then the security of this phase will only be limited by the quality of the biometric technique itself. There is also the advantage that an identity that is a biometric is unique to a person, as opposed to an identity such as, e.g., "Bob Smith", which might even change users over time.

As mentioned earlier, biometric measurements are noisy so that standard IBE does not work well with biometrics. To address this problem, Sahai and Waters [1] developed the Fuzzy-IBE, the error-tolerance of which allows for a private key derived from a measurement of a biometric to decrypt a ciphertext encrypted with a slightly different measurement of the same biometric. The framework has the advantage of being collusion resistant, avoiding the use of random oracles and having security reducible to an assumption similar to the Decision Bilinear Diffie-Hellman (DBDH) assumption. Before we introduce the scheme, a few preliminaries are in order.

4. Bilinear Maps

Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be groups of prime order p, and let g be a generator of $\mathbb{G}_1$. We say that $\mathbb{G}_1$ has an admissible bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ into $\mathbb{G}_2$ if the following two conditions hold:

- The map is bilinear, meaning that $e(g^a, g^b) = e(g, g)^{ab}\ \forall a, b \in \mathbb{G}_1$.
- The map is non-degenerate, meaning that $e(g, g) \neq 1$.

Recall also the DBDH assumption:

Definition 1. (Decision Bilinear Diffie-Hellman (DBDH) Assumption). Suppose a challenger chooses $a, b, c, z \in \mathbb{Z}_p$ at random. The DBDH assumption is that no polynomial-time adversary is able to distinguish the tuple ($A = g^a$, $B = g^b$, $C = g^c$, $Z = e(g, g)^{abc}$) from ($A = g^a$, $B = g^b$, $C = g^c$, $Z = e(g, g)^z$) with more than a negligible advantage.

We will later see that the security of the Fuzzy-IBE construction can be reduced to the following security assumption:

Definition 2. (Decision Modified Bilinear Diffie-Hellman (DMBDH) Assumption). Suppose a challenger chooses $a, b, c, z \in \mathbb{Z}_p$ at random. The DMBDH assumption is that no polynomial-time adversary is able to distinguish the tuple ($A = g^a$, $B = g^b$, $C = g^c$, $Z = e(g, g)^{\frac{ab}{c}}$) from ($A = g^a$, $B = g^b$, $C = g^c$, $Z = e(g, g)^z$) with more than a negligible advantage.

The modified assumption (DMBDH) implies the non-modified decision bilinear Diffie-Hellman assumption (DBDH), so that the security assumption of DMBDH is a stronger assumption. Indeed, suppose that we are given a DMBDH challenge (A, B, C, Z) and an algorithm which has a non-negligible DBDH advantage for the bilinear map involved. We can invert C and run the algorithm on the tuple (A, B, C^{-1}, Z). It is not hard to see that if we make the same decision as the algorithm on this tuple, we would have a non-negligible DMBDH advantage.

5. Fuzzy-IBE Construction

In the Fuzzy-IBE scheme, we view identities as sets of attributes[1]. We set a threshold value d, which represents the error-tolerance of the system in terms of set overlap. More Specifically, a secret key ω' can decrypt a ciphertext created by identity ω only if $|\omega \cap \omega'| \geq d$. The central authority will give each user a random polynomial $q(x)$ of degree $d - 1$ which evaluates to y at 0: $q(0) = y$.

For each of the attributes associated to a user's identity, the key generation algorithm will issue a private key component D_i which will be tied to the user's polynomial. If the user is able to "match" at least d components of the ciphertext with their private key components, then they will be able to decrypt. In practice, this will be done via polynomial interpolation. However, since the private key components D_i are tied to specific random polynomials, two or more users will not be able to combine their private key components to attack a single polynomial. This means that the system is resistant to collusion attacks.

We let $\mathbb{G}_1$ be a group of prime order p and let g be a generator of $\mathbb{G}_1$. Also, let $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ be a bilinear map.

[1]The following discussion borrows greatly from Sahai and Waters' paper [1].

We define the Lagrange coefficient $\Delta_{i,S}$ for $i \in \mathbb{Z}_p$ and a set S of elements in $\mathbb{Z}_p$:

$$\Delta_{i,S}(x) = \prod_{j \in S, j \neq i} \frac{x-j}{i-j}.$$

Identities will be subsets of a universe $\mathcal{U}$ of size $|\mathcal{U}|$. Each elements will be associated with a unique integer in $\mathbb{Z}_p^*$. The construction is as follows:

Setup(d) Define the universe $\mathcal{U}$ of elements. For simplicity, we take $\mathcal{U} = \{1, 2, ..., |\mathcal{U}|\}$ with integers in $\mathbb{Z}_p^*$. Next we choose $t_1, t_2, ..., t_{|\mathcal{U}|}$ and y uniformly at random from $\mathbb{Z}_p$. We publish the public key

$$pk = (T_1, T_2, ..., T_{|\mathcal{U}|}, Y),$$

where $T_i = g^{t_i}\ \forall i = 1, 2, ..., |\mathcal{U}|$ and $Y = e(g,g)^y$. The master key is

$$mk = (t_1, t_2, ..., t_{|\mathcal{U}|}, y).$$

Key Generation Given an identity $\omega \subset \mathcal{U}$, a $d-1$ degree polynomial q is randomly chosen such that $q(0) = y$. The private key consists of the components $(D_i)_{i \in \omega}$, where $D_i = g^{\frac{q(i)}{t_i}}$ for every $i \in \omega$.

Encryption Given a public key ω' and message $M \in \mathbb{G}_2$, we first choose a random value $s \in \mathbb{Z}_p$. The ciphertext is then published as

$$E = (\omega', E' = MY^s, \{E_i = T_i^s\}_{i \in \omega'}).$$

Decryption Suppose that E is a ciphertext as above and we hold a private key for ω, where $|\omega \cap \omega'| \geq d$. Choose an arbitrary set S in $\omega \cap \omega'$ with d elements. The ciphertext is decrypted as follows:

$$\begin{aligned} & E' / \prod_{i \in S} (e(D_i, E_i))^{\Delta_{i,S}(0)} \\ &= Me(g,g)^{sy} / \prod_{i \in S} (e(g^{\frac{q(i)}{t_i}}, g^{st_i}))^{\Delta_{i,S}(0)} \\ &= Me(g,g)^{sy} / \prod_{i \in S} (e(g,g)^{sq(i)})^{\Delta_{i,S}(0)} \\ &= M. \end{aligned}$$

The last equality is derived from using polynomial interpolation in the exponents. Indeed, consider the exponent of $e(g,g)$ in the expression:

$$sy - sq(i_1)\Delta_{i_1,S}(0) - sq(i_2)\Delta_{i_2,S}(0) - ... - sq(i_d)\Delta_{i_d,S}(0),$$

where $i_k \in S\ \forall k = 1, 2, ..., d$. Factoring out s, it suffices to see that $y - q(i_1)\Delta_{i_1,S}(0) - q(i_2)\Delta_{i_2,S}(0) - ... - q(i_d)\Delta_{i_d,S}(0) = 0$. This is true by the following reasoning. The expression $q(i_1)\Delta_{i_1,S}(x) + ... + q(i_d)\Delta_{i_d,S}(x)$ is the Lagrange interpolating polynomial through

$$(i_1, q(i_1)), (i_2, q(i_2)), ..., (i_d, q(i_d)).$$

By construction, $q(x)$ is a $d-1$ degree polynomial, so that d of its points determine it uniquely. Therefore $q(i_1)\Delta_{i_1,S}(0) + q(i_2)\Delta_{i_2,S}(0) + ... + q(i_d)\Delta_{i_d,S}(0)$ is the evaluation of q at $x = 0$, which by construction is equal to y.

6. Security of Fuzzy-IBE

The Fuzzy Selective-ID game is a security game in which the adversary is allowed to query for secret keys for identities which have less than d overlap with the target identity. Such a restriction makes sense from a practical point of view because if two identities share more than d attributes, then either the two identities are the same or d has not been set high enough so that two individuals cannot be properly distinguished via the biometric scan.

6.1. **Fuzzy Selective-ID.**

Init Adversary declares an identity α to be challenged upon.

Setup The challenger runs the setup phase of the algorithm and publishes the public parameters.

Phase 1 The adversary issues queries for private keys of many identities γ_j with the restriction that $|\gamma_j \cap \alpha| < d$ for all j (see discussion at the beginning of this section for an explanation).

Challenge The adversary submits two equal length messages M_0, M_1. The challenger picks a random $b \in \{0, 1\}$ and encrypts M_b with α. The ciphertext is forwarded to the adversary.

Phase 2 Phase 1 is repeated.

Guess The adversary outputs a guess b' of b.

We define the advantage of this game to be $\Pr[b' = b] - \frac{1}{2}$ and say that

Definition 3. (Fuzzy Selective-ID) A scheme is secure in the Fuzzy Selective-ID model of security if all polynomial-time adversaries have at most a negligible advantage in the security game.

We then have the following statement, the proof of which is adopted from [1]:

Theorem 4. *If an adversary can break Fuzzy IBE in the Fuzzy Selective-ID, then an algorithm can be constructed which has a non-negligible advantage in the DMBDH game.*

Proof. Suppose that $\mathcal{A}$ is an adversary having advantage ϵ in the Fuzzy Selective-ID game played against the Fuzzy IBE scheme. Let C be the challenger and $\mathcal{B}$ be the algorithm we are constructing to play the DMBDH game.

The challenger proceeds to set up the groups $\mathbb{G}_1, \mathbb{G}_2$, the universe $\mathcal{U}$, the map e and a generator g. He then flips a fair binary coin μ. If $\mu = 0$, the challenger sets $(A, B, C, Z) = (g^a, g^b, g^c, g^{\frac{ab}{c}})$; otherwise he sets $(A, B, C, Z) = (g^a, g^b, g^c, g^z)$ for random a, b, c, z.

Init The algorithm $\mathcal{B}$ runs $\mathcal{A}$ and receives the challenge identity α.

Setup Algorithm $\mathcal{B}$ then sets up the public parameters as follows. First, $Y = e(g, A) = e(g, g)^a$. For each $i \in \alpha$ it chooses random $\beta_i \in \mathbb{Z}_p$ and sets $T_i = C^{\beta_i} = g^{c\beta_i}$. For each $i \in \mathcal{U} - \alpha$, it chooses

a random $w_i \in \mathbb{Z}_p$, sets $T_i = g^{w_i}$ and then forwards these public parameters to $\mathcal{A}$. Notice that from the view of $\mathcal{A}$ all parameters are chosen random as in the construction.

Phase 1 Adversary $\mathcal{A}$ makes requests for private keys where the identity set overlap between the identities for each requested key and α is less than d.

Suppose $\mathcal{A}$ requests a private key γ where $|\gamma \cap \alpha| < d$. We define the following three sets:

$$\Gamma = \gamma \cap \alpha,$$

$$\Gamma' \text{ any set such that } \Gamma \subset \Gamma' \subset \gamma \text{ and } |\Gamma'| = d - 1$$

and

$$S = \Gamma' \cup \{0\}.$$

Next, we define the decryption key components, D_i, for $i \in \Gamma'$ as follows:

$$\text{If } i \in \Gamma: \ D_i = g^{s_i} \text{ where } s_i \text{ is chosen randomly in } \mathbb{Z}_p.$$

$$\text{If } i \in \Gamma' - \Gamma: \ D_i = g^{\frac{\lambda_i}{w_i}} \text{ where } \lambda_i \text{ is chosen randomly in } \mathbb{Z}_p.$$

The intuition behind these assignments is that we are implicitly choosing a random $d-1$ degree polynomial $q(x)$ by choosing its value for the $d-1$ points randomly in addition to having $q(0) = a$. For $i \in \Gamma$, we have $q(i) = c\beta_i s_i$ and for $i \in \Gamma' - \Gamma$ we have $q(i) = \lambda_i$.

Algorithm $\mathcal{B}$ can calculate the other D_i values where $i \notin \Gamma'$ since the simulator knows the discrete log of T_i $\forall i \notin \alpha$. The algorithm makes the assignments as follows:

$$\text{If } i \notin \Gamma': \ D_i = (\prod_{j \in \Gamma} C^{\frac{\beta_j s_j \Delta_{j,S}(i)}{w_i}})(\prod_{j \in \Gamma' - \Gamma} g^{\frac{\lambda_j \Delta_{j,S}(i)}{w_i}}) Y^{\frac{\Delta_{0,S}(i)}{w_i}}.$$

By interpolating, the algorithm is able to calculate $D_i = g^{\frac{q(i)}{t_i}}$ for $i \notin \Gamma'$ where $q(x)$ was implicitly defined by the random assignment of the other $d-1$ variables $D_i \in \Gamma'$ and the variable Y. Therefore $\mathcal{B}$ is able to construct a private key for the identity γ. Moreover, the distribution of the private key for γ is the same as that of the original scheme.

Challenge The adversary $\mathcal{A}$ will submit two challenge messages M_1 and M_0 to $\mathcal{B}$. Algorithm $\mathcal{B}$ flips a fair binary coin ν and returns an encryption of M_ν. The ciphertext is output as

$$E = (\alpha, E' = M_\nu Z, \{E_i = B^{\beta_i}\}_{i \in \alpha}).$$

If $\mu = 0$, then $Z = e(g,g)^{\frac{ab}{c}}$. If we let $r' = \frac{b}{c}$, then we have $E_0 = M_\nu Z = M_\nu e(g,g)^{\frac{ab}{c}} = M_\nu e(g,g)^{ar'} = M_\nu Y^{r'}$ and $E_i = B^{\beta_i} = g^{b\beta_i} = g^{\frac{b}{c} c\beta_i} = g^{r' c\beta_i} = (T_i)^{r'}$. Therefore, the ciphertext is a random encryption of the message m_ν under the public key α.

Otherwise, if $\mu = 1$, then $Z = g^z$. We then have $E' = M_\nu e(g,g)^z$. Since z is random, E' will be a random element of $\mathbb{G}_2$ from the adversary's view and the message contains no information about M_ν.

Phase 2 The algorithm $\mathcal{B}$ acts exactly as it did in Phase 1.

Guess Adversary $\mathcal{A}$ will submit a guess ν' of ν. If $\nu = \nu'$, algorithm $\mathcal{B}$ will output $\mu' = 0$ to indicate that it was given an MBDH-tuple. Otherwise, it will output $\mu' = 1$ to indicate it was given a random 4-tuple.

As shown in the construction, algorithm $\mathcal{B}$'s generation of public parameters and private keys is identical to that of the actual scheme.

In the case where $\mu = 1$, the adversary gains no information about ν. Therefore, we have $\Pr[\nu \neq \nu' | \mu = 1] = \frac{1}{2}$. Since $\mathcal{B}$ guesses $\mu' = 1$ when $\nu \neq \nu'$, we have $\Pr[\mu' = \mu | \mu = 1] = \frac{1}{2}$.

If $\mu = 0$ when $\nu = \nu'$, we have $\Pr[\mu' = \mu | \mu = 0] = \frac{1}{2} + \epsilon$.

The overall advantage of algorithm $\mathcal{B}$ in the DMBDH game is $\frac{1}{2}\Pr[\mu' = \mu | \mu = 0] + \frac{1}{2}\Pr[\mu' = \mu | \mu = 1] - \frac{1}{2} = \frac{1}{2}(\frac{1}{2} + \epsilon) + \frac{1}{2} \cdot \frac{1}{2} - \frac{1}{2} = \frac{1}{2}\epsilon$. □

References

[1] A. Sahai and B. Waters, *Fuzzy identity based encryption.* Eurocrypt 2005.

[2] U. Uludag, S. Pankanti, S. Prabhakar and A. Jain. *Biometric Cryptosystems: Issues and Challenges.* Proceedings of the IEEE, Vol. 92, NO. 6, June 2004.

[3] S. Raghavan, *Biometric authentication and encryption.*[Online]. Available http://www.cs.washington.edu/education/courses/csep590/06wi/finalprojects/raghavan.doc

[4] A. Juels and M. Wattenberg. *A Fuzzy Commitment Scheme.* In G. Tsudik, editor, Sixth ACM Conference on Computer and Communications Security, pages 28-36. *ACM Press, 1999.*

[5] A. Juels and M. Sudan. *A Fuzzy Vault Scheme.* In International Symposium on Information Theory (ISIT), page 408, IEEE Press. 2002.

[6] US global piracy losses estimated at $9.2B in 2002, IDG News Service. (2003, February 14). [Online]. Available http://www.computerworld.com/securitytopics/security/cybercrime/story/0,10801,78545,00.html

[7] Four out of every ten software programs are pirated worldwide, Business Software Alliance. (2002, June 10). [Online]. Available: http://global.bsa.org/usa/press/newsreleases//2002-06-10.1129.phtml

[8] Advanced encryption standard (AES), Federal information processing standards publication 197, National Institute of Standards and Technology. (2001). [Online]. Available: http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf