Technische Universität Braunschweig

Institute of Operating Systems and Computer Networks

Secure communication based on noisy input data

Fuzzy extractors and identity based encryption

Stephan Sigg

June 10, 2011

Overview and Structure

05.04.2011	Organisational
15.04.2011	Introduction
19.04.2011	Classification methods (Basic recognition, Bayesian, Non-parametric)
26.04.2011	Classification methods (Linear discriminant, Neural networks)
03.05.2011	Classification methods (Sequential, Stochastic)
10.05.2011	Feature extraction from audio data
17.05.2011	Feature extraction from the RF channel
24.05.2011	Fuzzy Commitment
31.05.2011	Fuzzy Extractors
07.06.2011	Error correcting codes
21.06.2011	Entropy
28.06.2011	Physically unclonable functions

Outline

Introduction

Fuzzy extractors describe a general approach for handling biometric data in cryptographic applications

It is assumed as little as possible about the biometric data.

An adversary may know more about a distribution than we do.

For instance,...
...we could determine the mean of a binomial distribution by taking n samples with accuracy of $\approx \frac{1}{\sqrt{n}}$.
A well motivated adversary could take more samples and determine the mean more accurately.

Theorem
text

Introduction

A large body of cryptographic literature describes means to obtain security provided a secret, uniformly random and reliably reproducible random string

Such string can be used to create a secret key or serve as a seed to generate a public/private key pair

If biometric inputs can be converted to such strings, a wide array of cryptographic techniques can be used to provide security from biometrics

A fuzzy extractor is a general term for a cryptographic system which takes noisy biometric input data and transforms it into a secure key

Introduction

A fuzzy extractor is a primitive that extracts a uniformly random string R from from its input w in a noise-tolerant way

If the input changes to some w' but still remains close, R can be reproduced

To improve reproduction accuracy, a public helper string P is also generated at the first execution

Introduction

In order to construct fuzzy extractors, a secure sketch is utilised

At an input w a procedure outputs a sketch s

Given s and a value w' which is close to w, it is possible to recover w

Secure in the sense that w retains its entropy although s is known.

Suitable representation and distance metric is defined based on a concrete applications requirements

Introduction

Can be applied in fields beyond biometrics, such as

- noisy inputs from human memory
- drawings used as passwords
- keys from quantum channels
- keys from noisy channels
- ...

Outline

Fuzzy extractors and secure sketches

A secure sketch enables the recovery of a string $w \in M$ from any close string $w' \in M$

Definition

An $(m, \overline{m}, t)$-secure sketch is a pair of efficient randomised procedures $(\mathtt{SS}, \mathtt{Rec})$ such that the following hold:

- **Sketch:** $\mathtt{SS}$ takes an input $w \in M$ and returns $s \in \{0,1\}^*$. $\mathtt{Rec}$ takes an element $w' \in M$ and $s \in \{0,1\}^*$
- **Correctness:** If $\mathtt{dis}(w, w') \leq t$, then $\mathtt{Rec}(w', \mathtt{SS}(w)) = w$
- **Security:** The min-entropy of w given s is high.

The entropy loss of a secure sketch is $m - \overline{m}$

Fuzzy extractors and secure sketches

A fuzzy extractor does not necessarily recover the original input.

It enables the generation of a close-to-uniform string R from w and its reproduction given any w' close to w

Definition

An (m, l, t, ϵ)-fuzzy extractor is a pair of procedures `Gen`,`Rep` with

Operation: `Gen` outputs a string $R \in \{0,1\}^*$ and a helper string P from w. `Rep` takes an element $w' \in M$ and $P \in \{0,1\}^*$

Correctness: If $\texttt{dis}(w, w') \leq t$ and $(R, P) \leftarrow \texttt{Gen}(w)$, then $\texttt{Rep}(w', P) = R$

Security: The string R is nearly uniform even given P.

The entropy loss of a fuzzy extractor is $m - l$

Fuzzy extractors and secure sketches

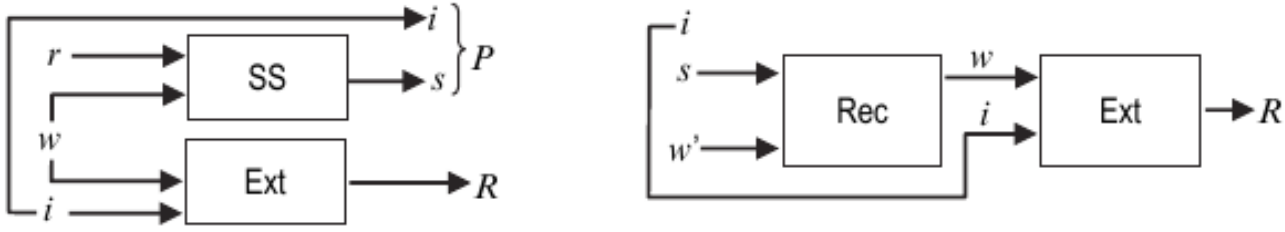
Lemma

Suppose we compose an $(m, \overline{m}, t)$-secure sketch (SS, Rec) for a space M and a universal hash function $Ext : M \to \{0,1\}^l$ as follows:

In Gen, choose a random i and let $P = (SS(w), i)$ and $R = Ext(w, i)$.

Let $Rep(w', (s, i)) = Ext(Rec(w', s), i)$.

The result is a fuzzy extractor.



Fuzzy extractors and secure sketches

Secure sketches – Hamming distance

Construction 1 – Code-offset

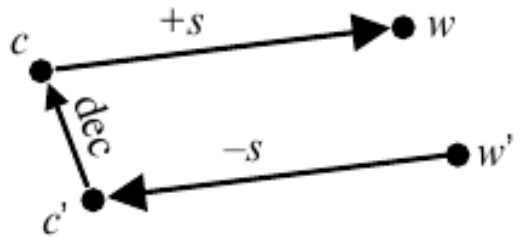
On input w select a uniformly random codeword $c \in C$

Set $\mathtt{SS}(w)$ to be the shift needed to get from c to w: $\mathtt{SS}(w) = w - c$

To compute $\mathtt{Rec}(w', s)$ subtract the shift s from w' to obtain $c' = w' - s$

Decode c' to get c

Compute w by shifting back to get $w = c + s$



Fuzzy extractors and secure sketches

Secure sketches – Hamming distance

Construction 2 – Syndrome construction

$\mathtt{SS}(w)$ computes $s = \mathtt{syn}(w)$

To compute $\mathtt{Rec}(w', s)$ find the unique vector $e \in \mathcal{F}^n$ of Hamming weight $\leq t$ such that $\mathtt{syn}(e) = \mathtt{syn}(w') - s$

Output $w = w' - e$

In a linear code, a syndrome is the multiplication of a received code word with the control matrix. It is only dependent on the error that occurred and not on the codeword transmitted. In the case of a transmission error, the syndrome identifies the incorrect bit positions which have to be corrected.

Fuzzy extractors and secure sketches

Secure sketches – Hamming distance

No secure sketch construction can have a better trade-off between error tolerance and entropy loss than construction 1

Since searching for better secure sketches for the Hamming distance is equivalent to searching for better error-correcting codes[1]

[1] Dodis, Ostrovsky, Reyzin, Smith, Fuzzy extractors: How to generate strong keys from biometrics and other noisy [illegible] Cryptology, Eurocrypt 2004, Lecture Notes in Computer Science, Vol. 2567, Pages 130-144, 2003

Fuzzy extractors and secure sketches

Secure sketches – Set difference

We consider relatively small subsets of a huge universe $\mathcal{U}$ with $|\mathcal{U}| = n$

This corresponds to representing an object by a list of its features

- Minutiae in a fingerprint
- short strings that occur in a long document
- Lists of favourite movies
- ...

The distance between two sets $w, w' \subseteq \mathcal{U}$ is the size of their symmetric difference:

$$\mathtt{dis}_{set}(w, w') =^{def} |w \triangle w'|$$

Fuzzy extractors and secure sketches

Secure sketches – Set difference

A set can be represented by its characteristic vector $x_w \in \{0,1\}^{|\mathcal{U}|}$ with 1 at positions $a \in \mathcal{U}$ if $a \in w$ and 0 else

Construction 3 – PinSketch

To compute $\mathtt{SS}(w) = \mathtt{syn}(x_w)$:

1. Let $s_i = \sum_{a \in w} a^i$ (Computations in GF(2^α))
2. Output $\mathtt{SS}(w) = (s_1, s_2, s_3, \ldots, s_{2t-1})$

To recover $\mathtt{Rec}(w', (s_1, s_2, \ldots, s_{2t-1}))$:

1. Compute $(s'_1, s'_2, \ldots, s'_{2t-1}) = \mathtt{SS}(w') = \mathtt{syn}(x_{w'})$
2. Let $\sigma = s'_i - s_i$
3. Use BCH decoding to find set v with $\mathtt{syn}(x_v) = (\sigma_1, \sigma_2, \ldots, \sigma_{2t-1})$
4. Output $w = w' \triangle v$

Fuzzy extractors and secure sketches

Secure sketches – Edit distance

Here, we assume the space of n-character strings over some alphabet $\mathcal{F}$

The distance between the strings is denoted by $\mathtt{dis}_{Edit}(w, w')$

Example applications

- Entering a password as a string
- Handwriting recognition
- Comparison of RNA/DNA-sequences

Fuzzy extractors and secure sketches

Secure sketches – Edit distance

It is difficult to work in the space spanned by the edit distance directly.

Instead, the edit distance is transformed to some other distance metric while approximately preserving the distance

Example: Mapping onto the Hamming distance[2]

Mapping: ϕ : Edit distance to Hamming distance

- If $\mathtt{dis}_{Edit}(w, w') \leq t$,
- then $\mathtt{dis}_{Ham}(\phi(w), \phi(w')) \leq tD$
- with $D = 2^{\mathcal{O}\left(\sqrt{\log n \log\log n}\right)}$

Then: Utilise Construction 1 for Hamming distance

[2] Ostrovsky, Rabani, Low distortion embeddings for edit distance, In Proceedings of the 37th annual ACM Symposium on Theory of Computing, 2005.

Fuzzy extractors and secure sketches

Secure sketches – Edit distance

It is difficult to work in the space spanned by the edit distance directly.

Instead, the edit distance is transformed to some other distance metric while approximately preserving the distance

Similar mappings also proposed for set difference[3]

[3] Broder, On the resemblance and containment of documents, Proceedings of compression and Complexity of

Outline

Robustness and active attacks

The appeal of biometric data as cryptographic secret stems from their

1. high entropy
2. availability to their owner
3. relative immunity to loss

Some challenges are

1. Each person is equipped only with a limited supply of non-renewable biometric entropy
2. The design of protocols over adversarial controlled channels is complicated by the lack of storage on the user side

Robustness and active attacks

However, in the face of an active adversary, fuzzy extractors might not be sufficient to establish a secure communication channel

1. The adversary could modify P maliciously
   - On a storage server
   - While in transit to the user
   - By clever modification of P, adversary might trick user to provide information about her secret
   - When the adversary has control over the error model applied
2. When biometric data is re-used for authentication with several distinct parties, the security of the protocol could be challenged
   - By a collusion attack under the knowledge of several $P_1, \ldots, P_n$

Robustness and active attacks

Robust secure sketch

In order to secure a fuzzy extractor against modification of P, the recovery algorithm is modified to abort with high probability in the case of tampering.

Definition

A (m, m', t, n, δ)-robust sketch is a well-formed (m, m', t)-secure sketch $(\mathtt{SS}, \mathtt{Rec})$ such that for adversaries A the advantage in the following game, $\mathbb{P}(\mathit{Success})$ is $\leq \delta$.

Registration: A is given a string $s \leftarrow \mathtt{SS}(w_0)$

Trials: A outputs n strings $s_1, \ldots, s_n$

Success: A wins if $\exists i, (s_i \neq s) \wedge \mathtt{Rec}(w_i, s_i) \neq \perp$

Robustness and active attacks

Generic robust secure sketch

Any well-formed secure sketch (SS',Rec') can be transformed into a robust sketch (SS,Rec) using a Hash function $H : \{0,1\}^* \to \{0,1\}^k$ (viewed as a random oracle)

SS(w)

1. $s' \leftarrow \mathrm{SS}(w)$
2. $h = H(w, s')$
3. return $s = (s', h)$

Rec(w, s)

1. parse s as (s', h)
2. $w' \leftarrow \mathrm{Rec}'(w, s')$
3. if $w' = \perp$ output $\perp$
4. if $H(w', s') \neq h$ output $\perp$
5. otherwise, output w'

Robustness and active attacks

Generic robust secure sketch

The intuition behind this construction is that the Hash function acts as a secret key authenticator for the public string P

Any tampering will be detected with overwhelming probability

Robustness and active attacks

Robust fuzzy extractor

Definition

A $(m, l, t, \varepsilon, n, \delta)$-robust fuzzy extractor is a (m, l, t, ε)-fuzzy extractor $(\mathtt{Ext}, \mathtt{Rep})$ such that for adversaries A the advantage in the following game, $\mathbb{P}(\mathit{Success})$, is $\leq \delta$.

Registration: A is given a pair $(R, P) \leftarrow \mathtt{Ext}(w_0)$

Trials: A outputs n strings $P_1, \ldots, P_n$

Success: A wins if $\exists i, (P_i \neq P) \wedge \mathtt{Rep}(w_i, P_i) \neq \perp$

Robustness and active attacks

Generic robust fuzzy extractors

Given a robust secure sketch $(\mathtt{SS}, \mathtt{Rec})$, we can obtain a robust fuzzy extractor by applying a second Hash function $G : M \rightarrow \{0,1\}^l$ (viewed as a random oracle)

If the robust sketch rejected the input, the robust fuzzy extractor will also reject

$\mathtt{Ext}(w)$

1. $s \leftarrow \mathtt{SS}(w)$
2. $R = G(w)$
3. return $(R, P = s)$

$\mathtt{Rec}(w, P)$

1. $w' \leftarrow \mathtt{Rec}(w, s)$
2. if $w' = \perp$ output $\perp$
3. otherwise, output $G(w')$

Robustness and active attacks

Towards reusability

Since people have a limited number of biometric features, there is a strong motivation to reuse the same features with multiple parties

Reusability is challenging since each instance P_i obtained from a new reading w_i will reveal additional information about it.

In order to guard against this problem, it is necessary to ensure that additional readings $P_i \leftarrow \mathrm{SS}(w_i)$ do not leak additional information that was not already present in $P_0 \leftarrow \mathrm{SS}(w_0)$

Robustness and active attacks

Towards reusability

Assume that w_i are obtained by deterministic functions $w_i = \phi_i(w)$

Definition

A secure sketch $(\mathtt{SS}, \mathtt{Rec})$ is (m, m', t, ϕ)-secure against a chosen perturbation attack if the advantage of any adversary A in the following game, $\mathbb{P}(\text{Success})$ is $\leq 2^{-m'}$

Preparation: A publishes an efficient sampling procedure for a random variable $W : \Omega \to M$. The challenger samples $w \leftarrow W$

Sketching: For $i = 1, 2, 3, \ldots,$ A designates a perturbation function $\phi_i \in \Phi$ and receives the sketch $P_i \leftarrow \mathtt{SS}(\phi_i(w))$

Guessing: A outputs $w' \in M$. A has Success if $w' = w$

Robustness and active attacks

Towards reusability

The perturbation security for fuzzy extractors is defined similarly.

Only changes:

1. A receives P_i from $(R_i, P_i) \leftarrow \mathtt{Ext}(\phi_i(w))$ in response to queries
2. A guesses the value of R_0 such that $(R_0, P_0) = \mathtt{Ext}(w)$

Robustness and active attacks

Reusable Sketches from Symmetries

Let $C' \subset M$ be an error-correcting code.

Let Q be a group of permutations $\pi : M \to M$

(closed under composition and inversion)

An element $p_Q \in C'$ is an Q-pivot if $\forall \pi \in Q$, $\pi(p_Q) \in C'$

All permutations $\pi \in Q$ form a subcode $C \subseteq C'$ that is closed under Q and on which Q acts transitively

Robustness and active attacks

Reusable Sketches from Symmetries

Let $(E \circ D) : M \to C$ map elements of M to the closest codeword in C'

Fix a Q-pivot $p_Q \in C'$ and construct

SS(w)

1. pick $\pi_1 \in Q'$ such that $\pi_1(w) = p_Q$
2. pick $\pi_2 \in Q$ at random
3. construct $\pi = \pi_2 \circ \pi_1 \in Q'$
4. output $s = \pi$

Rec(w',s)

1. interpret s as $\pi \in Q'$
2. let $\pi' = \pi^{-1} \circ E \circ D \circ \pi$
3. output $w = \pi'(w')$

Outline

Fuzzy identity based encryption

Fuzzy identity based encryption

In fuzzy Identity based encryption we view an identity as a set of descriptive attributes.

A fuzzy identity based encryption scheme allows for a private key for an identity w' iff the identities w and w' are close to each other

Fuzzy identity based encryption

In practice, it might be difficult to find biometrics that are

1. easy to scan
2. difficult for an adversary to obtain

Identity based encryption allows for a sender to encrypt a message to an identity without access to a public-key certificate.

A user with secret key for identity w can decrypt a cipher-text encrypted with public key w' iff w and w' are within a certain distance.

The biometric measurement for an individual need not be kept secret

Only: Attacker must not be able to fool key authority into believing that it owns a biometric identity that it does not.

Since biometric information is public, it must be robust against collusion attacks

Fuzzy identity based encryption

In identity based encryption a user's private key is produced as a set of private-key components

For each attribute in the user's identity, one private-key component is produced

Basically, the method of secret sharing is exploited[4]

[4] Shamir, How to share a secret, Communications of the association for computing machinery, 22(11), pp 612-613, 1979

Fuzzy identity based encryption

Exkurs: Secure a shared secret

Divide data D into n pieces such that D is reconstructible from any k pieces while knowledge of up to $k-1$ pieces exposes no information about D

Solution by polynomial interpolation[5]

Given n points in the 2-dimensional plane $(x_1, y_1), \ldots, (x_n, y_n)$ with distinct x_i, there is exactly one polynomial $q(x)$ of degree $k-1$ such that $q(x_i) = y_i$ for all i

We assume that the data D is or can be made a number

[5] Shamir, How to share a secret, Communications of the association for computing machinery, 22(11), pp 612-613, 1979

Fuzzy identity based encryption

Exkurs: Secure a shared secret

Solution by polynomial interpolation[6]

To divide it into pieces D_i we pick a random $k-1$ degree polynomial $q(x) = a_0 + a_1 x + \cdots + a_{k-1} x^{k-1}$ with $a_0 = D$ and evaluate $D_1 = q(1), D_2 = q(2), \ldots, D_n = q(n)$

Given any subset of k of these D_i we can find the coefficients of $q(x)$

(Solving an equation with k unknown values)

With only $k-1$ values, however, it is not feasible to calculate D.

By altering the underlying polynomial, we can even change the D_i pieces at any time without changing D

[6] Shamir, How to share a secret, Communications of the association for computing machinery, 22(11), pp 612-613, 1979

Fuzzy identity based encryption

Fuzzy Selective-ID

Init : Adversary A declares the identity, α, that she wishes to be challenged

Phase 1 : A is allowed to issue queries for private keys for many identities γ_j with $|\gamma_j \cap \alpha| < d$ for all j

Challenge : A submits equal length messages M_0, M_1. Challenger flips random coin b and encrypts M_b with α. Cipher-text is passed to A.

Phase 2 : Repeat Phase 1

Guess : A outputs a guess b' of b

The advantage of A in this game is defined as $P[b' = b] - \frac{1}{2}$

Fuzzy identity based encryption

Fuzzy Selective-ID

Definition

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 above game.

Fuzzy identity based encryption

Fuzzy Selective-ID

We view identities as sets of attributes and let d represent the error tolerance in terms of minimal set overlap.

When an authority creates a private key, it associates a random $(d-1)$ degree polynomial $q(x)$ with each user

For each of the attributes, the key generation algorithm issues a private key component that is tied to the user's random polynomial $q(x)$

If user matches at least d components, she can obtain the secret

Since the private-key components are tied to random polynomials, multiple users are unable to combine them for collusion attacks.

Fuzzy identity based encryption

Fuzzy Selective-ID

Let $\mathbb{G}_1$ be a bilinear group of prime order p and g be a generator of $\mathbb{G}_1$

Additionally, let $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ denote a bilinear map with $e(g^a, g^b) = e(g, g)^{ab}$

Elements are subsets of some universe $\mathcal{U}$ of size $|\mathcal{U}| \mod (p)$.

We associate each element with a unique integer in $\mathbb{Z}_p^*$

(in practice, an attribute will be associated with each element)

Fuzzy identity based encryption

Fuzzy Selective-ID – Setup

Choose $t_1, \ldots, t_{|\mathcal{U}|}$ uniformly at random from $\mathbb{Z}_p$

The published public parameters are

$$T_1 = g^{t_1}, \ldots, T_{|\mathcal{U}|} = g^{t_{|\mathcal{U}|}}, Y = e(g,g)^y$$

The master key is

$$t_1, \ldots, t_{|\mathcal{U}|}, y$$

Fuzzy identity based encryption

Fuzzy Selective-ID – Key generation

To generate a private key for identity $w \subseteq \mathcal{U}$,

1. A $(d-1)$st degree polynomial q with $q(0) = y$ is randomly chosen
2. Private key consists of components $\{D_i\}_{i \in w}$ with $D_i = g^{q(i)/t_i}$ for every $i \in w$

Fuzzy identity based encryption

Fuzzy Selective-ID – Encryption

Encryption with the public key w' of a message $M \in \mathbb{G}_2$ proceeds as follows

1. A random value $s \in \mathbb{Z}_p$ is chosen
2. The cipher-text is published as

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

Fuzzy identity based encryption

Fuzzy Selective-ID – Decryption

Given a private key w with $|w \cap w'| \geq d$

Choose an arbitrary d-element subset S of $w \cap w'$

The cipher-text can be decrypted as

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

Input:

$D_i = g^{q(i)/t_i}$

$\{E_i = T_i^s\}_{i \in w'}$

$E' = MY^s$

Outline

Questions?

Stephan Sigg
sigg@ibr.cs.tu-bs.de

Literature

C.M. Bishop: Pattern recognition and machine learning, Springer, 2007.

P. Tulys, B. Skoric, T. Kevenaar: Security with Noisy Data – On private biometrics, secure key storage and anti-counterfeiting, Springer, 2007.

W.W.Peterson, E.J. Weldon, Error-Correcting Codes, MIT press, 1972.

R.O. Duda, P.E. Hart, D.G. Stork: Pattern Classification, Wiley, 2001.