A Fuzzy Vault for Behavioral Authentication Systems

Md Morshedul Islam

January 2019

Abstract

1 Introduction

Fuzzy Vault.
Fuzzy Vault Security.
Fuzzy Vault Implementation.
Behavioral Authentication (BA) Systems.

1.1 Our Work

BA-Vault.
BA-Vault Security.
Unlocking BA-Vault.
Experiments.
Ethics approval.

1.2 Related Works

Paper organization.

2 Background

2.1 BA System

Profile in BA Systems.

2.2 Data Transformation

Random Projection.

2.3 Hypothesis Test

T-test.
f-test.
Fisher's Method.

2.4 Fuzzy Vault

Fuzzy Vault Security.
Fuzzy Vault Implementation.

3 BA-vault

Definition 3.1. (BA-vault) BA-vault uses behavioral data to lock and unlock the vault. A BA-vault $\texttt{BAVault}(\mathbb{F}_q, r, t, k)$ is a 4-tuples system,

1. A secret message m defines a polynomial $f(x)$ of degree k over $\mathbb{F}_q[x]$.
2. The polynomial is then evaluated on a *locking set* A of $t \geq k+1$ features point $\{x_i\}_{i=1}^{t}$ collected from a BA system profile $\mathbf{X}$. The evaluation of the polynomial by set A form a set of t pairs $\{(x_i, f(x_i))\}_{i=1}^{t}$ called legitimate points set.
3. To hide the legitimate points, the `BAVault` system mixed $r-t$ chaff points $\{(\alpha_i, \beta_i)\}_{i=t+1}^{r}$ with the t legitimate points which form a BA-vault $\mathbf{V}_{BA}$. In the chaff points $f(\alpha_i) \neq \beta_i$.
4. The `BAVault` system publish the vault $\mathbf{V}_{BA}$ along with the parameters $(\mathbb{F}_q, r, t, k)$.
5. To unlock the vault $\mathbf{V}_{BA}$, the `BAVault` system uses a second set of features point called *unlocking set* $B = \{y_i\}_{j=1}^{t'}$. All feature points of B is collected from $\mathbf{Y}$, a new BA system profile.
6. A matching algorithm $M(.,.)$ compares the features points of B with $\{x_i/\alpha_i\}_{i=1}^{r}$ of $\mathbf{V}_{BA}$ and outputs a set $C \subset V_A$. The unlocking attempt will be successful if C has more than $k+1$ feature points of A.

Two BA-vault algorithms Lock and Unlock are used to lock and unlock the vault, where $M(.,.)$ is the part of the algorithm Unlock. The order of the feature points on the sets A and B does not have any impact on the locking and unlocking procedures. In a BA-vault, locking set $A \in \mathbb{F}_q^t$, unlocking set $B \in \mathbb{F}_q^{t'}$, secret $m \in \mathbb{F}_q^k$ and fuzzy vault $V_A \in \mathbb{F}_q^r$. Also, in our proposed BA-vault the size of locking and unlocking set are the same ($t = t'$).

Definition 3.2. (BA-vault completeness property) In a `BAVault` system, the pair $(\textsc{Lock}, \textsc{Unlock})$ with parameter set (r, t, k) is complete with ϵ-fuzziness if the following two conditions hold. For every secret m, every profiles pair $(\mathbf{X}, \mathbf{Y})$ and a statistical distance function $\mathit{Dist}(.,.)$, (i) if $\mathit{Dist}(\mathbf{X} - \mathbf{Y}) \leq \epsilon$ for an small constant ϵ, it is the case that $\textsc{Unlock}(\mathbf{Y}, \textsc{Lock}(\mathbf{X}, m)) = m$ with overwhelming probability, and (ii) if $\mathit{Dist}(\mathbf{X} - \mathbf{Y}) > \epsilon$ then $\textsc{Unlock}(\mathbf{Y}, \textsc{Lock}(\mathbf{X}, m)) \neq m$ with overwhelming probability.

To recover the secret m from $\mathbf{V}_{BA}$, the claimed profile $\mathbf{Y}$ need to close to the profile $\mathbf{X}$. In this case, the set B will be close to the set A, i.e., the locking and unlocking set will have substantial overlap.

3.1 BA-vault Structure

Lock a BA-vault. Figure 1 shows the block diagram of BA-vault locking process. Vault locking follows the following steps.

1. *Polynomial construction.* To hide a secret m, the `BAVault` system appends some CRC bit with m and divide the extended message to $m' = [m_0, m_1, \cdots, m_k]$. It then encoded those m_i as the coefficients of a polynomial $f(x)$ of degree k in field $\mathbb{F}_q$.

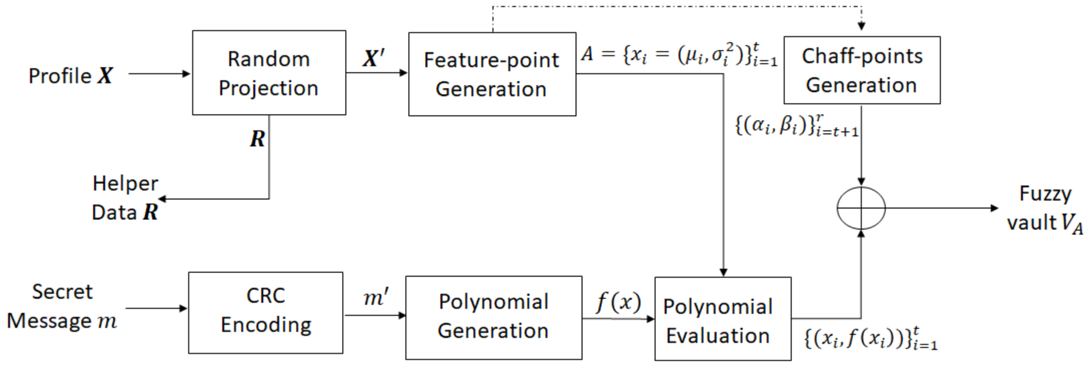

Figure 1: BA-vault locking process

2. *Capture a profile and RP.* The `BAVault` system constructs a profile $\mathbf{X}$ from the user's behaviour data and then uses RP to transform the profile $\mathbf{X}$ to $\mathbf{X}'$. Here, RP reduces the size of the profile from $\mathbb{R}^{n \times d}$ to $\mathbb{R}^{n \times t}$, where $d \geq t$. After RP, the distribution of all feature $F_i' \subset \mathbf{X}'$ becomes to the normal distribution [1].

3. *Publish helper data H_p.* The `BAVault` system publish random matrix $\mathbf{R}$ as a helper data. As the random matrix $\mathbf{R}$ is not related to the user's behavior, it does not leak anything about the profile $\mathbf{X}$.

4. *Feature points generation.* BA-vault represents the distribution of the features F_i' by their *means* and *variances* as $F_i' = (\mu_i, \sigma_i^2)$. All the pairs are then concatenate to the feature points $x_i = \mu_i \sigma_i$ (σ_i is standard deviation) and mapped them to the field $\mathbf{F}_q$ (see Section **??**). A (projected) profile $\mathbf{X}'$ produces a set of t feature points $\{x_1, x_2, ..., x_t\}$ called *locking set* A.

5. *Polynomial evaluation.* The polynomial $f(x)$ are evaluated at points A to produce $f(x_i)$. All feature points x_i and their evaluation form a set $\{(x_i, f(x)))\}_{i=1}^t$ called *legitimate points set.*

6. *Chaff point generation.* To hide the legitimate points, `BAVault` system added $r - t$ chaff points $\{(\alpha_i, \beta_i)\}_{i=t+1}^r$, where α_i is the concatenation of any random μ_i and σ_i and also $\beta_i \neq f(\alpha_i)$. All α_i will satisfy the minimum distance requirements (see Section **??**).

7. The union of legitimate points $\{(x_i, f(x_i))\}_{i=1}^t$ and chaff points $\{(\alpha_i, \beta_i)\}_{i=t+1}^r$ are then form a BA-vault $\mathbf{V}_{BA}$. The `BAVault` system scrambles all the pairs of $\mathbf{V}_{BA}$ to destroy the order information and publishes the data as a BA-vault $\mathbf{V}_{BA}$.

Unlock a BA-Vault. Figure 2 shows the block diagram of BA-vault unlocking process. BA-vault unlocking process follows the following steps:

1. *Capture the claimed profile and RP.* To unlock the BA-vault $\mathbf{V}_{BA}$, `BAVault` system captures the claimed profile $\mathbf{Y}$ and project the profile $\mathbf{Y}$ to $\mathbf{Y}'$. For RP, the vault system uses the same $\mathbf{R}$ that was published as helper data H_p. In $\mathbf{Y}'$, the distribution of $F_j' \subset \mathbf{Y}'$ are also normal/near to normal and the size of $\mathbf{Y}'$ is $\mathbb{R}^{n' \times t'}$.

2. *Feature points generation.* The `BAVault` system represents the distributions of the features $F_j' \subset \mathbf{Y}'$ by (μ_j, σ_j^2) and then concatenate them to feature points $y_j = \mu_j \sigma_j$ over $\mathbb{F}_q$. All t' ($t' = t$) feature points $y_1, y_2, ..., y_t'$ of $\mathbf{Y}'$ will form a *unlocking set* B.

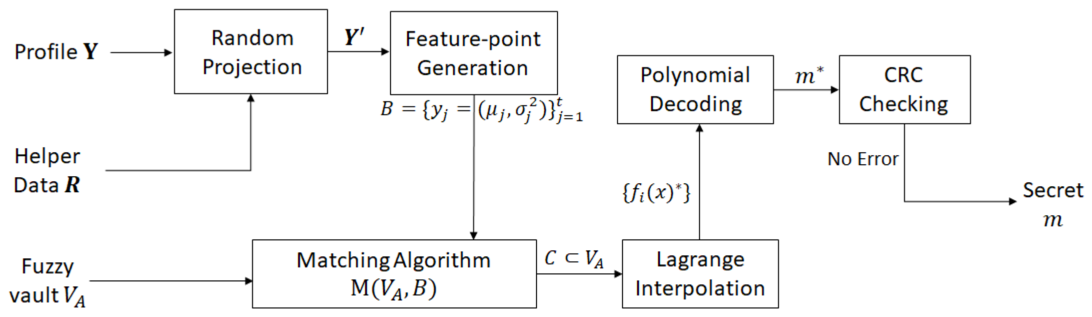

Figure 2: BA-vault unlocking process

3. *Separate legitimate points from the BA-vault.* The BA-vault unlocking algorithm UNLOCK uses a soft decision matching algorithm $M(.,.)$ to recover legitimate points and it works in two steps. In Step 1: $M^1(.,.)$ measures the similarity of each element of B with all first elements of V_A, and generate an $r \times t$ matrix of p-values (confidence values). In Step 2, $M^2(.,.)$ uses an optimization algorithm to find a subset $C \subset V_A$, where $|C| = t$. Optimization algorithm selects the " best" matching subset from the matrix to recover legitimate points. To unlock $\mathbf{V}_{BA}$ the `BAVault` systems requires at least $k+1$ legitimate points in C.

4. *CRC checking and recover secret.* `BAVault` system uses Lagrange interpolation to construct a polynomial $f(x)^* = m_0^* + m_1^* x^1 + \cdots + m_k^* x^k$ from $k+1$ recovered points. It decode all possible polynomials $f_i(x)^*$ one by one, concatenate the coefficients $m_0^*, m_1^*, \cdots, m_k^*$, to obtain the secret m^*. A CRC-decoder (CRC-16) is used to reject all incorrect m^*. If the CRC matches, $m^* = m$ with high probability and the `BAVault` accepted it as secret message.

4 BA-vault Design

4.1 Mapping Feature

4.2 Feature Point Similarity

4.3 Chaff-points Generation

4.4 BA-vault Matching

5 BA-vault Security Analysis

6 Experiments on BA-vault

7 Concluding Remarks

References

[1] Ping Li, Trevor J Hastie, and Kenneth W Church. Very sparse random projections. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 287–296. ACM, 2006.