The Fuzzy Vault for fingerprints is Vulnerable to Brute Force Attack, the Collusion Attack

Bastian Fischer
Seminar Biometry & Security
b-it WS '09/'10
Overview

• Motivation
• Short reminder of the Fuzzy Vault for fingerprints
• Vulnerability to Brute Force Attack
• The Collusion Attack
• Solution to increase security: Quiz
Motivation

• Fuzzy Vault scheme is an encryption scheme, which can tolerate errors in the keys (e.g. fingerprint scanning issues)
• Since an attacker could get access to the message of a vault, he will be able to obtain the key by brute force attack
• If multiple vaults are known, it is even easier to obtain the key by just comparing x-values
• The Fuzzy Vault scheme can be improved by implementing additional features which increase Security
Reminder: Fuzzy Vault (encoding)

• Polynomial $f \in F_q[X]$ of degree $k$ identifies a secret $S$
Reminder: Fuzzy Vault (encoding)

- Polynomial $f \in \mathbb{F}_q[X]$
  - of degree $k$
  - identifies a secret $S$
- hiding $f$ using $t > k$
  - minutiae locations, the locking set $\mathcal{L}$
Reminder: Fuzzy Vault (encoding)

- Polynomial $f \in \mathbb{F}_q[X]$ of degree $k$ identifies a secret $S$
- Hiding $f$ using $t > k$ minutiae locations, the locking set $\mathcal{L}$
- Evaluating the locking set $\mathcal{L}$ with $f$ to the genuine set $\mathcal{G}$
Reminder: Fuzzy Vault (encoding)

- Generate random Chaff Points to hide the genuine set $G$
- The Vault is now:

$$\mathcal{V} = \mathcal{V}(k, t, r, F_q) = G \cup C$$
Reminder: Fuzzy Vault (open the Vault)

- \(K + 1\) genuine points must be known to open the vault and to compute its interpolating polynomial.
- False points result in a completely different curve.
Reminder: Fuzzy Vault (decode)

- A second genuine set \( g' \) of the same fingerprint might not be identical to the original genuine set (distortion, rotation)
- If the distance is not too far, an error correction algorithm makes a successful decoding possible
Vulnerability to Brute Force Attack

\[ V = V(k, t, r, F_q) = G \cup C \]

- Choose \( t + 1 \) distinct points at random
Vulnerability to Brute Force Attack

- Choose $t + 1$ distinct points at random
- Compute $f \in \mathbb{F}_q[X]$ by interpolating the chosen points.
Vulnerability to Brute Force Attack

- Choose $t + 1$ distinct points at random.
- Compute $f \in \mathbb{F}_q[X]$ by interpolating the chosen points.
- If the graph of $f$ contains $t$ vault points, output $f$, otherwise start all over again.
Brute Force Attack will finally lead to...

- Choose $t + 1$ distinct points at random
- Compute $f \in \mathbb{F}_q[X]$ by interpolating the chosen points.
- If the graph of $f$ contains $t$ vault points, output $f$, otherwise start all over again.
Vulnerability to Brute Force Attack

- Given the Vault $\mathcal{V} = \mathcal{V}(k, t, r, \mathbb{F}_q) = \mathcal{G} \cup \mathcal{C}$, the probability $P$ that a brute force attack by randomly choosing $k + 1$ vault points and interpolate them will result in the correct $f$ is:

$$\frac{1}{P} = \frac{\binom{r}{t}}{\binom{k+1}{t}} < 1.1 \cdot \left(\frac{r}{t}\right)^{k+1}, \quad \text{for } r > t > 5.$$  

- Computing the interpolation polynomial $f$ can be done in $6.5 \cdot k \cdot \log^2(k) + O(k)$ operations.

- $\sum_{i=1}^{t-k-1}(1/q)^i$ is the probability that $t$ points lie on the graph of the polynomial $f$. 
Vulnerability to Brute Force Attack

- Complexity: An Intruder with the knowledge of an intercepted $\nu$ can recover the secret $S$ in $C \cdot (r/t)^k$ operations, where $C < 8rk$.

**Proof:** Since we have $1/P = \frac{r}{(k+1) \cdot t} < 1.1 \cdot (r/t)^{k+1}$, for $r > t > 5$.

An attacker has to compute the interpolated polynomial $f$ to his randomly chosen vault points each try, which can be done in $6.5^*\log^2(k) =: K$. 
Vulnerability to Brute Force Attack

So computing all interpolated polynomials can be done in 
\[ < 7.2^*k^*\log^2(k)^*(r/t)^k \] operations.

Search a point \((U,W) \in V \setminus T\) such that \(g(U)=W\).

This requires \(r/K\) interpolations. When no point is found, discard \(T\). If it was not discarded, search for a further point, which has this properties. This step has the probability \(1/q\). If a point has been found, add it to \(T\), otherwise discard it.

This continues until the attack is finished. Adding up all Numbers of those steps with weights given by the probability of occurrence, one finds:

\[
R < 7.2 \cdot (r/t)^k \cdot k \cdot K \cdot r \cdot \sum_{j=0}^{\infty} (1/q)^j = 7.2 \cdot (r/t)^k \cdot k \cdot K \frac{rq}{K(q-1)} < 8.0 \cdot (rk) \cdot (r/t)^k
\]
Factors that influence the complexity of a b.f.a.

- Restricting the region of interest
  - irrelevant, when minutiae uniformly distributed, because \( r \) and \( t \) are scaled by the same factor, \( r/t \) and the complexity remain unchanged.

- Increasing \( k \)
  - complexity grows
  - requires large unlocking sets
  - might be a problem for fingerprint scanners

- Increasing \#Chaffpoints
  - complexity grows
  - smaller distances between points in the vault
  - can compromise error corrections during unlocking

- Reducing \( t \) in the genuine List
  - Lesser points of interest
  - Could reduce the size of the unlocking set below the required minimum
The Collusion Attack

- Possible Attack, if an attacker is aware of more than one Fuzzy Vault, e.g. a few smartcards carrying the same biometric information (fingerprint).
- Attacker can compare the x-coordinates of both templates.
- Some may be different because of different scanners, but a few might be identical.
- All found identical x-coordinates are stored during this attack, the others are recognized as Chaffpoints and removed. The remaining points result in the effective vault.
- With these x-coordinates, the polynomial could be interpolated to unlock the vault, because it could be successfully authentificated after error correction.
- More known Vaults means higher probability to find identical x-coordinates.
Fuzzy Vault with Quiz

- Additional minutiae information increases security
- Def.: Let \((X, Y) \in \mathbb{F}_q^2\) a genuine vault point and \(\alpha\) the orientation with granularity \(\pi/n\), \(n\) small.
- \(\alpha\) is used to change Y-Coordinates of the genuine vault points.
Fuzzy Vault with Quiz

- Choose random $\beta$
- $j$ encodes transformation
  
  $$T : \mathbb{F}_q \rightarrow \mathbb{F}_q$$
  
- Compute $j$ so that $j^* \pi/n = \alpha - \beta \mod \pi$.
- Now genuine points are transformed to the pattern $(X,Y',\beta)$, with $Y'=T(Y)$.
- Chaffpoints are treated the same way
More improvement ideas...

- Using more fingers
- Non-random Chaff points
  - Guaranteed distances between vault points
  - Easier to maximize the number of points in the Field
- CRC
  - Additional checksum makes decoding easier