A ROBUST ONLINE SIGNATURE BASED CRYPTOSYSTEM

Ashok K. Bhateja
Scientific Analysis Group
Defence R & D Organization, Delhi, India

Santanu Chaudhury
Department of Electrical Engineering
Indian Institute of Technology, Delhi, India

P. K. Saxena
Scientific Analysis Group
Defence R & D Organization, Delhi, India

Outline

Introduction

- Cryptography: Protect information by ensuring
  - Confidentiality
  - Integrity and
  - Authenticity
- Cryptosystem:
  - Binds plaintext x and key k using a mathematical function f
  - Ciphertext $y = f(x, k)$
  - Extraction of x or k is computationally hard
- Management and maintenance of the keys is one of the major problems in a cryptosystem
- Cryptographic keys stored in highly secure location with
  - Password
  - Personal Identification Number (PIN)

Introduction

- Signatures are used
  - Financial transactions
  - Documents
  - Verification
- Dynamic features: velocity, slope along with static (shape) features.
- Variations in online signature are more than other biometric such as fingerprint, iris, and face
- Allowing for these variations and providing protection against forgers is a challenging task.

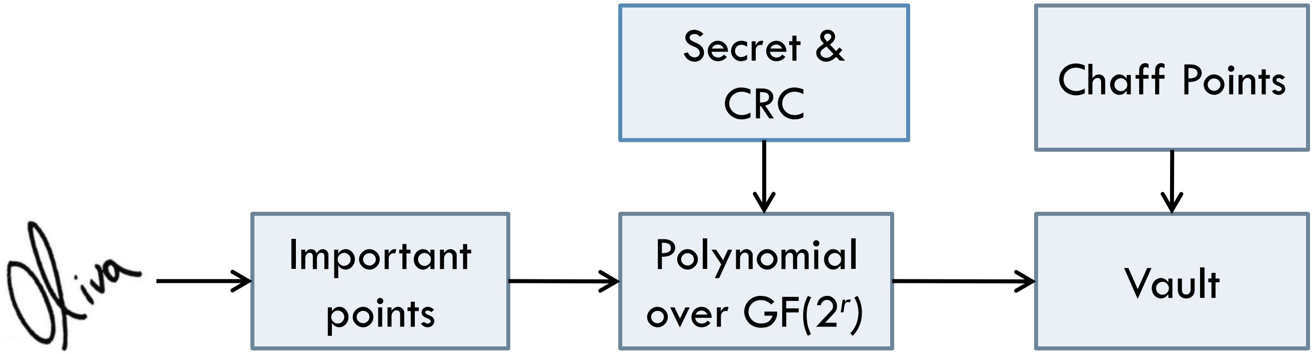
Problem Statement

- Development of a robust online signature based cryptosystem to hide the secret by binding it with important features of online signature
- Important features
  - Consistent in the genuine signature and
  - Inconsistent in the forged signature

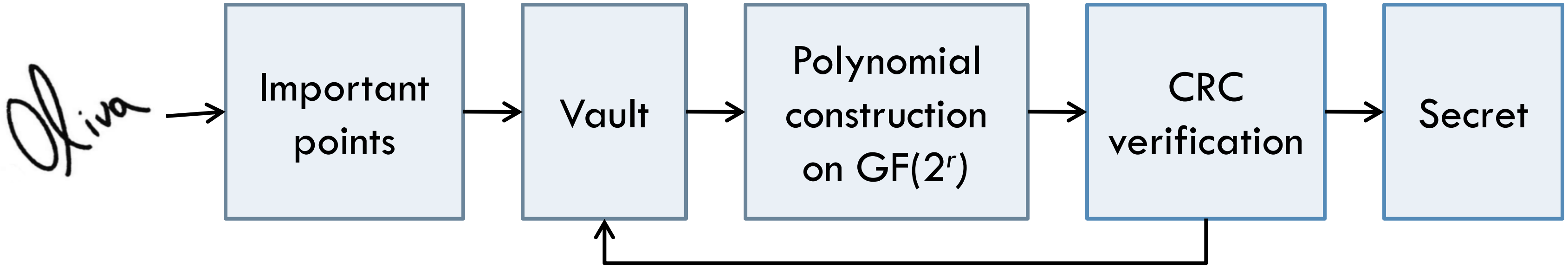
Fuzzy Vault

- Developed by Juels and Sudan [1] in 2002
- Implemented by Uludag et al. in 2005 using fingerprint biometric [2]
- Security is based on the infeasibility of the polynomial reconstruction problem
- In 2006, Kholmatov and Yanikoglu used trajectory crossing, ending and high curvature points of online signature [3] for the construction of the fuzzy vault.

Fuzzy Vault



Fuzzy Vault Encoding



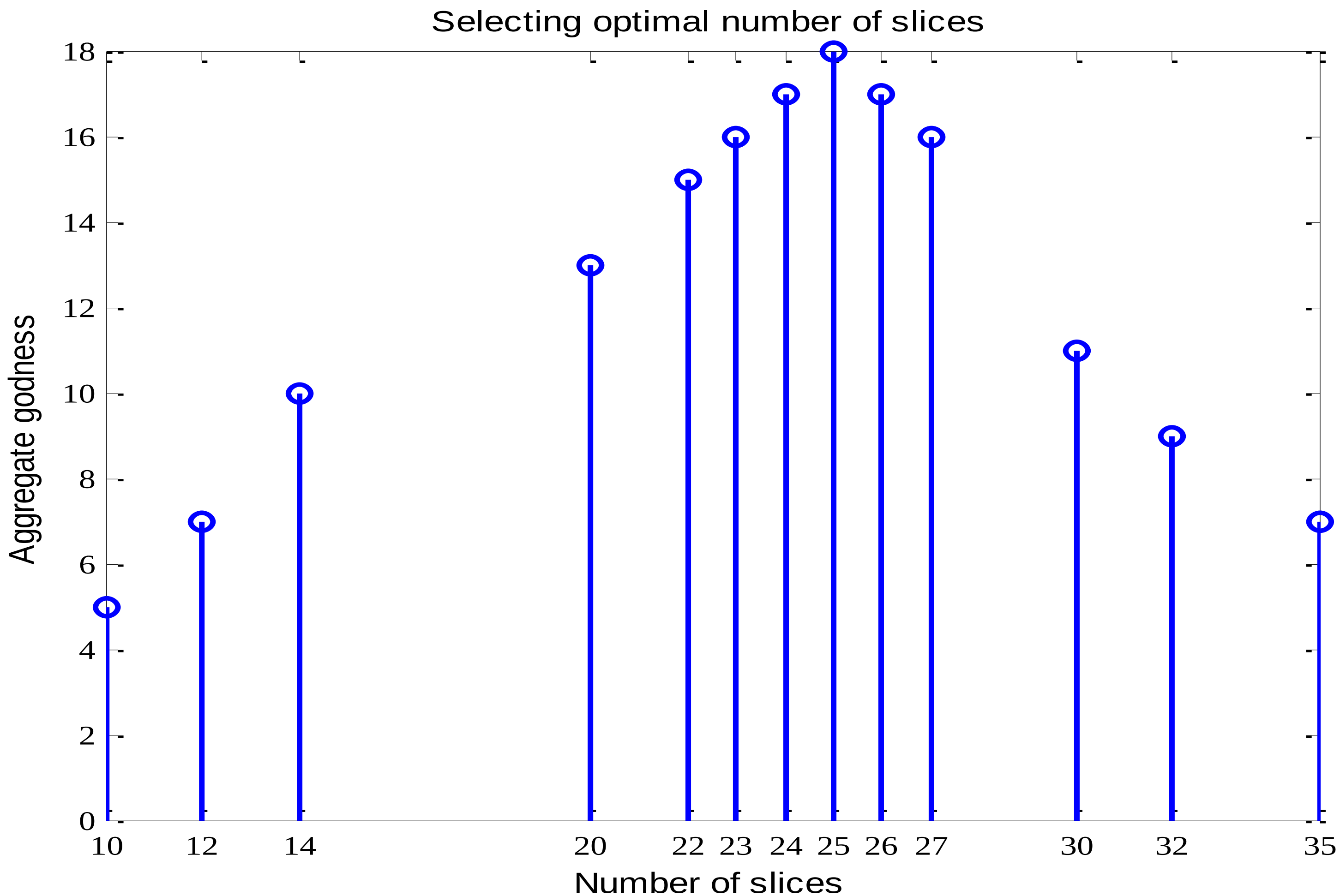
Fuzzy Vault Decoding

Proposed Online Signature Based Cryptosystem

A robust online signature based cryptosystem to hide the secret by binding it with important online signature templates

- [Slicing]: The online signature is divided into fixed number of slices ($m \times k$).
- [Feature Extraction]: Find the values of all the important features.
- [Classifiers input]: Form k sets of slices, with each set consisting of m consecutive slices. The m values of the features form the input for the classifier.
- [Training]: For each set of slices, train the networks using Weighted Back propagation with AdaBoost.
- [Encoding]: Creation of LUT
- [Decoding]: Finding secret

Optimal number of slices



Feature Extraction

- Divide the signature into n time slices

- Find S_i and S_i' i.e. sum of the variations of the genuine and forged signatures, about the mean of the genuine signature

$$S_i = \sum_{j=1}^{u} \sum_{k=1}^{s_g} \sigma_{ijk}^2 \ \& \ S_i' = \sum_{j=1}^{u} \sum_{k=1}^{s_f} {\sigma'_{ijk}}^2$$

Where σ_{ijk}^2 is variance of j^{th} user in k^{th} genuine signature in i^{th} slice and ${\sigma'_{ijk}}^2$ is the variance of j^{th} user in k^{th} forged signatures in i^{th} slice about the mean of genuine signature in the same i^{th} slice.

- Goodness function G_f of feature f

$$G_f = \frac{\sum_{i=1}^{n} S_i'}{\sum_{i=1}^{n} S_i}$$

- The features having goodness value greater than a threshold are the important features

Adaptive Boosting

- All data-points are assigned equal initial weights
- In each iteration:
  - A weak classifier is trained based on the weighted samples
  - The weights of misclassified data-points are increased
  - So next classifier gives more emphasis to data-points with more weight
- A weighted vote of selected weak classifiers is used to decide the output of the ensemble

AdaBoost – Weighted Learning

Psedocode

Given: $(x_1, y_1), \ldots, (x_m, y_m)$ where $x_i \in \mathscr{X}$, $y_i \in \{-1, +1\}$.
Initialize: $D_1(i) = 1/m$ for $i = 1, \ldots, m$.
For $t = 1, \ldots, T$:

- Train weak learner using distribution D_t.
- Get weak hypothesis $h_t : \mathscr{X} \to \{-1, +1\}$.
- Aim: select h_t with low weighted error:

$$\varepsilon_t = \Pr_{i \sim D_t}\left[h_t(x_i) \neq y_i\right].$$

- Choose $\alpha_t = \frac{1}{2} \ln\left(\frac{1 - \varepsilon_t}{\varepsilon_t}\right)$.
- Update, for $i = 1, \ldots, m$:

$$D_{t+1}(i) = \frac{D_t(i) \exp(-\alpha_t y_i h_t(x_i))}{Z_t}$$

where Z_t is a normalization factor (chosen so that D_{t+1} will be a distribution).

Output the final hypothesis:

$$H(x) = \operatorname{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right).$$

Weighted Back Propagation Algorithm

Forwards pass

- For each hidden layer and output layer neurons
  - Compute the weighted sum (*S*) of the activation of the previous layer neurons.
  - Find the activation of the neuron. i.e. sigmoid function of the sum *S*.
- Compute the error of each of the output layer neurons
- Find the weighted error i.e. weight of the training example × total error

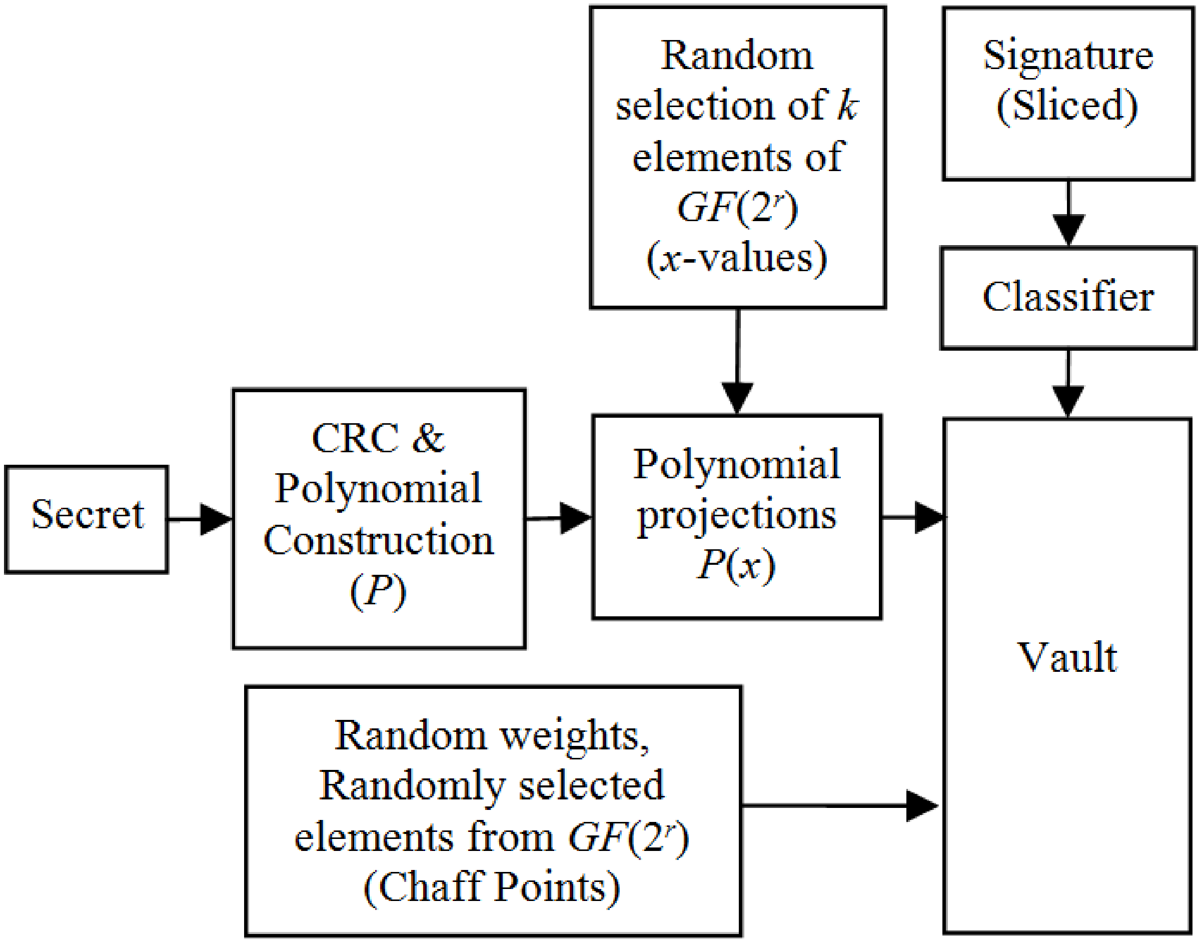
Backward pass

- Find local gradient of the neurons
- Adjust the weights.
- Iterate forward and backward pass until convergence of the network.

Online Signature Based Cryptosystem Encoding

- Creation of secret polynomial *P*.
  - Find CRC (Cyclic Redundancy Check) of the secret (*s* bits) using *r* bit generating polynomial
  - Concatenate the CRC with the secret. Let it be *SC*
  - Convert *SC* into the elements of the field.
  - Construct polynomial *P* of degree k-1 over the field $GF(2^r)$.
- Creation of LUT
  - Randomly select *k* rows of the table, one for each set of slices
  - Randomly select *k* elements of the field $GF(2^r)$, one for each set of slices. Call them *x*-values.
  - Find the polynomial projections of the *x*-values in the field
  - Store the weights of the BPNN (Back Propagation Neural Network) along with α's (importance of the classifier) in the first column of the selected row, corresponding *x*-value and their polynomial projection in second and third columns respectively
  - Fill the remaining second and third column entries of LUT by randomly selecting the elements of $GF(2^r)$
  - Fill the remaining entries of the first column by randomly generated weight values, not appearing in the selected *k* rows

Online Signature Based Cryptosystem Encoding



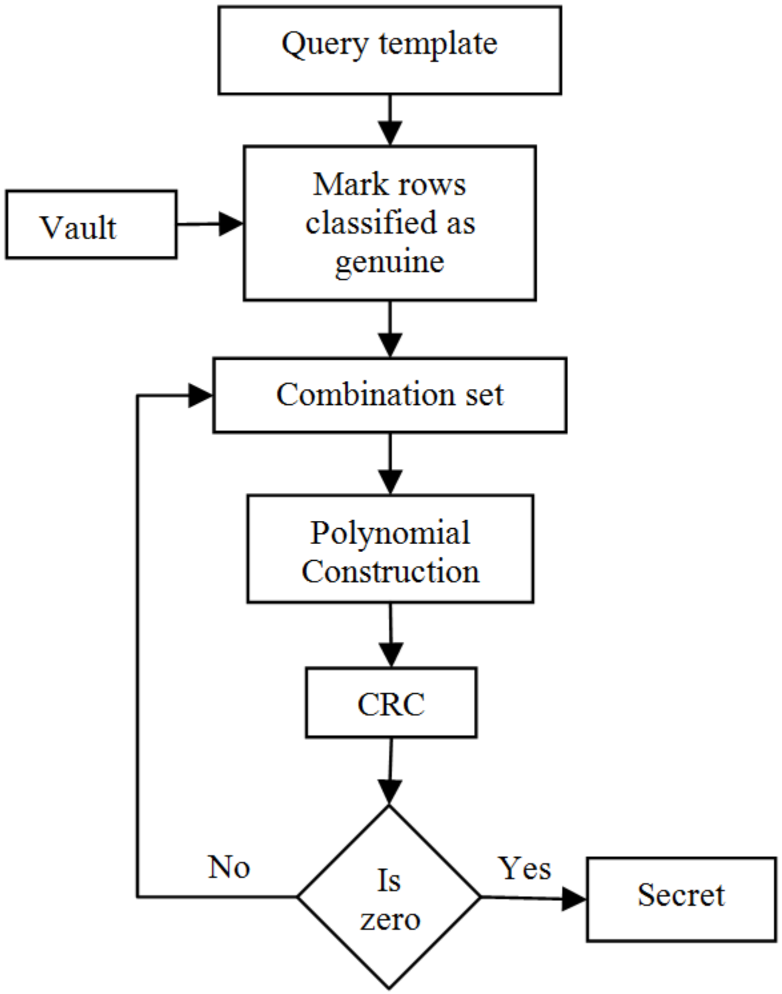
Look Up Table (Vault)

Weight & importance of classifier	r-bit random numbers in $GF(2^r)$ i.e. x	$P(x)$
⋮	⋮	⋮
$WS3$	1540	3981
⋮	⋮	⋮
$WS4$	2151	4367
⋮	⋮	⋮
$WS2$	5830	1087
⋮	⋮	⋮
$WS1$	7531	9034
⋮	⋮	⋮
$WS5$	1567	3304
⋮	⋮	⋮

Online Signature Based Cryptosystem Decoding

- Divide the query template into k sets each consisting of m slices.
- For each set of m slices, mark the rows whose classifier (weights and importance stored in the first column) classifies the signature as genuine.
- Take a combination of k pairs of $(x, P(x))$ points from the marked rows and construct the polynomial over $GF(2^r)$.
- Compute the CRC of the polynomial.
- If CRC is not zero, take another combination of k points, else stop.

Online Signature Based Cryptosystem Decoding



Experimental Results

- SVC 2004 database [11] was used.
- Total 1800 signatures of 45 users with 20 genuine and 20 forged signatures of each user were considered.
- Six important features extracted: p, v_x, v_y, v, az, al
- For training
  - 1350 signatures (15 genuine and 15 forged signatures of each user) were used.
  - A total of 1350 (6×5 for each user) networks with 5 input layer neurons, 3 hidden layer neurons and 2 output layer neurons (in each network) were trained.
- For testing
  - A set of 45 pairs of genuine-genuine were formed by selecting two genuine signatures of each person.
  - Another set of 45 pairs of genuine-forged signatures were formed by randomly selecting one genuine and one forged signature.
- 160-bits secret S: 128-bit secret + 32 bits of CRC
- Degree of polynomial over $GF(2^{32})$: 4
- 17.78% FRR and 2.22% FAR was obtained.

Conclusion

- Important features based on the consistency in the genuine signature and inconsistency in the forged signature were extracted
- Weighted back propagation algorithm is developed for training the network
- AdaBoost algorithm is used for combining the decision of the networks
- 17.78% FRR and 2.22% FAR was obtained.
- This scheme works well for all kinds of signatures without any constraint on the number of high curvature points and zero crossing points

REFERENCES

[1] A. Juels and M. Sudan, "A Fuzzy Vault Scheme," *Proceedings of IEEE International Symposium on Information Theory*, p. 408, 2002.

[2] U. Uludag, S. Pankanti, and A. K. Jain, "Fuzzy Vault for fingerprints," *Audio-and Video-Based Biometric Person Authentication. Springer Berlin Heidelberg*, pp. 310-319, 2005.

[3] A. Kholmatov and B. Yanikoglu, "Biometric cryptosystem using online signatures," *Computer and Information Sciences–ISCIS, Springer Berlin Heidelber*, pp. 981-990, 2006.

[4] H. Feng and C. C. Wah, "Online signature verification using a new extreme points warping technique," *Pattern Recognition Letters*, vol. 24.16., pp. 2943-2951, 2003.

[5] K. Huang and H. Yan, "Stability and style-variation modeling for on-line signature verification," *Pattern Recognition*, vol. 36, pp. 2253 - 2270, 2003.

[6] H. Lei, S. Palla, and V. Govindaraju, "ER2: an Intuitive Similarity Measure for On-line Signature Verification," in *9th Int'l Workshop on Frontiers in Handwriting Recognition (IWFHR-9 2004)*, 2004, pp. 191-195.

[7] J. Fierrez, J. Ortega-Garcia, D. Ramos, and J. Gonzalez-Rodriguez, "HMM-Based On-Line Signature Verification: Feature Extraction and Signature Modeling," *Pattern Recognition Letters*, vol. 28 (16), pp. 2325-2334, 2007.

[8] B. L. Van, S. Garcia-Salicetti, and B. Dorizzi, "On Using the Viterbi Path Along With HMM Likelihood Information for Online Signature Verification," *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART B: CYBERNETICS*, pp. 1237-1247 2007.

[9] C. Gruber, T. Gruber, S. Krinninger, and B. Sick, "Online Signature Verification With Support Vector Machines Based on LCSS Kernel Functions," *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART B: CYBERNETICS*, vol. 40 (4), 2010.

[10] A. Nagar and S. Chaudhury, "Biometrics based Asymmetric Cryptosystem Design using Modified Fuzzy Vault Scheme," *Proceedings of IEEE International Conference Pattern Recognition, Hong Kong, China*, vol. 4, pp. 537-540, August 2006.

[11] *Online signature database SVC 2004*. Available: http://www.cse.ust.hk/svc2004/download.html

Thank You