A match maker with weighted movie preference

Tuvshinkhuu Purevsuren and DeaHun Nyang*

*Inha University.

Abstract

Fuzzy vault with weighted feature was introduced in [4], to encrypt cryptographic key with a fuzzy information. According to weighted feature fuzzy vault scheme they mainly focused on the biometric which is human faces. Human's face image will be biometric key. Our work focused on movie matching scheme using weighted fuzzy vault system. We designed the scheme to share secret that suppose secret key is divided into n pieces and hided in a fuzzy vault using your favorite movies set called . If someone wants to recover hidden secret using same taste or interest of movies, he or she create similar size of favorite movies set called B then if and only if A and B overlap sufficiently, he or she can recover shared secret key k .

I. Introduction

Secret sharing is to distribute a secret among participants. Then the secret can be recovered if a sufficient number of shares are combined together. In generalized secret sharing scheme, a dealer who wants to share a secret has a secret key which is divided into n pieces. There are many schemes that introduced how to securely and efficiently share a secret. Among them, fuzzy vault scheme is more suitable for our scheme [2] [3]. According to [4], weighted feature vault can be created from human face images. By adding intermediate layer that includes genuine weight values and counterfeit weight values as a chaff, the scheme can reflect the layer of importance of feature sets. The heavier weight will give more information about point on the polynomial than with lighter weights.

In this paper we introduce simple match maker movie preference scheme based on Nyang and Lee's fuzzy face vault [4]. For our match maker, we newly adapt fuzzy weight vault model that represented the relation between weight value and the number of chaffs under sufficient security level.

II. Fuzzy face vault

In [4], fuzzy face vault consists of two layers: the intermediate layer and polynomial layer. Let's assume that there are a set of data and each data has a different weight. In the intermediate layer, each data converted into several points on the polynomial layer according to the weight value. During the conversion, each data belongs to distinct buckets, which have the certain number of chaffs. Those chaffs are also converted into counterfeit points on the polynomial layer.

For example, let $X = x, x_2, \cdots\}$ be a set of data with the weight $W = \{w_1, w_2, \cdots\}$. In

the intermediate layer, is converted into Y $Y, Y_2, \cdots\}$ where $Y_i = \{h(x_i \| 1),$ $, h(x_i \| 2), \cdots, h(x_i \| w_i)\}$ and $h()$ is a cryptographic hash function. Note that the hash values represent the x-axis values on the polynomial layer. The corresponding y-axis values are determined using the secret polynomial.

There are a lot of points, including the converted points from X and chaff points, in the polynomial layer. If user achieve a certain similarity of original data X, he or she can find the sufficient number of points on the secret polynomial and recover the secret from the polynomial.

The authors did not mention about detailed parameters such as the number of chaffs and degree of polynomial for constructing real application. The question arises when the data lays in a certain value range, to find out the matched data on the intermediate layer we should use another similarity check. It may decrease the overall performance of the application.

III. Match maker

As noticed in the fuzzy face vault, they used similarity matching which decreases availability to find out the points on the secret polynomial. In this section, we show how to build fuzzy movie vault scheme which is match maker using weight movie preference.

To construct match maker movie preference scheme, we also used layered structure of fuzzy vault: captured movie layer and intermediate layer, polynomial layer.

In the captured movie layer, the fuzzy movies and original movies set (favorite movies set) which is hides the secret k, are divided into bucket according to their weight values. To determine weight value of the movies we used following linear equation $f(x) = ax + b$ where b indicates basic weight and a is slope of weight. However, we can choose weight as a s-curve, indices and so on.

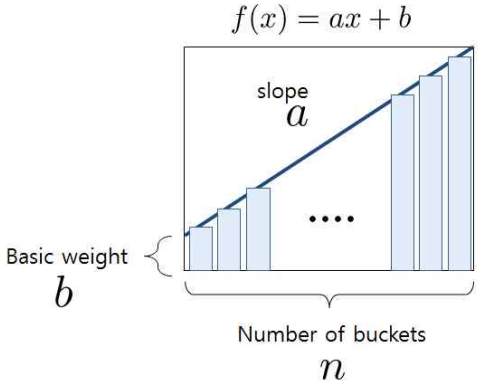

Fig.1. Fuzzy movie vault, weight value and bucket

The number of the bucket will be determined by number of the points on the polynomial which is required to recover the secret polynomial.

The intermediate layer includes genuine weight values of the favorite movies and counterfeit weight values as a chaffs. In the intermediate layer, the heaviest weight of the movie will give you more points on the polynomial than lightest weight values.

The polynomial layer includes Reed-Solomon codewords representing as secret as a corresponding polynomial with a chaff points which is not on the polynomial.

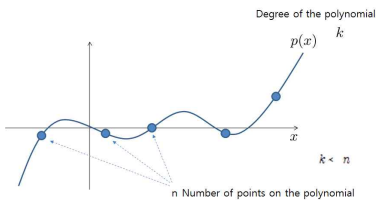

Fig.2. Secret Polynomial

For example, if secret value is $k = 123$, our secret polynomial will be defined $p(x) = x^2 + 2x + 3$ whose coefficients are the secret then codewords can be computed as

following $(x\ ), P(x_2), \cdots, P(x_n)]$[5]. This codewords are computed over x-coordinates corresponding to the movies in the favorite movies set. To hide the codewords, chaff points are added, such as , random noise in the form of random (x_i, y_i) pairs.

To unlock a vault, one must identify the codewords that was hidden under the favorite movie set A. Thus, create the unlock favorite movie set called B, and if ones movies set B overlap sufficiently with original movie set A as movie match maker scheme, you can recover the some pieces of codeword. According to intermediate layer some movies with a heavier weight will give multiple codewords and some will give less codeword depending on weight values. Any discrepancy between those sets will bring a certain amount of errors. However this errors (or noise) may be removed by Reed-Solomon decoding algorithm [7]. Consequently, the polynomial that encoded the secret key can be successfully reconstructed by similar favorite movies sets with that in the fuzzy movie vault construction.

I. Conclusion

In this paper, we presented method to share secret using match maker weighted features fuzzy movie vault scheme. In our scheme we used exact matching between the captured movies set with the original movie set (called favorite movies set). It means that we improved similarity matching of fuzzy face vault scheme into the exact matching scheme. Also fuzzy face vault is heavily depending on the weighted feature however our scheme slightly depends on the weight . Moreover, we improved our scheme security $\sum_{i=\tau}^{n} \frac{1}{(c1+1)f(i)+c2} \leq \frac{1}{2^k}$ where $c1 > 0$ is chaffs, $c2 > 0$ is basic chaffs, $f(x)$ is weight function, k is degree of the polynomial and n is the number of points on the polynomial. The weighted fuzzy movie fault scheme is using well-know techniques from error-correcting codes and cryptography.

The hardness to recover the secret sharing is based polynomial decoding Reed-Solomon codeword.

[Reference]

[1] Morgan Barbier. Re-encoding reformulation and application to welch-berlekamp algorithm. In IEEE, ISIT'14, 2014.

[2] Ari Juels and Madhu Sudan. A fuzzy vault scheme. IACR Cryptology ePrint Archive, 2002:93, 2002.

[3] Ari Juels and Martin Wattenberg. A fuzzy commitment scheme. In CCS '99, Proceedings of the 6th ACM Conference on Computer and Communications Security, Singapore, November 1-4, 1999., pages 28 - 36, 1999.

[4] DaeHun Nyang and KyungHee Lee. Fuzzy face vault: How to implement fuzzy vault with weighted features. In Universal Access in Human Computer Interaction. Coping with Diversity, 4th International Conference on Univer- sal Access in Human-Computer Interaction, UAHCI 2007, Held as Part of HCI International 2007, Beijing, China, July 22-27, 2007, Proceedings, Part I, pages 491 - 496, 2007.

[5] Irving S. Reed and Gustave Solomon. Polynomial codes over certain finite fields. Journal of the Society for Industrial and Applied Mathematics, 8:300 - 304, 1960.

[6] Adi Shamir. How to share a secret.

Commun. ACM, 22(11):612 - 613, 1979.

[7] L.R. Welch and E.R. Berlekamp. Error correction for algebraic block codes, December 30 1986. US Patent 4,633,470.