Special Issue - 2015
International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
NCACCT-2015 Conference Proceedings

Cryptobiometric Key Generation

Eeshita Roy Chowdhury
Student
Computer Science and Engineering
DSATM
Bangalore, India

Dr. C. Nandini
Professor
Computer Science and Engineering
DSATM
Bangalore, India

Aparna Ghosh
Student
Computer Science and Engineering
DSATM
Bangalore, India

***Abstract* - Crypto Biometrics system is recently emerging as an effective process to generate a cryptographic key. Conventional cryptographic key generation is using password which can be guessed or cracked. Further, the large size of strong key results in delay while encryption / decryption. Biometric field has however emerged in the recent days to reduce process delay and enhances the level of accuracy. Conventional techniques depend on biometric features like face, fingerprint, hand geometry, iris, signature, keystroke, voice and the like for the extraction of key. Instead of storing key we will generate the key dynamically with the help of biometrics. In this paper we propose a novel method for generating Cryptographic key by hashing the fingerprint minutiae and using different set of symmetric hash functions generated using combinations where the "n" is generated by the random function for different users, and in this proposed method we have increased the use of random number generator, because of this the decryption becomes harder and hence is safe, secured. We propose this method to be used with the existing AES algorithm with 128 bit key size.**

Keywords— Cryptographic key, hash function, minutiae , fingerprint

I. INTRODUCTION

In earlier cryptosystems, user authentication was based on possession of secret keys. The safe keeping of the keys was really difficult and so was to carry them around. To overcome this problem many Crypto biometric techniques have been proposed.

Existing techniques generates the key directly from the biometric data. If the biometric template is compromised, the user loses his biometric template permanently. There is a lack of privacy if different applications use the same biometric template of a user. Fingerprints are among the more reliable biometrics and there is a long history of their use in criminal cases.

We have used pre existing image processing techniques to enhance the image of the finger prints and get k coordi nates from it. Instead of generating a key directly from biometrics, they introduce a method of biometric locking: A predefined random key is locked with a biometric sample by forming a phase-phase product. This product can be unlocked by another genuine biometric sample. Biometric locking appears a promising idea because the biometric key can be randomly defined.

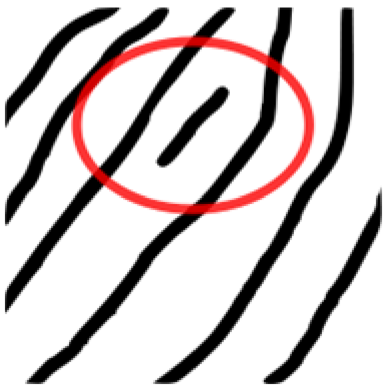
II. FINGERPRINTS BIOMETRICS

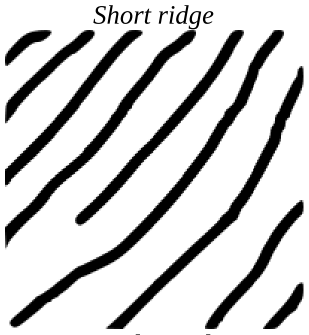
A fingerprint is made of a number of ridges and valleys on the surface of the finger. Ridges are the upper skin layer segments of the finger and valleys are the lower segments. The ridges form so-called minutiae points: ridge endings (where a ridge end) and ridge bifurcations (where a ridge splits in two).

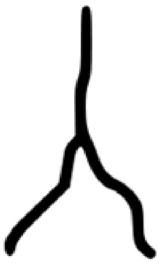
Many types of minutiae exist, including dots (very small ridges), islands (ridges slightly longer than dots, occupying a middle space between two temporarily divergent ridges), ponds or lakes (empty spaces between two temporarily divergent ridges), spurs (a notch protruding from a ridge), bridges (small ridges joining two longer adjacent ridges), and crossovers (two ridges which cross each other).

A. *Fingerprint Processing*

Fingerprint is preprocessed by Histogram Equalization and Filters are used to enhance the image. Binarization is applied on fingerprint image. Then Morphological operation is used to extract Region of Interest.


Short ridge


Ridge ending

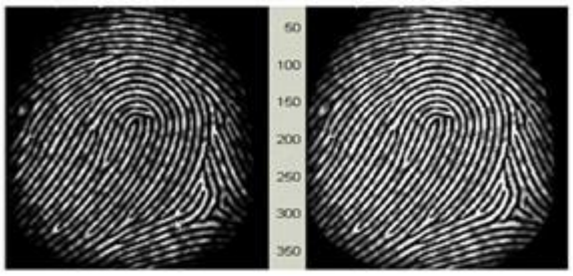


Bifurcation

Fingerprint loops

B. Histogram Equalization

Histogram equalization increases the contrast of images, especially when usable data of the image represented by close contrast values. Perceptional information of the image is increased through Histogram equalization. It permits pixel value to expand. The used Fingerprint image use bimodal type. Histogram equalization converts range from 0 to 255 which will enhance visualization effect. Sample fingerprint before and after histogram equalization is shown in figure.



Original image After histogram equalization

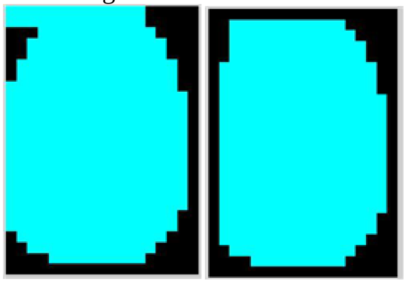
C. Binarization

Fingerprint Image Binarization is to transform the 8-bit Gray fingerprint image to a 1-bit image with 0-value for ridges and 1-value for furrows. After the operation, ridges in the fingerprint are highlighted with black color while furrows are white. A locally adaptive binarization method is performed to binarize the fingerprint image.

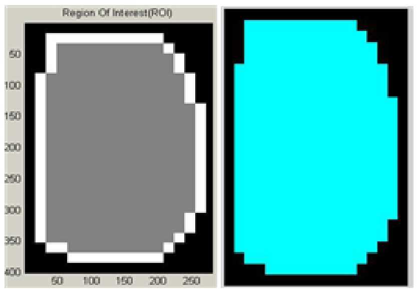
Original gray scale image after binarization

D. ROI Extraction By Morphological Operations

Two Morphological operations [12] called 'OPEN' and 'CLOSE' are adopted. The 'OPEN' operation can expand images and remove peaks introduced by background noise. The 'CLOSE' operation can shrink images and eliminate small cavities. The bound is the subtraction of the closed area from the opened area. Then the algorithm throws away those leftmost, rightmost, uppermost and bottommost blocks out of the bound so as to get the tightly bounded region just containing the bound and inner area.



Original Image Area After CLOSE operation



After OPEN operation ROI + Bound

E. Minutiae Points Extraction

Ridge Thinning is to eliminate the redundant pixels of ridges till the ridges are just one pixel wide uses an iterative, parallel thinning algorithm. In each scan of the full fingerprint image, the algorithm marks down redundant pixels in each small image window (3x3). And finally removes all those marked pixels after several scans.

After the fingerprint ridge thinning, marking minutiae points is relatively easy. For each 3x3 window, if the central pixel is 1 and has exactly 3 one-value neighbors, then the central pixel is a ridge branch. If the central pixel is 1 and has only 1 one-value neighbor, then the central pixel is a ridge ending.

Suppose both the uppermost pixel with value 1 and the rightmost pixel with value 1 have another neighbor outside the 3x3 window, so the two pixels will be marked as branches too. But actually only one branch is located in the small region. So a check routine requiring that none of the neighbors of a branch are branches is added.

III. ALGORITHM

1. Creating a function to calculate the factorial of a no.(this is to calculate $^{n}c_{r}$ values as $^{n}c_{r}= n!/((n-r)!*r!)$)
2. Creating a hash function for generating the key. We have used a sorted array of coordinates obtained from the fingerprints and its length as the input parameter.
3. We have used the following mathematical formula to generate 10 hash functions.

   $H_1= (^{n1}c_{r1}+ {}^{n1}c_{r2}\ldots\ldots\ldots + {}^{n1}c_{rk})$

   $H_2= (^{n2}c_{r1}+ {}^{n2}c_{r2}\ldots\ldots\ldots + {}^{n2}c_{rk})$

   .

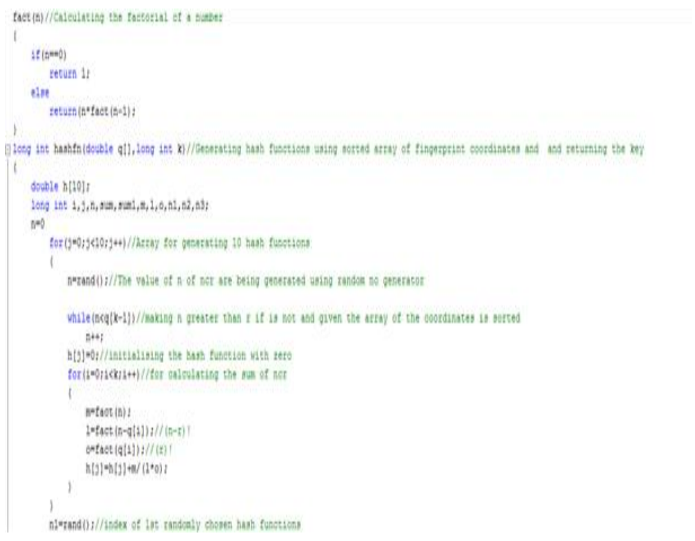
   .

   .

   $H_{10}= (^{n10}c_{r1}+ {}^{n10}c_{r2}\ldots\ldots\ldots + {}^{n10}c_{rk})$

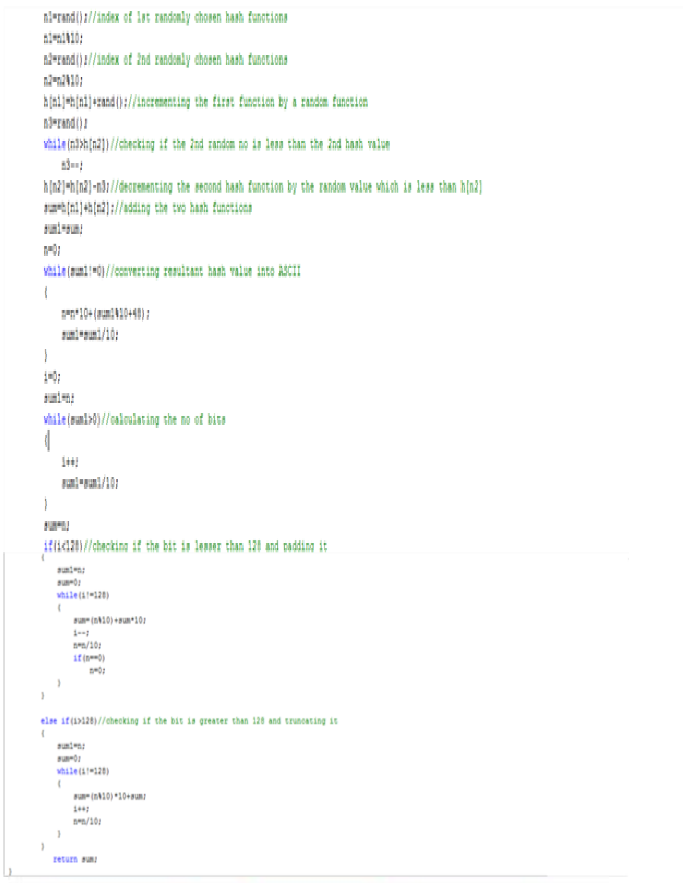
   Where n1, n2….n10 are generated using random number generator and n1, n2…n10>rk, where r1, r2…rk are the coordinates obtained from the fingerprint.

4. Choosing two indices for the hash function randomly.
5. Generating a random number by which the first function is incremented and generating another random function which is less than the second hash function and decrementing the hash function with it.
6. Then the two resultant hash values are added and converted into ASCII values.
7. Then it is checked whether the resultant hash value is greater than 128 bits. If so it is truncated.
8. If it is lesser than 128 bits then we are padding it with zeros.
9. If it is equal to 128 bits then the hash value is returned as the key.



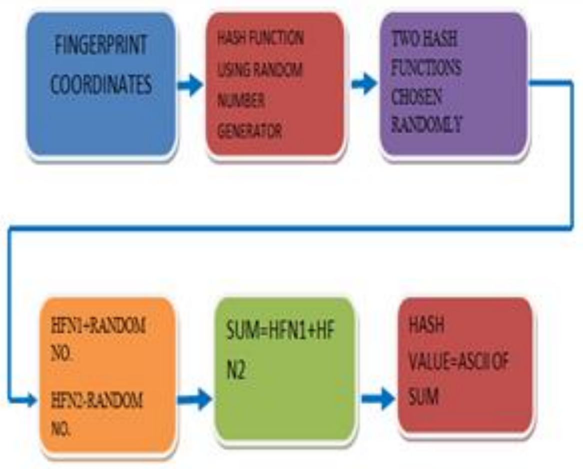
```
fact(n)//Calculating the factorial of a number
{
    if(n==0)
        return 1;
    else
        return(n*fact(n-1));
}
long int hashfn(double q[],long int k)//Generating hash functions using sorted array of fingerprint coordinates and  and returning the key
{
    double h[10];
    long int i,j,n,sum,sum1,m,l,o,n1,n2,n3;
    n=0
        for(j=0;j<10;j++)//Array for generating 10 hash functions
        {
            n=rand();//The value of n of ncr are being generated using random no generator

            while(n<q[k-1])//making n greater than r if is not and given the array of the coordinates is sorted
                n++;
            h[j]=0;//initialising the hash function with zero
            for(i=0;i<k;i++)//for calculating the sum of ncr
            {
                m=fact(n);
                l=fact(n-q[i]);//(n-r)!
                o=fact(q[i]);//(r)!
                h[j]=h[j]+m/(l*o);
            }
        }
        n1=rand();//index of 1st randomly chosen hash functions
```



```
n1=rand();//index of 1st randomly chosen hash functions
n1=n1%10;
n2=rand();//index of 2nd randomly chosen hash functions
n2=n2%10;
h[n1]=h[n1]+rand();//incrementing the first function by a random function
n3=rand();
while(n3>h[n2])//checking if the 2nd random no is less than the 2nd hash value
    n3--;
h[n2]=h[n2]-n3;//decrementing the second hash function by the random value which is less than h[n2]
sum=h[n1]+h[n2];//adding the two hash functions
sum1=sum;
n=0;
while(sum1!=0)//converting resultant hash value into ASCII
{
    n=n*10+(sum1%10+48);
    sum1=sum1/10;
}
i=0;
sum1=n;
while(sum1>0)//calculating the no of bits
{
    i++;
    sum1=sum1/10;
}
sum=n;
if(i<128)//checking if the bit is lesser than 128 and padding it
{
    sum1=n;
    sum=0;
    while(i!=128)
    {
        sum=(n%10)+sum*10;
        i--;
        n=n/10;
        if(n==0)
            n=0;
    }
}

else if(i>128)//checking if the bit is greater than 128 and truncating it
{
    sum1=n;
    sum=0;
    while(i!=128)
    {
        sum=(n%10)*10+sum;
        i++;
        n=n/10;
    }
}
   return sum;
}
```



IV. ADVANTAGES

- The main advantage of crypto biometry is that we do not have to worry about the safe keeping of the key or losing the key since it is generated from a fingerprint.
- Fingerprints are unique. So no other fingerprint can generate the same hash value.
- In this method we have extensively used the random number generator so that decrypting the key becomes very hard because even the programmers are not aware of the value that has been used to generate that hash value and it is different for different set of coordinates and impossible to revert.

V. CONCLUSION

We have described a novel Cryptographic key generation technique that uses the fingerprint features to generate the key. Our proposed method achieves the two important properties of Cryptographic keys, reproducibility and unguessable property. There by we can generate a wide range of cryptographic keys by changing the hash function by running the program another time as it uses random number generator which generates different hash value. If the hash function itself is compromised, new hash function can be issued, but the biometric signal itself is not lost forever.

Different applications can use different hash functions for the same biometric template, thus securing the biometric signal as well as preserving privacy. Because the biometric itself is providing as first level of authentication and this approach can be implemented in any applications where security is critical factor. Also it provides as two factor authentication which are fingerprint biometric and cryptographic key.

VI. ACKNOWLEDGEMENT

We would like to thank our HOD Dr.**C. Nandini and B. Shylaja**
Department of Computer Science and Engineering, Dayananda Sagar Academy of Technology and Management, Bangalore, India for giving us immense support and encouragement and for taking out time from their busy schedule to clear our queries and make this paper possible.

REFERENCES

[1.] C. Nandini and B. Shylaja." Efficient Cryptographic key Generation from Fingerprint using Symmetric Hash Functions." International Journal of Research and Reviews in Computer Science (IJRRCS)Vol. 2, No. 4, August 2011, ISSN: 2079-2557© Science Academy Publisher, United Kingdom.

[2.] Dhanraj, C. Nandini, and Mohd. Tajuddin "An Enhanced Approach for Secret Key Algorithm based on Data Encryption Standard" International Journal of Research and Reviews in Computer Science (IJRRCS)Vol. 2, No. 4, August 2011, ISSN: 2079-2557© Science Academy Publisher, United Kingdom www.sciacademypublisher.com

[3.] www.google.com

[4.] www.wikipedia.com