Generating a public key from voice biometric to enhance the security

May Thu Myint
University of Computer Studies, Mandalay
mthumyint@gmail.com

Abstract

In this paper, we motivate and summarize our work on repeatedly generating cryptographic keys from audio files that are user input. Then, the processing steps for the speaker's signal which is used to produce the public key. There are several biometric systems in existence that deal with cryptography, but the proposed voice biometric system introduces a novel method to generate cryptographic key. The goal of this work is to generate the public key using RSA and RC4 stream cipher upon its user speaking voice files and then this key is used to encrypt user's voice data and text files. The keys resists cryptanalysis even against an attacker who captures all system information related to generating or verifying the cryptographic key. Finally the study proposed the use of RSA, Diffie-Hellman and RC4 algorithms to identify/authenticate the voice of the speaker and therefore use the voice as the public key and to secure the communication between two users.

1. Introduction

Voice encryption systems are used to guarantee end-to-end security for speech in real time communication systems such as GSM, VoIP, Telephone, analogue Radio. Also, Security is needed for the purposes of e-banking, ATM, access to e- mail and computer accounts , access to personal information and biometrics application Security approaches can be broken down into two approaches: passive (authentication) and active (identification). Passive approaches are like a shield in that they protect against a clear and present danger such as a hacker attempting to access a computer system, while active approaches are more like prevention via a preemptive strike as in arresting terrorists before they plant a bomb. The use of PINs and passwords somehow improves the situation, but the fundamental problem with PINs is that they identify a card but not its user. While all of the traditional approaches have their strengths, they also have corresponding weakness. In this paper RSA, Diffie-Hellman (DH) and RC4 cryptosystems will be used to ensure the security of the communication channel. The keys here are divided into public key and private key. Public key is generated from the speaker voice and the corresponding private key will be considered as the DH private key. A shared secret will be calculated to generate the input key for the RC4. RC4 algorithm will generate a key-stream to complete the encryption and decryption process. This paper is organized as follows: In Section 2, we provide related work, Section 3 provides feature extraction method .Next in Section 4, we describe the impact of RSA and DH as public key cryptographic methods and RC4 as stream cipher and the proposed framework from capturing the speaker utterance, generating the cryptographic keys. Finally, in Section 5 we conclude remarks on this system.

2. Related Work

Voice is a combination of physiological and behavioral biometrics. The features of an individual's voice are based on the shape and size of the appendages (e.g., vocal tracts, mouth, nasal cavities, and lips) that are used in the synthesis of the sound. Based on the spoken text, there are two types of the biometric voice recognition system, that is, text dependent and text independent. A text dependent voice recognition system is based on the utterance of a fixed predetermined phrase, as we used in our proposed system. A text independent voice recognition system recognizes the speaker independent of what he/she speaks. Monrose, Reiter, Li and Wetzel were among the first: their system [10] is based on key-stroke dynamics. A short binary string is derived from the user's typing patterns and then combined with her password to form a hardened password. Each key- stroke feature is discretized as a single bit, which allows some error tolerance for feature variation.

A. Jagadeesan et al. [10] projected an efficient approach based on multimodal biometrics (Iris and Fingerprint) for generating a secure cryptographic key. At first, the minutiae points and texture properties are extracted from the fingerprint and iris images respectively and these features are fused at the feature level to obtain the multi-biometric template. Finally, the multi-biometric template is used for generating a 256-bit cryptographic key.

3. Voice Recognition

Voice recognition systems fall into two categories: text-dependent and text-independent**.** A text dependent voice recognition system is based on the utterance of a fixed predetermined phrase. A text independent voice recognition system is recognized the speaker independent of what he/she speaks, as we used in our proposed system**.** Voice Recognition system has desirable attributes of features; Practical: Occurs naturally and frequently in speech and easily measurable. Robust: Not change over time or be affected by speaker's health. Secure: Not be subject to mimicry.

3.1. End Point Detection

Endpoint detection algorithms generally use the combination of zero-crossing rate and energy OR the combination of zero-crossing rate and average magnitude of a given utterance. The first technique that applies to each speech recording process, speech waveforms have blank parts in the beginning and in the end. These parts are purely noise and they do not contain any information. The two main reasons for doing this is firstly, most of the speaker specific information or features reside in the voiced segment of the speech signal. Secondly, removal of the silent segment reduces unnecessary computation. Average energy of the signal is computed and segments of the speech signal with energy lower than the threshold set are removed. Changes to the threshold value might be required under different ambient/noise conditions [3]. ZCR refer to the rate that the amplitude of the sound wave changes sign.

3.2. Feature Extraction

Feature extraction converts digital speech signal into sets of numerical descriptors called feature vectors that contain key characteristics of the speaker.

3.3. Mel-Frequency Cepstral Coefficients

Mel-frequency Cepstral coefficient is one of the most prevalent and popular method used in the field of voice feature extraction. Mel is a unit of measure of perceived pitch or frequency of a tone. MFCC has two types of filter which is are spaced linearly at low frequency below 1000 Hz and logarithmic spacing above 1000 Hz. MFCCs are computed by taking the windowed frame of the speech signal, putting it through a Fast Fourier Transform (FFT) to obtain certain parameters. Mel-scale warping is to retrieve feature vectors that represents useful logarithmically compressed amplitude and simplified frequency information. The Mel-frequency cepstral represents the short-term power spectrum of a sound using a linear cosine transform of the log power spectrum of a Mel scale.

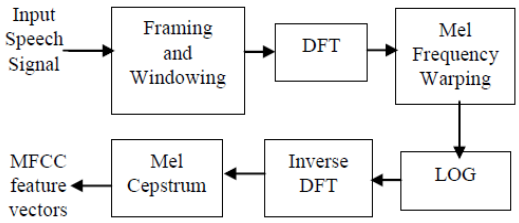

Figure 1: Block diagram of Mel Frequency Cepstral Coefficients

3.3.1. Mapping Frames to Features

The main target now is to define features of these frames that are exactly the same when the same user speaks the same utterance. From these features, an *m*-bit feature descriptor is then derived. The feature used to generate the feature descriptor *b* from the data vectors *V*(1) ... *V*(N) depends on the amplitude values of the data vectors. In this approach the i-th feature $\varphi i=0$ if the amplitude value is negative, $\varphi i=1$ otherwise. More precisely, let the mean and standard deviation of $\boldsymbol{x}_i$ over the last $\boldsymbol{h}$ successful logins (for some parameter h) be $\mathbf{U}_i$ and $\mathbf{R}_i$ respectively. Consider the partial feature descriptor $\boldsymbol{B}$ defined:

$$B(i) = \begin{cases} 0 & \text{If } U_i + KR_i < t \\ 1 & \text{If } U_i - KR_i > t \\ F & \text{Otherwise} \end{cases}$$

4. Cryptography

Public key cryptography is the science of using mathematics to encrypt and decrypt information. Cryptographic techniques are divided into two generic types: Symmetric key and Asymmetric key. Symmetric key is a conventional type of cryptography which is also known as secret key cryptography [2]. The same key is used for encryption and decryption process. Examples of the symmetric key cryptosystems are Data Encryption Standard (DES), Triple DES and Advanced Encryption Standard (AES). Asymmetric key algorithms which give an alternative way of securing data require a huge amount of time to do the computation for encryption and decryption.

4.1. RSA

RSA was created by Ron Rivest, Adi Shamir, and Leonard Adleman in 1977 [4]. It is the only asymmetric algorithm in widespread use that is used for private/public key generation and encryption. Two mathematical problems play the important rule for the RSA cryptosystem [6]: the problem of factoring very large numbers, and the RSA problem. The integer factorization problem is the problem of finding a nontrivial factor of a composite almost prime number. The RSA problem is simply the task of taking e-th roots modulo a composite *n*, trying to get the plaintext *m* such that $m^e = c \bmod n$, where the RSA public key is *e*, and *n*. An attacker needs to factor *n* into *p* and *q*, and computes *(p-1)(q-1)* which allows the determination of *d* from *e*.

4.2. Diffie-Hellman Key Exchange

The concept of DH key exchange is commonly known as DH. DH represents the last names of the inventors Whitfield Diffie and Martin Hellman. The method was introduced in 1976, and it was the first practical method for agreeing on a shared secret key based on a secure key exchange protocol over an unsecured communications channel. DH generates a secret number just for one transaction. This is called a

session key or a symmetric key. The security of the DH cryptosystem depends on the discrete logarithm problem. It assumes that computationally infeasible to calculate the shared secret key, $K = g^{X_A X_B} \bmod n$, given the two public values ($g^{X_A} \bmod n$) and ($g^{X_B} \bmod n$) where n is a sufficiently large prime.

4.3. RC4 Stream Cipher

RC4 is the most widely used stream cipher designed in 1987 by Ron Rivest for RSA Security [6]. It is a variable key size stream cipher with byte-oriented operations. The algorithm is based on the use of random permutations. RC4 is a very fast stream cipher and it is used in the SSL/TLS. The encryption is done by applying XOR operation on a byte of the plaintext with one byte of the key-stream. Decryption process is the same to the encryption process but the same key-stream byte is XOR ed with the cipher text instead.

Table 1. Average time needed to generate RSA keys in seconds

RSA key size	Time to prepare RSA keys on P 1.7 GHz	Time to prepare RSA keys on P.412MHz
64-bit	0.21875	0.515625
128-bit	0.40625	1.40625
256-bit	0.53125	2.359375
512-bit	0.78125	2.765625
1024-bit	0.328125	6.09375

Table 1 shows average timing results for generating RSA keys for variable key sizes. The average time is done using ten tests for each key size. From Table 3, it is obvious that using 64-bit long RSA keys is very fast. After RSA keys have been computed, DH key exchange comes next. DH function first computes the corresponding private key. Secondly, it computes the second user public key. Finally, exchange the key using RSA and computes secret shared key. Table 2 shows the timing result to generate the secret key.

Table 2. Average time needed to generate DH secret keys in seconds

DH Secret key size	Time to prepare DH key on P 1.7 GHz	Time to prepare DH key on P.412MHz
64-bit	0.015625	0.21875
128-bit	0.0625	0.3125
256-bit	0.09375	0.4375
512-bit	0.15625	0.828125
1024-bit	0.28125	1.53437

When the secret key is known, the key is used as RC4 key. Table 3 shows the time required to initialize RC4. Finally, the encryption process will start. Encryption tests were done and the timing results were taken for either doing XOR operation alone, or encoding and then doing XOR operation over bytes of data.

Table 3. Time needed to initialize RC4

Time in seconds on P 1.7 GHz	Time in seconds on P412 MHz
0.0006	0.0056

Table 4. Time needed to initialize RC4

Operation	Time in seconds on P 1.7 GHz	Time in seconds on P412 MHz
XOR(1-byte)	0.00001	0.00003
Encode & XOR(1-byte)	0.00007	0.00015
XOR (2-byte)	0.00002	0.00006
Encode & XOR (2-byte)	0.00015	0.0003

Table 4 shows the timing results for generating key stream and performing encryption or decryption process. Each byte of data needs one byte of key stream.

4.4. Generating Keys, Encryption and Decryption Flow

To generate the RSA private key the following steps are performed:

(i) Generate two large random and distinct primes p and q, each roughly the same size.

(ii) Compute $(n = pq)$ and the totient function $\phi(n) = (p - 1)(q - 1)$

(iii) Check the public key e with $\phi(n)$ such that $gcd(e, \phi) = 1$. If this condition is achieved then move on to step 4, else go back to step 1.

(iv) Use the extended Euclidean algorithm to compute the unique integer d, $1 < d < \phi(n)$, such that $ed \equiv 1 \pmod{\phi(n)}$.

(v) Share (n, e), and keep private key d.

Next, the RSA private key obtained above will be considered as the DH private key and another process will start to generate a secret key for both users as follows:

(i) The two global integers n and g are fixed and known by each user.

(ii) The user's DH private key here is the RSA private key which has been already computed. So Alice's private key is $X_A = D_A$ while Bob's is $X_B = D_B$.

(iii) From n, g and X, each part computes his own DH public key, $Y_A = g^{X_A} \bmod n$ for Alice and $Y_B = g^{X_B} \bmod n$ for Bob.

(iv) Both users exchange their DH public keys after encrypting them with RSA public keys.

(v) In order to get the secret key or session key, each user decrypt the DH public key. Here, Alice will

compute $K=Y_B^{X}{}_A$ mod n to get the secret key and Bob will compute $K=Y_A^{X}{}_B$ *mod n* to get the same secret key.

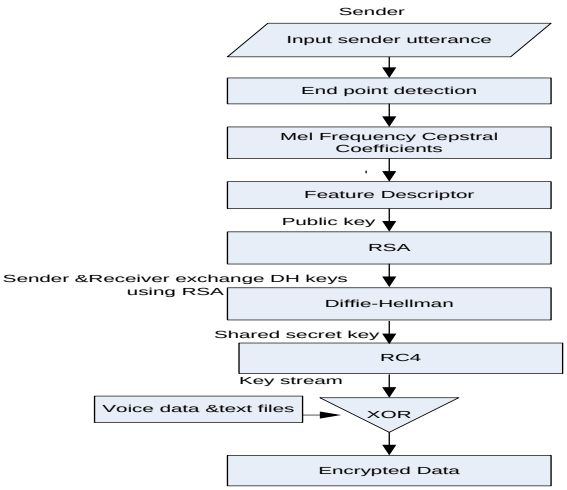

Figure 2: Proposed System Flow Chart from Sender

Now both users can compute the secret key by completing DH key exchange algorithm. The result is the same secret key for both sides; this secret key is used only for one communication session. Next, having the shared secret key in hand, the key will be used as the key for the RC4 stream cipher algorithm. RC4 stream cipher will start to produce a key stream; the key stream will be the same for both sides.

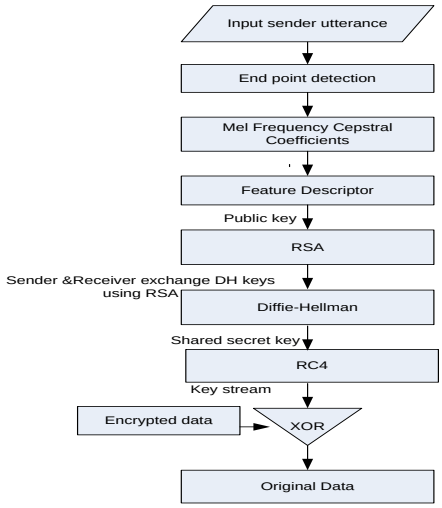

Figure 3: Proposed System Flow Chart from Receiver

This key stream will be used for doing two operations; encrypting the speaker voice and messages, and decrypting the received information from the other speaker. The process of generating these keys is next discussed in detail. Fig. 2 illustrates the encryption flow from sender.

Decryption is simply done by XORing the key stream with the encrypted data stream to get the original data stream as illustrated in Fig. 3. The audio stream can be applied to pre-processing steps before doing encryption. That is encode the audio stream before doing encryption and decodes the audio stream after decryption to get the original audio stream.

5. Conclusion

This paper has proposed a method of combining audio Feature Extraction and public key cryptography for encryption and decryption of voice data and text files. The cryptographic keys have been generated using recorded voice combination patterns which is stable throughout person's life real time application. For Feature Extraction, Mel-Frequency Cepstral Coefficients was selected due to the fast process, the quality of the results and it is applicable to any spoken word. RSA algorithm generates public keys which are used to secure the exchange process of DH public keys. It is secure to exchange DH public keys without encryption but exchanging them this way gives additional security for the proposed method. Moreover, the overall results of this system are: Key authentication is done implicitly and automatically using MFCC feature extraction and the generated key is secure against the man-in-the middle attack. The key is based on RSA, DH and RC4 cryptosystems. This key is further used as an RC4 key which is a fast algorithm to generate key stream and it is suitable for real-time applications. Future work will include an investigation by setting up the keys is required to fast the time enough for both PC and mobile phone usage.

References

[1] M.G. Sumithra, K. Thanuskodi and A. Helen Jenifer Archana, "A New Speaker Recognition System with Combined Feature Extraction Techniques, Journal of Computer Science, 459-465, 2011 ,ISSN 1549-3636 2011 .

[2] K. Sharma, H . P . Sinha and R .K . Aggarwal, " Comparative study of speech recognition system using various feature extraction techniques",2012.

[3] A. Keerio, B. Kumar Mitra, Philip Birch, Rupert Young, and Chris Chatwin. "On Preprocessing of Speech Signals". *International Journal of Signal Processing ; Vol.5 No.3 2009.*

[4] R. Singh, R. M. Stern, and B. Raj, "Signal and Feature Compensation Methods for Robust Speech Recognition", *Chapter in CRC Handbook on Noise Reduction in Speech Applications.*

[5] R. M. Stern, A. Acero, F. H. Liu, and Y. Ohshima, "Signal Processing for Robust Speech Recognition", *Chapter in Speech Recognition*

[6] G. Saha, S. Chakroborty, S . Senapati, " A new silence removal and end point detection algorithm for speech and speaker recognition applications",2011.

[7] F.L. Huang, "An Effective Approach for Chinese Speech Recognition on Small size of Vocabulary", an International Journal (SIPIJ) Vol.2, No.2, June 2011.

[8] Koji Kitayama, Masataka Goto, Katunobu Itou and Tetsunori Kobayashi, "Speech Starter: Noise-Robust Endpoint Detection by Using Filled Pauses", Eurospeech 2003, Geneva, pp. 1237-1240.

[9] Jain, A. K.; Ross, A. and Prabhakar, S., "An introduction to biometric recognition", *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 14, No. 1, 2004, pp. 4-20.

[10] Monrose, F.; Reiter, M. K.; Li, Q. and Wetzel, S., "Cryptographic Key Generation from Voice", in *Proceedings*

of the IEEE Symposium on Security and Privacy - S&P'01, Oakland, CA, USA, May 14-16, 2001, pp. 202-213.