Volume 4, No. 1, January 2013

Journal of Global Research in Computer Science

ISSN-2229-371X

RESEARCH PAPER

Available Online at www.jgrcs.info

AN APPROACH FOR VERIFYING DIGITAL WATERMARKED-FINGERPRINT USING FUZZY COMMITMENT SCHEME

Bhupendra Kumar*[1] and Deo Brat Ojha[2]
[1]Research Scholar Mewar University, Chittorgarh, Rajsthan, India
bhupe2002@gmail.com
[2]Professor, Department of Mathematics, Mewar University, Chittorgarh, Rajsthan, India

Abstract: The proposed approach meets the requirements of current era of research. However, with the changing scenario in the world, it is important that utilize approaches with full effectiveness for complete utilization. Digital watermarked data are utilized as a second modality, or source, in order to assist the system performance under acoustically degraded condition. In order to extract maximum information from the watermarked image and distinguish error, the proposed approach fulfills all the requirements.

Keywords: **C**ryptography, Fuzzy commitment scheme, Digital Watermarking, Honesty.

INTRODUCTION

Commitment schemes are an essentials ingredient of many cryptographic protocols. Commitments schemes are the process in which the interest of the party involve in a process are safeguarded and the process itself is made as fair as possible. Parties which perform according to the prescribed rules and aimed to achieve the protocol objective are called 'honest' [1]. Fuzzy commitment scheme was first Juels and Martin, fuzziness was introduced later for generating cryptography key [2, 3, 4].

The impression of commitment scheme is indispensable for the construction of modern cryptographic protocols. Since security violation is usual phenomena hence the need of commitment scheme in cryptographic protocol cannot be ruled out. Now a days, dishonesty between communicating parties emerges as salient problem. The vital role of 'fuzzy decision making' under fuzzy commitment scheme makes assure about appropriateness of communication between two parties, even after this assurance dishonesty may play their role.

In this paper, we elaborate possible cases that are the treacherous role of communicating parties. The organization of the paper is as follows: Section 2 gives some definitions and notation that will be used in the sequel, Crisp commitment scheme, Hamming distance, error correction function, measurement of nearness, fuzzy membership function, Commitment scheme, Fuzzy Commitment scheme and fuzzy decision making. In section 3, we analyze four possible cases for without trusted party and three possible cases with trusted party.

PRELIMINARIES

a. ***Egyptian times:*** Early use of biometrics
b. ***14th century:*** Chinese merchants used hand palm prints and footprints on paper to distinguish young children
c. ***19th century:*** Biometric methods used to solve
d. ***End of 19th century:*** Fingerprints became popular for forensic use The system had the problem, that no easy way of sorting and identifying fingerprints was known
e. ***1900:*** Classification system distinguishing fingerprint classes proposed Variations of this system are the basis of many fingerprint identification systems nowadays

Crisp Commitment Schemes:

In a commitment scheme, one party A (sender) aim to entrust a concealed message 'm' to the second party B (receiver), intuitively a commitment scheme may be seen as the digital equivalent of a sealed envelope. If A wants to commit a message 'm', he just puts it into the sealed envelope, so that whenever A wants to reveal the message to B, A opens the envelope. First of all the digital envelope should hide the message from: B should be able to learn 'm' from the commitment. Second, the digital envelope should be bind, which means that A cannot change his mind about 'm', and by checking the opening of the commitment one can verify that the obtained value is actually the one A had in mind originally[5].

Definition:

Let $C\{0,1\}^n$ be a code set which consists of a set of code words c_i of length n. The distance metric between any two code words c_i and c_j in C is defined by

$$dist(c_i, c_j) = \sum_{r=1}^{n} \left| c_{ir} - c_{jr} \right| \qquad c_i, c_j \in C$$

This is known as Hamming distance [6].

Definition:

An error correction function f for a code C is defined as $f(c_i) = \{c_j / dist(c_i, c_j)$ is the minimum, over $C - \{c_i\}\}$. Here, $c_j = f(c_i)$ is called the nearest neighbor of c_i [3].

Definition:

The measurement of nearness between two code words c and c' is defined by $\text{nearness}(c,c') = dist(c,c')/n$, it is obvious that $0 \le \text{nearness }(\text{c},\text{c}') \le 1$[3].

Definition:

The fuzzy membership function for a codeword c' to be equal to a given c is defined as[3]

$$
\begin{array}{ll}
FUZZ(c') = 0 & \text{if nearness}(\text{ç c}') = \text{z} \le \text{z}_0 < 1 \\
\qquad = \text{z} & \text{otherwise}
\end{array}
$$

Definition :

Commitment scheme[1] is a tuple***{P, E,M }*** Where ***M*** ={0,1}n is a message space, ***P*** is a set of individuals , generally with three elements A as the committing party, B as the party to which Commitment is made and TC as the trusted party , ***E*** = { (t_i, a_i) } are called the events occurring at times t_i, i = 1,2,3 , as per algorithms a_i , i = 1,2,3. The scheme always culminates in either acceptance or rejection by A and B.

The environment is setup initially, according to the algorithm *Setupalg*(a_1) and published to the parties A and B at time t_1. During the Commit phase,

A uses algorithm *Commitalg*(a_2), which encapsulates a message m∈M, along with secret string S∈R$\{0,1\}^k$ into a string C. The opening key (secret key) could be formed using both m and S. A sends the result C to B (at time t_2).

In the Open phase, A sends the procedure for revealing the hidden Commitment at time t_3, and B uses this. *Openalg*(a_3): B constructs C' using *Commitalg*, message m and opening key, and checks weather the result is same as the commitment C
Decision making:
If (C = C')
Then A is bound to act as in 'm'
Else he is free to not act as 'm'

Definition :

Fuzzy Commitment scheme[2] is a tuple ***{P, E, M, f }*** Where ***M***$\in\{0,1\}^k$ is a message space which consider as a code, ***P*** is a set of individuals, generally with three elements A as the committing party, B as the party to which Commitment is made and TC as the trusted party , ***f*** is error correction function (def. 2.3) and ***E*** = { (t_i, a_i) } are called the events occurring at times t_i , i = 1,2,3 , as per algorithms a_i , i = 1,2,3. The scheme always culminates in either acceptance or rejection by A and B.

In the setup phase, the environment is setup initially and public commitment key K generated, according to the algorithm *Setupalg*(a_1) and published to the parties A and B at time t_1.

During the Commit phase, Alice commits to a message m∈M according to the algorithm
Commitalg(a_2) into string C.

In the Open phase, A sends the procedure for revealing the hidden Commitment at time t_3 and B use this.*Openalg*(a_3): B constructs C' using *Commitalg*, message t(m) and opening key, and checks weather the result is same as the received commitment t(C), where t is the transmission function.
Fuzzy decision making:
If (nearest (t(C),*f*(C'))≤ z_0)
Then A is bound to act as in 'm'
Else he is free to not act as 'm'

Digital Watermark:

The type of watermark influences the effectiveness of the watermark in various applications [7,8,9,10,11,12]. For example, both perceptible and imperceptible watermarks can detertheft, but they do so in very different ways. Perceptible watermarks are especially useful for conveying an immediate claim of ownership. The main advantage of perceptible watermarks, in principle at least, is that they virtually eliminate the commercial value of a document or media object without significantly lessening the document's utility for legitimate, authorized purposes. That is to say, the watermark in Figure makes it clear that the document belongs to someone, but it does this without preventing the appreciation of the artifact. A familiar example of a visible watermark is in the video domain where CNN and other television networks place their translucent logo at the bottom right of the screen image.

Imperceptible watermarks on the other hand are only effective as a theft deterrent if the potential thief has a reason to believe that watermarks might be present, and further that they may be used to prosecute unauthorized possessions. If we assume that the majority of potential thieves are at best A computationally challenged. However, though weak in terms of discouraging theft, imperceptible watermarks really shine as a means of identifying the source, version or serial number, author, creator, owner, distributor or authorized consumer of a document or image. For this purpose, the objective is to permanently and unalterably mark the image so that the credit or assignment is beyond dispute. In the event of illicit usage, the watermark would facilitate the claim of ownership, the receipt of copyright revenues, or the success of prosecution.

In some cases, both watermarking schemes are equally effective. For example, both may be used to determine the location, and trace image migration, of documents over the networks. Several computing companies are developing the software to deploy watermark agents in network patrols to detect infringement. It should be remembered that watermarking software can assign a unique watermark to each document or object for each authorized user or consumer.

Least Significant Bit Substitution:

The most straight-forward method of watermark embedding, would be to embed the watermark into the least-significant-bits of the cover object .Given the extraordinarily high channel capacity of using the entire cover for transmission in this method, a smaller object may be embedded multiple times. Even if most of these are lost due to attacks, a single surviving watermark would be considered a success.

LSB substitution however despite its simplicity brings a host of drawbacks. Although it may survive transformations such as cropping, any addition of noise or loss compression is likely to defeat the watermark. An even better attack would be to simply set the LSB bits of each pixel to one…fully defeating the watermark with negligible impact on the cover object. Furthermore, once the algorithm is discovered, the embedded watermark could be easily modified by an intermediate party. The algorithm however would still be vulnerable to replacing the LSB's with a constant. Even in locations that were not used for watermarking bits, the impact of the substitution on the cover image would be negligible. LSB modification proves to be a simple and fairly powerful tool for stenography, however lacks the basic robustness that watermarking applications require.

OUR APPROACH

Proposed Watermarking Scheme:

Basic Blocks and Value Assignments:
Let us consider a Java class file consisting of many methods. Each method can be decomposed into a control-flow diagram, consisting of only four possible types of basic blocks:

a. Simple sequential blocks, consisting of at most 8 instructions.
b. Extended blocks, consisting of a sequence of one or more sub-blocks, which can be of any type.
c. Extended if-then-else blocks, consisting of two extended sub-blocks, corresponding to the THEN block and the ELSE block.
d. Extended iteration blocks, consisting of one extended sub-block, which undergoes iteration.

Watermark Insertion Algorithm:

Our algorithm is divided into the following two phases:

a. The software producer Alice generates a watermark W which needs to be inserted into the Java class file P.W can be expressed in the form of either a number or a sequence of bits.
b. For each method Mi in P, do the following:
(a). Extract a control-flow diagram from Mi.
(b). Assign a value Vi based on the scheme described in the previous section.
(c). Generate a control-flow structure S' consisting of a sequence of basic blocks of dummy code and dummy variables whose value Vw is the same as that of the watermark.
(d). Insert S' at appropriate places into the method Mi, thus changing the value of Mi from Vi to Vi+W. The following points should be taken care of while inserting S' into each method Miss' contains dummy code and dummy variables. It is mandatory that the dummy code use the dummy variables and generate values which can be merged with that of the original code, e.g. if the original code is using some constant, S'could be used to calculate the constant and feed it to the original code. Other important points to note during the watermark insertion process are: For redundancy purposes which will be explained later, if N methods need to be protected, it is best if 2N+1 method are watermarked. The watermark W which is inserted should ideally be of the form of a digital fingerprint or author authentication mark, utilizing public-key cryptography. The original values Vi are stored safely for use in the watermark extraction phase. These values should only be known to the software producer and to nobody else.

Error Correction:

Receiver check that $dist(t(c)c^{,}) > 0$, he will realize that there is an error occurs during the transmission. Receiver applies the error correction function f to $c^{,} : f(c)$.
Then receiver will compute nearness
$(t(c), f(c^{,})) = dist(t(c) f(c^{,})) / n$

$$\begin{aligned} FUZZ(c') &= 0 \qquad \text{if nearness}(ç\,c') = z \le z_0 < 1 \\ &= z \qquad \text{otherwise} \end{aligned}$$

Watermark Extraction and Inspection:

Extraction. The watermark extraction algorithm proceeds as follows:

a. The original values Vi of each method Mi are accepted from the user.
b. The values Vi' are extracted from each method in the Java class file.
c. For each method, the difference Vi' - Vi is calculated.
d. For any 2N+1 watermarked methods, if more than N+1 methods generate an equal difference W', then the forensic software deems the watermark of the entire class file to be W'.
e. Any method whose difference Vi' - Vi is different from W' by a factor of4, say, is assumed to have been tampered with - the forensic software can generate various combinations of possible transforms which could have created the difference4:Forexample, if the difference 4= -8, the tampering agent could have:
(a). Deleted a sequential basic block of size 8, or
(b). Deleted two variables, or even
(c). Deleted an IF-THEN-ELSE and inserted a block of size 8 (possible on a software cracking exercise).

CONCLUSION

The main feature of this approach is digital watermarking with fuzzy scheme, here fuzzy correction obtain any information about the positions in which the error occurs. Thus the information rate is increasing and information leakage rate decreasing by using this approach.

REFERENCE

[1]. M. Blum, "coin flipping by telephone", "Advances in Cryptology-A report on CRYPTO'81, pp.11-15, 1981.

[2]. A.Jules and M. Wattenberg. "A fuzzy commitment scheme" in proceedings of the sixth ACM Conference on computer & communication security, pages 28-36, November 1999.

[3]. A.A.Al-saggaf , H.S.Acharya,"A Fuzzy Commitment Scheme" IEEE International Conference on Advances in Computer Vision and Information Technology,28-30,November,2007 – India.

[4]. Xavier boyen "Reusable cryptography fuzzy extractors" in proceedings of the eleventh ACM Conference on computer & communication security, pages82-91, ACM Press 2004.

[5]. Alawi A. Al-Saggaf and Acharya H. S. “A Generalized Framework for Crisp Commitment Schemes “eprint.iacr.org/2009/202.

[6]. V.Pless, “Introduction to theory of Error Correcting Codes”, Wiley, New York 1982.

[7]. M. Holliman, W. Macy, and M. Yeung, “Robust frame-dependent video watermarking,” in Security and Watermarking of Multimedia Contents II, ser.Proceedings of SPIE, vol. 3971, January 2000, pp. 186–197.

[8]. D. Curran, N. Hurley, M. Cinneide, Securing Java through Software Watermarking, PPPJ 2003, Kilkenny, Ireland, 2003.

[9]. W. Zhu, C. Thomborson, F-Y Wang, A Survey of Software Watermarking, LNCS vol3495, Springer-Verlag, pp. 454 - 458, April 2005.

[10]. A.K. Jain, L. Hong, S. Pankanti, and R. Bolle, “An identity authentication system using fingerprints”, Proc. IEEE, vol. 85, no.9, Sept. 1997, pp. 1356-1388.

[11]. M. Arnold, M. Schmucker and S.D. Wolthusen, Techniques and Applications of Digital Watermarking and Content Protection, Artech House, London, 2003.

[12]. Chuhong Fei, D. Kundur, and R.H. Kwong, “Analysis and design of watermarking algorithms for improved resistance to compression,” IEEE Transactions on Image Processing, Volume: 13, Issue: 2, Feb. 2004, pp. 126 – 144.