An Analysis Of Fraudulence In Fuzzy Commitment Scheme With Trusted Party

D.B.Ojha* ojhdb@yahoo.co.in
Deparment of Mathematics, R.K.G.I.T. , Ghaziabad (INDIA),
Ajay Sharma ajaypulast@rediffmail.com
Department of Information Technology, R.K.G.I.T., Ghaziabad (INDIA)
Ramveer Singh ramveersingh_rana@yahoo.co.in
Department of Information Technology, R.K.G.I.T., Ghaziabad (INDIA)
Shree Garg garg.shree@rediffmail.com
Department of Mathematics, S.I.E.T. , Saharanpur (INDIA)
Awakash Mishra awakashmishra@gmail.com
Deparment of M.C.A., R.K.G.E.C., Ghaziabad (INDIA)
Abhishek Dwivedi dwivediabhi@gmail.com
Deparment of M.C.A., R.K.G.E.C., Ghaziabad (INDIA)

Abstract

This paper attempt has been made to elaborate the possible cases of dishonesty between the two communicating parties under fuzzy commitment scheme. However there could be many instances where the transmission involves complete security, even if it contains errors arising purely because of the factors over which either sender or receiver have complete control. The concept itself is illustrated with the help of simple situations.

Keywords: Cryptography, Error correcting code, Fuzzy logic and commitment scheme, Error correction, Honesty.

1. INTRODUCTION:

Commitment schemes are an essentials ingredient of many cryptographic protocols. Commitments schemes are the process in which the interest of the party involves 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 firstly introduced by 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 here, three possible cases in commitment scheme with trusted party.

2. PRELIMINARIES:

2.1. 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 facilitate to open the envelope. First of all the digital envelope should hide the message from B and it 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].

2.2 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].

2.3 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].

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

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

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

2.6 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'

2.7 Definition : ***Fuzzy Commitment scheme[2]*** is a tuple ***{P, E, M, f }*** Where ***M***□$\{0,1\}^k$ 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'

3. ANALYSS OF A FUZZY COMMITMENT SCHEME:

This section presents an analysis of possible attacks against a fuzzy commitment scheme.

Let our analysis mainly consider a tuple [7],

{P,E,M,K,g(w,m),C,S,V(v,w),f,α_i}. …… (1)

Where 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 algorithms occurring at times t_i, i=1,2,3 , as per algorithms a_i, i=1,2,3 , M ⊆ $\{0,1\}^k$ is a message space which consider as a code, K is the public commitment key according to the algorithm setupalg (a_1) and publish to the parties A and B at time t_1, g_w is an encoding function with key w, C is the image set under g is a code set, which satisfies the closure property under K operation , S is a element of set C, V is the set of verifier's tags for key w with value v, f is error correction function (def. 2.3) , α_i are possible attacks.

3.1 With trusted party:

Now equation (1) will become a tuple

{P,E,M,K,g(w,m),C,V(v,w),f,α_i}. …… (1)

Where 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 algorithms occurring at times t_i, i=1,2,3 , as per algorithms a_i, i=1,2,3 , M ⊆ $\{0,1\}^k$ is a message space which consider as a code, K is the public commitment key according to the algorithm setupalg (a_1) and publish to the parties A and B at time t_1, g_w is an encoding function with key w, C is the image set under g is a code set, which satisfies the closure property under K operation , V is the set of verifier's tags for key w with value v, f is error correction function (Def. 2.3) , α_i are possible attacks where i = 1, 2, 3

CASE α_1: Dishonesty of A for g with w

During the commit phase at time t_2:

Let g_w: M → C, ∀ m ∈ M, $Vv_{:w}$→ {0, 1}, ∀ w.

In this case we represent an attack where the committer ignore his key, here the trusted party TC gives a key w to A for hide the commitment and a verifier tag v to B which B can verify the key that A will reveal later.

In this attack, A commit a value 'm', compute g_w (m) and send this value to B. Now to open the commitment A sends w' to B and since every g_w is injective, knowing w' B can compute inverse $g_{w'}^{-1}(m) = m'$. To verify that w'=w and therefore m'=m, B computes his verifying function V (v, w').
Now if V (v, w') =1 than A can cheat to B successfully and B accept the commitment else B reject accordingly.

CASE α_2: Dishonesty of A for g regards v

During the commit phase at time t_2:

Let g_w: M → C, ∀ m ∈ M, $Vv_{: w}$→ {0, 1}, ∀ w .

In this case, A attack like this, he try to compute the set V_v of all the tags that B have. He than pick the tag v_0 □ V_v that maximizes Pr [V =v | w=w_0]. Let α_2= Pr [V =v | w=w_0]. By an averaging argument $\alpha_2 \geq 1/|V_v|$. Now A picks two value m=m' and compute g_w(m). But by concealing property, there is another key w' such that $g_{w'}$ (m') which is equal to g_w (m) and V (v, w') =1 which allow A to cheat B successfully.

CASE α_3: Denial of Service

Parties which act accordingly to the prescribe rule and aimed to achieve the protocol objective are called 'honest'. When at least one honest party is involved, the protocol succeeding despite the objective not having being achieved is a infringe or contravene (discussed earlier) of security.

The protocol is expected to fail is some of the parties act dishonestly – thus it is never in the interest of the dishonest party to perform an action that is guaranteed to lead to the protocol failing.

Here we disregarding the kind of 'Denial of service' attack where dishonest party start up protocol runs but intentionally never complete them.

Conclusion:

Perfidious behavior of one or both communicating parties still in existence, even after having strong cryptographic protocol, which maligns the soul of commitment scheme and shows the failure of it's objective.

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]. Alexandre Pinto, Andr´e Souto, Armando Matos, Luis Antunes "Galois Field Commitment Scheme" eprint.iacr.org, November 2006.