A generalized Framework for Crisp Commitment Schemes

Alawi A. Al-saggaf
Faculty of Computer Science and Engineering
Al-Ahgaff University – Hadhramout-Yemen
(Ph.D. research at SICSR-SIU)
Alwiduh@yahoo.com
Acharya H. S.
Symbiosis Institute of Computer Studies and Research
Symbiosis International University – Pune-India
haridas.acharya@symbiosiscomputers.com

Abstract

Crisp Commitment schemes are very useful building blocks in the design of high-level cryptographic protocols. They are used as a mean of flipping fair coins between two players and others. In this paper an attempt has been made to give a generalized framework for Crisp Commitment schemes is called an Ordinary Crisp Commitment Scheme (OCCS). The Hiding and Binding properties of OCCS are well defined. We also review some the existing of different Crisp Commitment schemes and we show how it is follow our presenting framework.

Keywords: Cryptography, Commitment Schemes, Discrete Mathematics

1. Introduction:

The notion of Crisp Commitment scheme is at the heart of most the constructions of modern Cryptography protocols. Protocols are essentially a set of rules associated with a process or a scheme defining the process. Commitment schemes are the processes in which the interests of the parties involved in a process are safeguarded and the process itself is made as fair as possible, used as a sub-protocols in such applications as zero knowledge proofs[20], secure multiparty computations[21], sealed-bid auctions and e-voting.

In the Commitment scheme, one party, whom we denote the sender namely Alice, aim to entrust a concealed message m to the second party namely Bob. Intuitively a commitment scheme can be seen as the digital equivalent of a sealed envelope [22]. If Alice wants to commit to some message m she just puts it into the sealed envelope, so that whenever Alice wants to reveal the message to Bob, she opens the envelope. Clearly, such a mechanism can be useful only if it meets some basic requirements. First of all the digital envelope should hide the message from Bob. (This is often referred in the literature as the hiding property). Second, the digital envelope should be binding, meaning with this that Alice cannot change her mind about m, and by checking the opening of the commitment one can verify that the obtained value is actually the one Alice had in mind originally (this is often referred to as the binding property).

Many crisp commitment schemes of OCCS type are in use over a considerable period of time [1,3,4,5,6,7,19]. The organization of this paper is: In the next section we give preliminaries. Section 3 the presenting generalized framework OCCS. In Section 4 we discuss some of the existing Commitment schemes and we show how it is follow our presenting OCCS. Section 5 conclusion.

2. Preliminaries:

Following mathematical and statistical concepts form the basis of our discussions.

Discrete logarithm.

Definition 1: Let Zp* be multiplicative group modulo p, let α be a generator of Zp* and let $\beta \in$Zp*. Let Zp-1 ={0,1,2,….,p-2} be additive group modulo p. Then the discrete logarithm is a function f : Zp* → Zp-1 and denoted by $\mathrm{dlog}_\alpha(\beta)$ (mod p). Which assigned to a unique integer $x\in$Zp-1 such that $\beta = \alpha^x$ (mod p).

Definition 2: Discrete Logarithm assumption: Given a prime p, let Zp* be the set of all positive integer numbers less than p and co-prime with p. Let α be a generator Zp* and an element $\beta\in$Zp*. Find an integer x, $0 \le x \le p-2$ such that $\beta = \alpha^x$ mod p.

Lemma 1: Let G be a finite cyclic group and g be a primitive root of G. for all q divisors of $|G|$, let Gq be the subgroup of G generated by b$|G|$/q . Then the groups Gq are all subgroups of G. in particular, every subgroup of G is cyclic, and for each divisor q of G there is a unique subgroup of G of order q, namely Gq.

Definition 3: Hash Function

A hash function is a function h: X → Y which has a minimum, the following two properties:

a. Compression – h maps an input x of arbitrary finite bit-length, to an output h(x) of fixed length.
b. Easy of computations: given h and an input x, h(x) easy to compute.

The following problems become naturally associated with any hash function:

1. Pre-image problem: Given h: X → Y and $y\in Y$, find $x\in X$ such that h(x)=y.
2. Second Pre-image problem: Given h: X → Y and $x\in X$, find $x'\in X$ such that $x\neq x'$ and h(x')=h(x).
3. Collision problem: Given h: X → Y , find x, $x'\in X$ such that $x\neq x'$ and h(x')=h(x)

Complexities of solving above problems, given a hash function, gives strength to any security policy that uses hash functions.

Definition 4: Universal hash function:
Let S and T be two sets and let U be a family of functions from S to T is called a Universal family of hash functions if for any two distinct elements s_1, s_2 in S and for any two elements t1,t2 in T we have Prob[u(s1)=t1 and u(s2)=t2]=1/|T|2 for u∈U.

Definition 5: Pseudo- Random Generator Function:
Let m(n) be some function such that m(n)>n. G:$\{0,1\}^n \rightarrow \{0,1\}^{m(n)}$ is a pseudo-random generator (PRG) if for all polynomial *p* and all probabilistic polynomial time machines D that attempt to distinguish between out of the generator and truly random sequences except for many n's:

$$|\text{Prob}[D(y)=1]-\text{Prob}[D(G(s))=1]| < 1/p(n)$$

Where the probabilities taken over y $\{0,1\}^{m(n)}$ and s $\{0,1\}^n$ chosen uniformly at random.

Definition 6: Statistical Difference between two probability distributions:
Let D1 and D2 be two discrete probability distributions defined over the same set X. the statistical difference between D_1 and D_2 is denoted by $\| D_1 - D_2 \|$ and is given by

$$\| D_1 - D_2 \| = \sum_{x\in X} |\text{Prob } D_1(x) - \text{Prob } D_2(x)| \quad ...... (1)$$

Definition 7: Pre-image set of y under h: Let h: X → Y be a hash function. Then the set of pre-images of y∈Y in X is defined by

$$\Omega y = \{x\in X \mid h(x)=y \text{ and } y\in Y\}$$

Theorem 1:

The set of pre-images $\{ \Omega y\}_{y\in Y}$, naturally define a partition on X.

Proof: follows from the fact that pre-images of two distinct elements are disjoint sets, and union of all pre-images is the domain set.

Associated with every hash function we can always define a uniform distribution function.

Definition 8: Uniform distribution on X: Let h: X → Y be a hash function.

Now define: Uy(x) = Prob[x∈X : h(x)=y]

It can be seen that above distribution is well defined. We get a family of distributions for each of y in Y. Hence we can consider two elements y1 and y2 to be close under hashing if the statistical difference between the corresponding uniform distributions is small.

3. An Ordinary Crisp Commitment Scheme: (OCCS)

An Ordinary Commitment scheme is a tuple{ M, X, Y,K, F, P, $E(e_i,t_i)$} where:
M: Message space, such that M=$\{0,1\}^n$.
X: A set of the witnesses, such that X =$\{0,1\}^k$.
Y: A set of the Commitments, such that Y =$\{0,1\}^{\ell}$.
K: is the set of Indices k encoded by unary (1^k) we called it the security parameter of public commitment key $F_k \in$**F**.

F : A family of public commitment keys (PCK), where $F_k \in$**F** and F_k:M × X → Y.

P: A set of individuals, generally with three elements A as a committing party, B as the party to which a commitment made and TC as the trusted party.

$E(e_i,t_i)$: The events occurring at times t_i as per algorithms e_i for i=1,2,3.

- The environment is setup initially, according to the algorithm *Setupalg – e_1.* TC select k▯ K which sufficient large and then generate a public Commitment key (PCK) $F_k \in F$ and publish it to the parties A and B at time t_1.
- During the commit phase at time t_2, A run the *Commitalg – e_2* , she encapsulate m∈M along with witness chosen at random x∈X into a Commitment $F_k(m,x)=y \in Y$, and sends y to B.

- In the Open phase at time t_3, A sends to B the necessary information m' and x' for revealing the Commitment, B uses this and run the *Openalg*-e3: by reconstruct $F_k(m',x')=y'$ and checks weather the result is same as the Commitment y.
  Decision making: If(y'=y)
  Then B is bound to act as in m'=m
  Else he is free to not act as m'=m

As shown in figure 1.

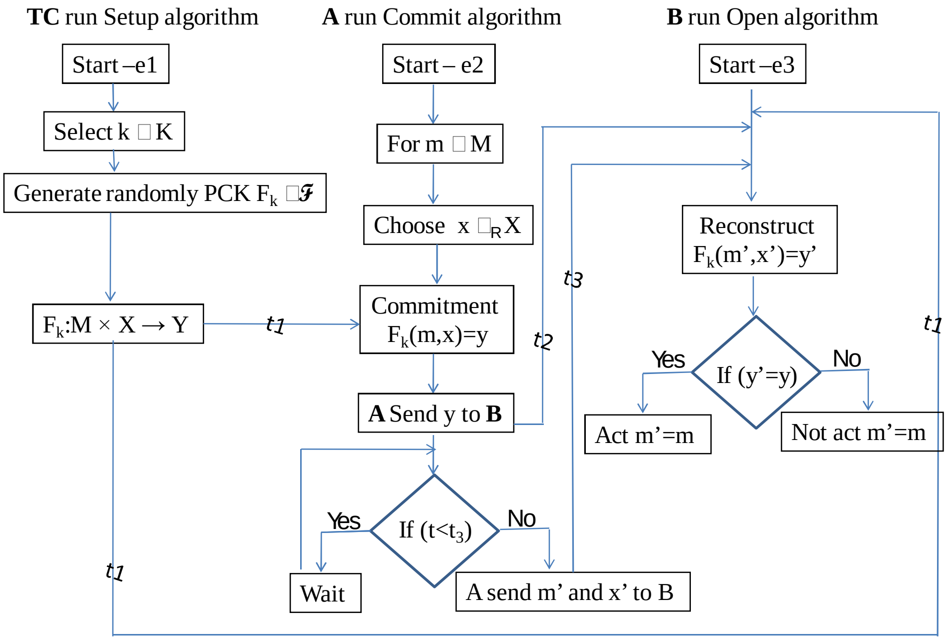

Figure 1: An Ordinary Crisp Commitment scheme

Definition 3.1 Hiding- Commitment:

A Commitment y in an Ordinary Crisp Commitment scheme is called λ-hiding (λ is a function of k). If for any $m\in M=\{0,1\}^n$, $x\in_R X=\{0,1\}^k$ and given $F_k\in \mathbf{F}$, Let U_{Fk} and U'_{Fk} are two different distributions over a set Ω_y obtained after running commit algorithm. Then

For all $(m_1, x_1), (m_2, x_2)\in\Omega y$ $\| U_{Fk}(m_1, x_1) - U'_{Fk}(m_1, x_1)\| = \lambda$ where $0\leq \lambda \leq 1$

i.e. the receiver cannot distinguish between what the sender committed to m_1 or m_2 only in λ. (note that $\in_R$ is stands for chosen randomly)

Definition 3.2 Binding – Commitment:

A commitment y in an Ordinary Crisp Commitment scheme is called μ-binding (μ is a function of k). If given $m\in M$, $x\in_R X$ and $y\in Y$ such that $F_k(m,x)=y$. Then the probability of obtain $m'\in M$ and $x'\in_R X$ such that $(m,x)\neq(m',x')$ and $F_k(m,x)=F_k(m',x')=y$ is given by:

$$Prob[(m',x')|(m,x): F_k(m',x')=F_k(m,x)]=\mu \text{ where } 0\leq \mu \leq 1$$

4. Different Commitment Schemes in Practice:

There are many Commitment Schemes presented in the last decade which based in different hard mathematical function. Schemes will be differing depending on the choice of algorithm F_k, table 1 [5] summarized the underlying schemes.

4-1 Pederson Commitment Scheme:

Pederson Commitment scheme [3] is non-interactive Commitment scheme which is based on discrete logarithm problem. Let

i. q: be k-bit large prime number.
ii. p: be a prime number such that $p\equiv1 \pmod q$.
iii. Z_q : is an additive group modulo q .

iv. Z_p^* : is a multiplicative group modulo p.

v. G_q : is a unique subgroup of order q of Z_p^* (Lemma 1)

vi. let $M= Z_q \subseteq\{0,1\}^k$, $X= Z_q \subseteq\{0,1\}^k$ and $Y= G_q\subseteq\{0,1\}^k$.

Setup Phase:

Input: Security parameter k encoding by unary 1^k.

- Select randomly k-bit prime number q.
- Select randomly a prime number p such that $p\equiv 1 \pmod q$).
- Calculate a generator g of the group Gq according to the algorithm:

```
1- randomly choose a from $1\le a\le p-1$.
2- If $g=a^{(p-1)/q} \neq 1 \pmod p$)
3- Then return g.
4- Else go to step 1.
```

- Calculate an element h in Gq according to the algorithm:

```
1- randomly choose b from $1\le b\le q-1$.
2- If $h=g^b\neq 1 \pmod p$)
3- Then return h.
4- Else go to step 1.
```

Output: The Public Commitment Key $F_k:Z_q\times Z_q \rightarrow G_q$ defined as $F_k(m,x)=g^m h^x$ (modp)

TC distribute the public Commitment key to the parties Alice and Bob at time t1.

Commit phase:

Input: Alice select her message $m\in M\subseteq\{0,1\}^k$ to be commit to.

1. She choose randomly a witness $x\in X\subseteq\{0,1\}^k$.
2. She compute the commitment $y=F_k(m,x)= g^m h^x$ (mod p).

Output: The concealed Commitment y.

Alice sends y to Bob at time t2.

Open phase:

Input: Alice reveal the message $m'\in M\subseteq\{0,1\}^k$ and the witness $x'\in X\subseteq\{0,1\}^k$ to Bob.

- Bob recomputed the commitment $y'=F_k(m',x')= g^m h^x$ (mod p).

```
If (y'=y)
    Then Bob is bound to act as in m'=m
Else Bob free to not act as m'=m.
```

Output: Bob is bound to act as in m'=m or not.

4-2 Halevi - Micali Commitment Scheme: Halevi and Micali Commitment scheme [5] is non-interactive Commitment scheme which is based on Collision Free hashing Function (CFHF). Let

i. $h:\{0,1\}^* \rightarrow\{0,1\}^k$: be a collision free hash function chosen from a family H.

ii. $u:\{0,1\}^{O(k)} \rightarrow\{0,1\}^k$: be a universal hash function chosen from a family U.

iii. $M=\{0,1\}^n$, $X=\{0,1\}^{O(k)}$ and $Y=\{0,1\}^{O(k).}$

Setup phase:

Input: Security parameter k encoding by unary 1^k.

- Choose CRHF h:$\{0,1\}^* \rightarrow \{0,1\}^k$ from a family of CRHF H.

Output: The Public Commitment Key F_k: M × X →Y defined as Fk(m,x)=(h(x),u).

Where u:$\{0,1\}^{O(k)} \rightarrow \{0,1\}^k$: be a universal hash function chosen from a family U such the u(x)=h(m) by the party Alice.

Commit phase:

Input: Alice select her message m∈M=$\{0,1\}^n$ which to be commit to.

1. She compute h(m)=s, where s ∈$\{0,1\}^k$.
2. She choose randomly a witness x∈X=$\{0,1\}^{O\ (k)}$
3. She choose a universal hash function u randomly from the family U.
4. She compute u(x)

```
        If (u(x)=s).
              Compute h(x)=c.
         Else go to step 2.
```

Output: The concealed Commitment (c, u).

Open phase:

Input: Alice reveal the message m'∈M=$\{0,1\}^n$ and the witness x'∈X=$\{0,1\}^{O\ (k)}$ to Bob.

- Compute h(x')

```
      If  ( h(x')=y )
          Calculate u(x') and h(m')
          If  (u(x')=h(m'))
              Then Bob is bound to act as in m'=m
           Else Bob is free to not act as in m'=m
      Else Bob is free to not act as in m'=m
```

Output: Bob is bound to act as in m'=m or not.

4-3dsNaor Commitment Scheme: Naor Commitment scheme [4] is an interactive Commitment scheme which is based on Pseudo-Random Generator.

Let

i. G:$\{0,1\}^k \rightarrow \{0,1\}^{3k}$: Pseudo-Random Generator.

ii. E:$\{0,1\}^n \rightarrow \{0,1\}^\ell$: An encoding function, where ℓ=3/2 k.

iii. R=(r_1,r_2, …….,r_{3k}): be a vector chosen from $\{0,1\}^{3k}$, such that *dist*(**0**,R)=ℓ, where *dist* is hamming distance and **0** is a 3k-bit zero vector (this vector is chosen by B and sends to A in commit phase).

M=$\{0,1\}^n$, X=$\{0,1\}^k$ and Y=$\{0,1\}^{3k}$.

Setup phase:

Input: Security parameter k encoding by unary 1^k.

- She choose Pseudo-Random Generator G:$\{0,1\}^k \rightarrow \{0,1\}^{3k}$.
- She choose an encoding function E:$\{0,1\}^n \rightarrow \{0,1\}^\ell$, where ℓ=3/2 k.
- Let R=(r_1,r_2, …….,r_{3k}): be a vector chosen from $\{0,1\}^{3k}$, (which chosen by Bob and sends to Alice before committing to her message).

Output: The Public Commitment Key F_k: M × X →Y defined as:

$$F_k(m,x)=\begin{cases} B_i(x) & \text{if } r_i=0 \\ B_i(x)\text{Xor}E\ (m) & \text{if } r_i=1 \end{cases} \qquad 1\leq i\leq 3k$$

TC sends PCK to the parties Alice and Bob at time t1.

Commit phase:

Input: Alice select her message $m \in M=\{0,1\}^n$ which to be commit to.

- Bob select a vector $R=(r_1,r_2, \ldots\ldots,r_{3k})$ from $\{0,1\}^{3k}$ s.t. $dist(\mathbf{0},R)=\ell$ and sends to Alice.
- Alice choose randomly the witness $x \in X=\{0,1\}^k$.
- Alice compute $G(x)=(B_1(x), B_2(x),\ldots\ldots, B_{3k}(x))$
- Alice encode her message using E.
- Alice compute the commitment

$$y=F_k(m,x)= \begin{array}{ll} B_i(x) & \text{if } r_i=0 \\ B_i(x)\text{Xor}E\ (m) & \text{if } r_i=1 \end{array}$$

Output: The concealed Commitment $y=(B_i(x), B_i(x)\text{Xor}E\ (m))$. Alice send y to Bob at time t2

Open phase:

Input: Alice reveal the message $m' \in M=\{0,1\}^n$ and the witness $x' \in X=\{0,1\}^k$ to Bob

- Bob compute $G(x')=(B_1(x'), B_2(x'),\ldots\ldots, B_{3k}(x'))$

```
While ($r_i$=0)
    If ($B_i(x')= B_i(x)$)
        Bob compute $E(m')= B_i(x')\text{Xor}(B_i(x)\text{Xor}E\ (m))$
        Bob compute m' by decoding E(m').
        Then Bob is bound to act as in m'=m
    Else Bob is free to not act as in m'=m
```

Output: Bob is bound to act as in m'=m or not.

Table 1: Comparison between existences an elementary Commitment Schemes

Name of the scheme	Original year of publishing	Complexity of assumptic	Local computation
Pederson[3]	CRYPTO 1991	Discrete Logarithm Problem(DLP)	Modular Multiplication
Naro[4]	CRYPTO 1989	Pseudo-Random Generator(PRG)	Pseudo-R andom Generator Error-Correcting Codes
Halevi-Micali[5]	CRYPTO 1996	Collision Resistance Has Function (CRHF)	CRHF Universal Hash Function

Name of the scheme	#rounds for Commitment	Length of commitment string	Length of security parameter =
Pederson[3]	1-Round	O(max(k,n))	1024-bits
Naro[4]	2-Round	O(max(k,n))	64-bits
Halevi-Micali[5]	1-Round	O(k)	128-bits

4- Conclusion:

We present a general framework of commitment scheme called an ordinary crisp commitment scheme OCCS. Some existing commitment schemes had been discuss and we show it is follows our framework.

References:-

[1] Manuel Blum, Coin flipping by telephone. Advances in Cryptology: A Report on CRYPTO ’81, pp. 11–15, 1981, http://www.cs.cmu.edu/~mblum/research/pdf/coin/

[2] Ronald Rivest, Adi Shamir, and Leonard Adleman, A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. Communications of the ACM 21,2, pp. 120–126, 1978, http://theory.lcs.mit.edu/~rivest/rsapaper.pdf.

[3] Torben Pryds Pedersen, Non-Interactive and Information- Theoretic Secure Verifiable Secret Sharing. Advances in Cryptology - CRYPTO ’91, 11th Annual International Cryptology Conference, pp. 129–140,1991, http://www.cs.cornell.edu/courses/cs754/2001fa/129.PDF.

[4] M.Naor: *Bit Commitment using pseudo-randomness*, Proceedings of Crypto 89, Springer Verlag LNCS series. http://citeseer.ist.psu.edu/naor91bit.html.

[5] Shai Halevi, Silvio Micali, Practical and Provably-Secure Commitment Schemes from Collision-Free Hashing. Advances in Cryptology - CRYPTO ’96, 16th Annual International Cryptology Conference, pp. 201–215, 1996, http://coblitz.codeen.org:3125/citeseer.ist.psu.edu/cache/papers/cs/778/ftp:zSzzSztheory.lcs.mit.eduzSzpubzSzpeoplezSzshaihzSzcomitmnt2.pdf/halevi96practical.pdf.

[6] Shai Halevi, *Efficient Commitment Schemes with Bounded sender and Unbounded Receiver*, Proceedings of Crypto '95 LNCS. Vol.963 Springer-Verlag 1995 pages 84-96. http://citeseer.ist.psu.edu/halevi96efficient.html

[7] Eiichiro Fujisaki, Tatsuaki Okamoto, Statistical Zero Knowledge Protocols to Prove Modular Polynomial Relations. Advances in Cryptology - CRYPTO ’97, 17th Annual International Cryptology Conference, pp. 16–30, 1997, http://dsns.csie.nctu.edu.tw/research/crypto/HTML/PDF/C97/16.PDF

[8] Yuval Ishai, Eyal Kushilevitz, Rafail Ostrovsky, Sufficient Conditions for Collision-Resistant Hashing. Theory of Cryptography, Second Theory of Cryptography Conference, TCC 2005, pp. 445–456, 2005, http://www.cs.ucla.edu/~rafail/PUBLIC/66.pdf.

[9] Ivan Damg°ard, Jesper Buus Nielsen, Commitment Schemes and Zero-Knowledge Protocols, 2006, http://www.daimi.au.dk/~ivan/ComZK06.pdf.

[10] Hans Delfs and Helmut Knebl: Introduction to Cryptography Principle and Applications. 2002 Springer- Verlag Berlin Heidelberg.

[11] William Stallings, 2001: *Network Security Essentials applications and standards*, Wesley Longman (Singapore) Ptd. Ltd. Indian branch.

[12] Oded Goldreich: Foundation of Cryptography, Vol. 1 ,June 2001. Fragment of the book available online.http://www.wisdom.weizmann.ac.il/~oded/foc-book.html.

[13] Alfred Menezes, Paul Van Oorschot and Scott Vanstone: Handbook of Applied Cryptography, CRC press 1996.

[14] Marc Fischlin, *Trapdoor commitment schemes and their applications*, Ph.D. thesis, 2001.

[15] Marc Fischlin and Roger Fischlin, *Efficient Non-Malleable Commitment Schemes* Advances in Cryptology | Crypto 2000, Lecture Notes in Computer Science, Vol.1880, pp.414{432, Springer-Verlag, IACR.

[16] D. Boneh and M. Naor, "*Timed Commitment*"

[17] G. Di Crescenzo, J.Katz, A Smith and R. Ostrovsky: *Efficient and Non-interactive and Non-Malleable Commitment*, EUROCRYPT 2001 pages 40-59 2001.

[18] Liina Kamm: Research Seminar in Cryptography ' Commitment Schemes', University of Tartu. November 15, 2006.

[19] Jordi C. Roca and Josep D. Ferrer, Anon-Repudiable Bitstring Commitment Scheme Based on a Public Key Cryptosystem, *Information Technology: Coding and Computing, 2004. Proceedings. ITCC 2004. International Conference on April 2005*, pages 778-780 vol. 2 IEEE 2004.

[20] O. Goldreich, S. Micali and A. Wigderson: Proofs that Yeild Nothing but their Validity or All Languages in NP have Zero-Knowledge Proof System. J. ACM 691-729 (1991).

[21] O. Goldreich, S. Micali and A. Wigderson: How to Play any Mental Game or a Completeness Theorem for Protocols with Honest Majority. STOC'87.

[22] Alawi A. Al-saggaf, Acharya H. S: Mathematics Of Bit-Commitment Schemes, Bulletin of the Marathwada Mathematical Society, Vol. 8, No. 1, June 2007, pages 08 – 15.