Wireless Pers Commun (2012) 63:459–467
DOI 10.1007/s11277-010-0143-0

A New Verifiable Multi-secret Sharing Scheme Based on Bilinear Maps

Ziba Eslami · Saideh Kabiri Rad

Published online: 10 September 2010
© Springer Science+Business Media, LLC. 2010

Abstract In a (t, n)-threshold multi-secret sharing scheme, several secrets are shared among n participants in such a way that any t (or more) of them can reconstruct the secrets while a group of $(t - 1)$ can not obtain any information. Therefore, when such schemes are used to distribute sensitive information over a network, fault tolerance property is achieved since even if $n - t$ of the nodes go out of function, the remaining t nodes suffice to recover the information. In 2009, Wang et al. proposed a verifiable (t, n)-threshold multi-secret sharing scheme (WTS) based on elliptic curves in which the secrets can change periodically [Wireless Pers. Commun., Springer-Verlage, doi:10.1007/s11277-009-9875-0]. In this paper, we propose a verifiable (t, n)-threshold multi-secret sharing scheme based on bilinear maps. Our scheme does not require a secure channel and participants can verify the shares pooled in the reconstruction phase. Our proposed scheme is multi-use such that in order to change the secrets, it is sufficient to renew some public information. Furthermore, the proposed scheme is flexible to the threshold value. Therefore, our proposed scheme has all the merits of (WTS), however, we achieve two major improvements. First when the secrets are to be changed, we require to publish fewer public values. This reduction can be very important in certain applications such as steganographic use of secret sharing schemes. The second is that (WTS) is designed with the assumption that the number of secrets (m) is equal to the threshold t so that the case $m > t$ is handled by repeating the scheme $\lceil \frac{m}{t} \rceil$ times. However, in designing the scheme we do not assume any restrictions on the number of secrets.

Keywords Threshold scheme · Verifiable scheme · Multi-secret sharing · Bilinear pairing · Multi-use scheme

Z. Eslami (✉) · S. Kabiri Rad
Department of Computer Science, Shahid Beheshti University, G.C., Tehran, Iran
e-mail: z_eslami@sbu.ac.ir

S. Kabiri Rad
e-mail: saideh.kabiri@gmail.com

Z. Eslami
School of Mathematics, Institute for Research in Fundamental Sciences (IPM),
P.O. Box 19395-5746, Tehran, Iran

1 Introduction

Secret sharing schemes are cryptographic procedures to share a secret $\mathscr{K}$ among a set of participants $\mathscr{P}$ such that only authorized subsets of $\mathscr{P}$ can recover the secret. Such schemes were independently introduced by Shamir [1] and Blakley [2] to safeguard cryptographic keys from loss. In recent times, secret sharing schemes have found applications in diverse areas such as access control systems, e-voting schemes and digital cash protocols, to name a few.

A very important example of a secret sharing scheme is the (t, n)-threshold scheme which allows a mutually trusted party (called the dealer) to distribute the shares among n participants in such a way that any t of them can recover the original secret, but any group knowing only $t - 1$ or fewer shares can not. Shamir's scheme, which is based on polynomial interpolation, and Blakley's scheme, based on the intersection of affine hyperplanes, are examples of such schemes. However, one can distinguish the following drawbacks in these schemes [3]:

1. Only one secret can be shared during one secret sharing process.
2. Once the secret has been reconstructed, it is required that the dealer redistributes a fresh share over a secure channel to every participant.
3. A malicious participant may provide a fake share to other participants so that s/he which may become the only one who gets to reconstruct the true secret.

In order to overcome the first problem, multi-secret sharing schemes (MSS) were proposed by He and Dawson [4]. In these schemes, several secrets can be shared while holding one share by each participant. Jackson et al. [5], classified multi-secret sharing scheme into"one-time-use" and "multi-use" types, where in the latter type the second problem is resolved, i.e., after reconstructing the secrets, the distributed shares must not be updated. Problem 3 is tackled by adding the concept of verifiability. In a verifiable secret sharing scheme the validity of the shares can be verified, hence neither the dealer nor participants are able to cheat. The first realization of a verifiable secret sharing scheme was proposed by Chor et al. [6]. Verifiability plays an important role in protocols such as secure multi-party computations [7,8]. Recently, elliptic curves and bilinear maps have been used in providing verifiability [9,10]. In this paper, we propose a verifiable (t, n)-threshold multi-secret sharing scheme based on elliptic curves and bilinear maps. Our proposal is of multi-use type and is flexible to the threshold value. We compare our scheme to Wang et al.'s scheme [10] and show that our scheme needs fewer number of public values. In Sect. 5, we describe why this is very important for a secret sharing scheme. Moreover, contrary to Wang et al.'s scheme, the number of secrets can be greater than the threshold.

The rest of this paper is organized as follows: in Sect. 2 elliptic curves and bilinear maps are briefly introduced and a review of the scheme of Wang et al. is presented. The proposed scheme is presented in Sect. 3 and its security analysis is given in Sect. 4. Section 5 provides comparison with (WTS) and conclusions appear in Sect. 6.

2 Related Work

In this section, we first briefly introduce elliptic curves (2.1) and bilinear maps (2.2) and then proceed to review the scheme of Wang et al. (2.3).

2.1 Elliptic Curves

An elliptic curve defined over $GF(q)$ is given by the equation: $E : y^2 = x^3 + ax + b$, where $a, b \in GF(q)$ and $4a^3 + 27b^2 \neq 0$. The points of E (plus an infinite point O) together with a special operator "+", form an finite abelian group.

The elliptic curve discrete logarithm problem (ECDLP): Given two points P and Q on $E(GF(q))$, find the integer k such that $kP = Q$ if such a k exists. There is no polynomial time algorithm (on lgq) for solving ECDLP [11].

2.2 Bilinear Maps

Let G be a cyclic additive group generated by P whose order is a prime q. We define the following problems for all $a, b, c \in Z_q^*$:

1. Computational Diffie-Hellman Problem (CDHP): Given (P, aP, bP), compute abP.
2. Decision Diffie-Hellman Problem (DDHP): Given (P, aP, bP, cP), decide whether $c = ab$ in Z_q^*.

We call G a Gap Diffie-Hellman (GDH) group, if DDHP can be solved in polynomial time, but no probabilistic algorithm can solve CDHP with non-negligible advantage within polynomial time [12].

No candidate for GDH group is known except some supersingular elliptic curve or hyperelliptic curve over finite field, which are equipped with a bilinear map such as the Weil pairing [13], the Tate pairing [14] or the self bilinear pairing in [15]. Let G_1 be a cyclic additive group generated by P whose order is a prime α, and let G_2 be a cyclic multiplicative group of the same order α. We assume that the DDHP problems in G_1 is easy, the DDHP problem in G_2 is hard, and both the CDHP problem in G_1 and the discrete logarithm problem (DLP) in G_2 are hard. A bilinear map is a map $e : G_1 \times G_1 \to G_2$ with the following properties.

1. Bilinear: $e(aP, bQ) = e(P, Q)^{ab}$ for all $P, Q \in G_1$ and $a, b \in Z_\alpha^*$.
2. Non-degenerate: There exists $P, Q \in G_1$, such that $e(P, Q) \neq 1_{G_2}$.
3. Computable: There is an efficient algorithm to compute $e(P, Q) \in G_2$, for all P, $Q \in G_1$.

2.3 Review Of Wang et al.'s Scheme

There are four phases: (1) system setup phase, (2) secrets distribution phase, (3) secrets reconstruction phase and finally (4) secrets update phase.

2.3.1 System Setup Phase

a. Let $E(F_q)$ be a supersingular elliptic curve (q is a prime number). For some prime number α, there is an additive subgroup G_1 with order α and an extended field, including a multiplicative group G_2 of non-zero elements of order α such that the assumptions of Sect. 2.2 holds. Let $e : G_1 \times G_1 \to G_2$ be the bilinear mapping. The dealer chooses a generator P of G_1, chooses a function $h : G_1 \to Z_\alpha^*$, and publishes $< \alpha, G_1, G_2, e, P, h >$ on the bulletin.
b. Each participant U_i, $(i = 1, \ldots, n)$ downloads the public information $< \alpha, G_1, G_2, e, P, h >$, randomly selects a private number $s_i \in Z_\alpha^*$, computes $P_i = s_i P$ and then sends P_i to the dealer.

c. The dealer ensures that $P_i \neq P_j$ where $i \neq j$ in order to keep different participants from using the same secret key and publishes $P_i(i = 1, \ldots, n)$ on the bulletin.

2.3.2 Secrets Distribution Phase

It is assumed that there are as many secrets as the threshold value t, i.e., $K_1, \ldots, K_t$ are the secrets. The dealer performs the following steps:

a. Randomly selects $s \in Z_\alpha^*$, compute sP and publish it. Then s/he uses $P_i(i{=}0, \ldots, n-1)$ and the hash function h to compute the matrix $M_{n\times t}$ of rank t:

$$M = \begin{bmatrix} h(ss_0P) & h^2(ss_0P) & \ldots & h^t(ss_0P) \\ h(ss_1P) & h^2(ss_1P) & \ldots & h^t(ss_1P) \\ \vdots & \vdots & \ddots & \vdots \\ h(ss_{n-1}P) & h^2(ss_{n-1}P) & \ldots & h^t(ss_{n-1}P) \end{bmatrix} \tag{1}$$

b. Puts all t secrets which he wants shared in a $t \times 1$ column vector $X = (K_1, K_2, \ldots, K_t)^T$, where T represents the transpose of a matrix, and computes an $(n \times 1)$ column vector V:

$$V = M \times X = \begin{bmatrix} h(ss_0P) & \ldots & h^t(ss_0P) \\ h(ss_1P) & \ldots & h^t(ss_1P) \\ \vdots & \ddots & \vdots \\ h(ss_{n-1}P) & \ldots & h^t(ss_{n-1}P) \end{bmatrix} \begin{bmatrix} K_1 \\ K_2 \\ \vdots \\ K_t \end{bmatrix} = \begin{bmatrix} I_0 \\ I_1 \\ \vdots \\ I_{n-1} \end{bmatrix} \tag{2}$$

c. Publishes $I_i(i = 0, \ldots, n-1)$ on the bulletin.

2.3.3 Secrets Reconstruction Phase

Let t participants $U_i(i = 0, \ldots, t-1)$ pool their shares $ss_iP(i = 0, \ldots, t-1)$. Each U_i generates the i-th row of the matrix M and sends it to the combiner who is one of the participants. Now, the combiner solves t equations to recover the t unknown secrets.

2.3.4 Secrets Update Phase

The dealer chooses a new threshold t', new secrets $K_1', \ldots, K_t'$, and the new seed s' and proceeds as secret distribution phase and finally publishes $I_i'(i = 0, \ldots, n-1)$ and $s'P$.

3 The Proposed Scheme

In this section, we propose a new verifiable dynamic (t, n)-threshold multi-secret sharing scheme using elliptic curves and bilinear pairing. The scheme consists of four phases: (1) initialization phase, (2) secrets distribution phase, (3) secrets reconstruction and verification phase, and (4) secrets redistribution phase. Throughout this section, we denote the dealer by D, the participants by $U_0, \ldots, U_{n-1}$ and the secrets by $K_1, \ldots, K_m$. Note that unlike the scheme of Wang et al., we do not restrict the number of secrets by t.

3.1 Initialization Phase

This phase is the same as Sect. 2.3.1, i.e., D publishes $< \alpha, G_1, G_2, e, P, h >$ and P_i, $i = 0, \ldots, n-1$ where $P_i = s_i P$ with U_i's secret shadow s_i. D further publishes a generator $g \in Z_\alpha^*$.

3.2 Secrets Distribution Phase

In this phase D performs the following steps.

a. Randomly selects $s \in Z_\alpha^*$ and publishes sP.
b. Constructs the matrix $M_{(n+m-t)\times(n+m)}$:

$$M = \begin{bmatrix} 1 & 1 & \ldots & 1 \\ 1 & g & \ldots & g^{n+m-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & g^{n+m-t-1} & \ldots & g^{(n+m-t-1)(n+m-1)} \end{bmatrix} \tag{3}$$

c. Computes $ss_i P$ together with $h(ss_i P)$ for $i = 0, \ldots, n-1$ and constructs the column vector
$A = [h(ss_0 P), h(ss_1 P), \ldots, h(ss_{n-1} P), K_1, \ldots, K_m)]^T$.
d. Publishes $< sP, C_0, \ldots, C_{n+m-t-1} >$ where

$$M \times A = \begin{bmatrix} 1 & 1 & \ldots & 1 \\ 1 & g & \ldots & g^{n+m-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & g^{n+m-t-1} & \ldots & g^{(n+m-t-1)(n+m-1)} \end{bmatrix} \begin{bmatrix} h(ss_0 P) \\ \vdots \\ h(ss_{n-1} P) \\ K_1 \\ \vdots \\ K_m \end{bmatrix} = \begin{bmatrix} C_0 \\ C_1 \\ \vdots \\ C_{n+m-t-1} \end{bmatrix} \tag{4}$$

3.3 Secrets Reconstruction and Verification Phase

Note that (4) is a system of $(n+m-t)$ linear equations in $(n+m)$ unknowns over G_2. Let t participants $U_i (i = 0, \ldots, t-1)$ pool their shares. When the combiner (who can be one of the participants) receives $s_i sP (i = 0, \ldots, t-1)$, s/he first checks if $e(ss_i P, P) = e(sP, s_i P)$. By the property of bilinear mapping, this ensures verifiability of the shares.

Next, s/he computes $h(ss_i P)$ for $i = 0, \ldots, t-1$. Therefore, t of the unknowns of (4) are determined and the combiner can now solve the system of $n+m-t$ equations and $n+m-t$ unknowns to recover the (m) secrets.

3.4 Secrets Redistribution Phase

The dealer chooses a new threshold t', new secrets $K_1', \ldots, K_{m'}'$, and the new seed s'. The dealer then proceeds as secrets distribution phase and finally publishes $< s'P, C_0', \ldots, C_{n+m'-t'-1}' >$. Unlike the scheme of Wang et al., we do not need to reconstruct the coefficient matrix (4) and only add or remove some of its rows and columns.

4 Security Analysis

We conduct security analysis of the proposed scheme by proving the following theorems.

Theorem 1 *Any t or more participants can reconstruct m secrets $K_1, K_2, \ldots, K_m$.*

Proof Without loss of generality, suppose that $U_i (i = 0, \ldots, t-1)$ provide their secret shadows $s_i s P (i = 0, \ldots, t-1)$. Then, (4) is reduced to a system of $n + m - t$ equations and $n + m - t$ unknowns with coefficient matrix:

$$M' = \begin{bmatrix} 1 & \cdots & 1 \\ g^{t} & \cdots & g^{n+m-1} \\ \vdots & \ddots & \vdots \\ g^{(t)(n+m-t-1)} & \cdots & g^{(n+m-1)(n+m-t-1)} \end{bmatrix} \tag{5}$$

M' is a Vandermonde matrix on distinct elements $\left(g^{t}, \ldots, g^{n+m-t-1}\right)$. Therefore, $det\left(M'\right) \neq 0$ and its inverse can be computed to obtain the secrets. □

Theorem 2 *Any group of $(t-1)$ (or fewer) participants can't compute any secrets.*

Proof This is clear since in this case, (4) is reduced to a system of $m + n - t$ equations and more than $n + m - t$ unknowns. □

Theorem 3 *The proposed scheme does not require a secure channel, i.e., the participants' shadows s_i can not be obtained from $P_i (= s_i P)$.*

Proof If an attacker wants to compute s_i from $s_i P$, s/he must solve an instance of the elliptic curve discrete logarithm problem in G_1 which is hard by our assumption on the choice of G_1. The same is true if the attacker chooses to use bilinear map $e(s_i P, P) = e(P, P)^{s_i}$ and reduce ECDLP in G_1 to an instance of DLP in G_2. □

Theorem 4 *The dealer's secret information s can not be obtained from public information sP.*

Proof If an attacker wants to compute s from sP, s/he must solve an instance of the elliptic curve discrete logarithm problem in G_1 which is hard by our assumption on the choice of G_1. The same is true if the attacker chooses to use bilinear map $e(sP, P) = e(P, P)^{s}$ and reduce ECDLP in G_1 to an instance of DLP in G_2. □

Theorem 5 *The shares provided by participants during the reconstruction phase can be verified so that cheaters are identified.*

Proof Suppose that U_i provides $s_i s P$. As mentioned above, given $s_i P$ and sP, it is computationally infeasible to compute $s_i s P$ in G_1. Therefore, only the dealer or U_i could compute this value. Hence, upon receiving $s_i s P$, we can use the property of bilinear mapping and check if $e(ss_i P, P) = e(sP, s_i P)$ holds to identify cheaters. □

5 Comparison

In this section, we compare our method with the scheme of Wang et al. [10] in two aspects: (1) the number of values published publicly and (2) the restrictions on m. we show that our scheme outperforms (WTS) in both of them.

We would like to mention that having few public values can be considered as an important factor in a secret sharing scheme. As an example, consider the steganographic sharing of a secret image considered in [16–19]. In this application, there is a secret image which should be shared secretely among n participants over open channels such as the internet. Therefore, data transmission should be done in such a way that invokes no suspicion and further any t of shareholders can reconstruct the image with verification. One possible solution is to select n innocent-looking images (called cover images) corresponding to participants and then embed the data of the secret image into these cover images using a secret sharing scheme. Now, since we have steganographic considerations, we can not publicly announce any values and must embed all information in cover images. Clearly, as the amount of data to be embedded increase, the visual quality of the resulting cover images will deteriorate and this in turn invokes suspicion. Therefore, if a secret sharing scheme has many public values (or very large ones), it can not be considered suitable for such applications.

We now consider the number of public values in our proposed scheme, denoted here by NPV_{Ours}, and the same number in (WTS), denoted by NPV_{WTS}. We have the following lemma.

Lemma $NPV_{\text{Ours}} \leq NPV_{\text{WTS}}$.

Proof We first compute NPV_{Ours}. During initialization, n values $(s_i P, i = 0, \ldots, n-1)$ are published. However, since these values are published only once (and the same happens in (WTS)), we do not consider them. During secret distribution, $< sP, C_0, \ldots, C_{n+m-t-1} >$ i.e. $n+m-t+1$ public values are announced. Therefore, $NPV_{\text{Ours}} = n+m-t+1$. Note that this number is the same whether $m \leq t$ or $m > t$.

In (WTS), again n values are once published for setup. However, in distribution phase, we have:

$$NPV_{\text{WTS}} = \begin{cases} n+1, & \text{if } m \leq t, \\ \left\lceil \frac{m}{t} \right\rceil n+1, & \text{otherwise.} \end{cases}$$

Moreover, to handle the case $(m > t)$, (WTS) is applied $\left\lceil \frac{m}{t} \right\rceil$ times. Now, It is not difficult to show that $NPV_{\text{Ours}} \leq NPV_{\text{WTS}}$. To see this

$$\begin{aligned} &If\ m \leq t: \ n+1 \geq n+m-t+1 \\ &If\ m > t: \ n \geq t \Rightarrow \frac{n}{t}(m-t) \geq m-t \Rightarrow \left(\frac{m}{t}-1\right) n \geq m-t \\ &\qquad\qquad \Rightarrow \left\lceil \frac{m}{t} \right\rceil n+1 \geq n+m-t+1 \end{aligned}$$

□

Another advantage for the proposed scheme is that the same matrix M (3) is used even if t or more the secrets are changed and there is no need to reconstruct it. However, if these changes occur in (WTS), the dealer should choose a new s and therefore the coefficient matrix M (1) should be reconstructed. Furthermore, in order to construct M in (WTS), the combiner

Table 1 Comparison between the two schemes

	Proposed scheme	Wang et al.
Precomputation of M	Yes	No
Restrictions on m	No	Yes
Verification	Yes	Yes
Updating	Yes	Yes
Multi-use	Yes	Yes
Ecc-based	Yes	Yes
No. of public values in Dist. phase	if $m \leq t\ n+m-t+1$	$n+1$
	if $m > t\ n+m-t+1$	$\lceil \frac{m}{t} \rceil n+1$
No. of public values in Recon. phase	t	$t(t-1)$

should wait till all the shares are provided, while our matrix M is constant and can be computed in advance. Finally, as seen from the proof of the above Lemma, our scheme does not restrict the number of secrets m as (WTS) do. Note also that all the other merits of (WTA) such as verification ability, updating of the secrets, having multi-use property as well as being ECC-based are preserved by our proposal. The results of comparison are summarized in the following table.

6 Conclusion

In this paper, we propose a new dynamic threshold multi-secret sharing scheme. Our scheme is multi-use so that participants can use their private shadows even if the secrets change. The scheme can further identify cheaters using bilinear mapping. Compared to existing methods in the literature, our scheme achieves better performance regarding public values (Table 1).

Acknowledgments The research of the first author was in part supported by a grant from IPM (No. 89940041).

References

1. Shamir, A. (1979). How to share a secret. *Communication of the ACM, 22*, 612–613.
2. Blakley, G. (1979). Safeguarding cryptographic keys. *AFIPS Conference Proceedings, 48*, 313–317.
3. Zhao, J., Zhang, J., & Zhao, R. (2007). A practical verifiable multi-secret sharing scheme. *Computer Standards Interface, 29*, 138–141.
4. He, J., & Dawson, E. (1994). Multistage secret sharing based on one-way function. *Electronics Letters, 30*(19), 1591–1592.
5. Jackson, W. A., Martin, K. M., & O'keefe, C. M. (1994). On sharing many secrets. *Asiacrypt, 94*, 42–54.
6. Chor, B., & Goldwasser, S. (1985). Verifiable secret sharing and achieving simultaneity in the presence of faults [a]. In *Proceedings of 26th IEEE Symposium.* FOCS, pp. 251–260.
7. Hwang, R. -J., & Chang, C. -C. (1998). An on-line secret sharing scheme for multi secrets. *Computer Communications, 21*(13), 1170–1176.
8. Shao, J., & Cao, Z.-F. (2005). A new efficient (t,n) verifiable multi-secret sharing (vmss) based on ych scheme. *Applied Mathematics and Computation, 168*, 135–140.

9. Chen, W., Long, X., Bai, Y. B., & Gao, X. P. (2007). A new dynamic threshold secret sharing scheme from bilinear maps. In *International conference on parallel processing workshops*, pp. 19–22.
10. Wang, S. J., Tsai, Y. R., & Shen, J. J. (2008). *Verifiable threshold scheme in multi-secret sharing distributions upon extensions of ecc, Wireless Pers Commun* (pp. 405–410). New York: Springer. doi:10.1007/s11277-009-9875-0.
11. Koblitz, N. (1993). *Introduction to elliptic curves and modular forms*. New York: Springer.
12. Washington, L. C. (2003). *Elliptic curves: Number theory and cryptography*. Boca Raton: CRC Press.
13. Boneh, D., & Franklin, M. (2001). Identity-based encryption from the weil pairing. *Advances in Cryptology, Lecture Notes on Computer Science, 2139*, 213–229.
14. Galbraith, S. D., Harrison, K., & Soldera, D. (2002). Implementing the tate pairing. *Algorithmic Number Theory Symposium ANTS-V, LNCS, 2369*, 324–337.
15. Lee, H. -S. (2004). Self-pairing map and its applications to cryptography. *Applied Mathematics and Computation, 151*, 671–678.
16. Eslami, Z., Razzaghi, S., & Ahmadabadi, J. Z. (2010). Secret image sharing based on cellular automata and steganography. *Pattern Recognition, 43*, 397–404.
17. Lin, C., & Tsai, W. (2004). Secret image sharing with steganography and authentication. *The Journal of Systems and Software, 73*, 405–414.
18. Yang, C., Chen, T., Yu, K., & Wang, C. (2007). Improvements of image sharing with steganography and authentication. *The Journal of Systems and Software, 80*, 1070–1076.
19. Chang, C., Hsieh, Y., & Lin, C. (2008). Sharing secrets in stego images with authentication. *Pattern Recognition, 41*, 3130–3137.

Author Biographies

Ziba Eslami received her B.S., M.S., and Ph.D. in Applied Mathematics from Tehran University in Iran. She received her Ph.D. in 2000. From 1991 to 2000, she was a resident researcher in the Institute for Studies in Theoretical Physics and Mathematics (IPM), Iran. During the academic years 2000–2003, she was a Post Doctoral Fellow in IPM. She served as a non-resident researcher at IPM during 2003–2005. Currently, she is the Head of and a Professor in the Department of Computer Sciences at Shahid Beheshti University in Iran. Her research interests include design theory, combinatorial algorithms, cryptographic protocols, and steganography.

Saideh Kabiri Rad received her B.S. degree in Computer Science in 2007 from Shahid Bahonar University, Kerman, Iran. She is currently an M.S. student of Computer Science in Shahid Beheshti University, Tehran, Iran.