A Proxy E-Raffle Protocol Based on Proxy Signatures

Nasrollah Pakniat and Ziba Eslami
Department of Computer Science
Shahid Beheshti University, G.C.
Tehran, Iran
n.pakniat@mail.sbu.ac.ir, z_eslami@sbu.ac.ir

Abstract

In 2009, Chang and Cheng proposed an efficient proxy raffle protocol. In their scheme, the raffle tickets are known by participants as well as raffle organizers. Hence, unless we implicitly add the unrealistic assumption that the organizer is trustworthy, this may cause problems and threaten the security of the e-raffle from the viewpoint of participants. In this paper, we use the concept of proxy signatures and symmetric cryptography to propose an efficient proxy raffle scheme which overcomes this weakness. We show that the proposed scheme achieves all other security requirements mentioned in the literature. The proposed scheme is shown to outperform the scheme of Chang and Cheng in terms of communication load and computational complexity.

Keywords: E-Raffles, Proxy raffle protocol, Proxy signature

1 Introduction

One of the most popular low-cost marketing strategies is offering a raffle draw upon purchase. Merchants usually propose attractive raffles especially at their annual celebrations. Depending on the amount of their purchase, customers will be entitled to obtain raffle tickets. In order to encourage consumers, the prizes must naturally be valuable and of interest to consumers. This can therefore be an extremely powerful method to promote the sale. On the other hand, with the advancement of network technologies, more and more commercial transactions are done over the Internet (e-commerce) and this makes the design of secure and fair electronic raffle schemes a challenging task. It should also be noted that e-raffles are fairly a new research area and there is yet no standard for such protocols. The essential requirements of an e-raffle scheme reported so far are as follows [1, 2].

Anonymity: Participants should remain anonymous in the overall raffle protocol to ensure privacy and security. Participants can not link a raffle ticket to the one who casts it (except their own) and the true identity of the winner remains a secret.

Accuracy: It should be impossible to modify or remove raffle tickets.

Verifiability: All valid raffle tickets must be publicly verifiable. No valid raffle ticket can be miscounted or removed.

Fairness: Each participant has an equal chance of winning the prize. No one can predict or intervene in the outcome.

Security: No one can masquerade as a qualified consumer in order to request a raffle ticket. The prize shall be rewarded only to the real raffle winner.

In 2005, Chen et al. proposed the first raffle scheme for the Internet [1]. This scheme is based on blind signatures, hashing chain [3] and the Secure Socket Layer (SSL) protocol [4, 5]. The scheme satisfies accuracy, verifiability, and fairness. In [2], it was shown that Chen et al.'s scheme doesn't preserve anonymity of winner and is vulnerable to impersonation, denial of service and man-in-the-middle attacks. In 2009, Chang and Cheng [2] used the concepts of symmetric and asymmetric cryptography to propose a more efficient raffle scheme which also overcomes previous weaknesses. However, in the e-raffle protocol proposed by Chang and Cheng, the raffle organizer knows the content of tickets issued to participants. Therefore in case of malicious behavior, it is possible that several valid participants claim the prize while the scheme is unable to detect who is guilty. For example, consider the situation where a malicious raffle organizer reveals the winner's ticket to another participant (E). Now, it is possible for the raffle organizer to grant (E) the prize unjustly and without being detected. The same problem occurs if the true winner reveals his/her ticket information to another participant. In other words, the fact that two different entities have access to raffle tickets, makes it impossible to determine the source of malicious behavior. Note that this does not happen in traditional paper-based raffle games where there exists a single hard copy of each raffle coupon and the raffle organizers are with no doubt found guilty in case of multiple copies. It is therefore important to define a new security requirement which puts the responsibility on a certain entity in case of dispute in e-raffle schemes. The goal of the present paper is to propose a new efficient proxy raffle scheme which achieves all the above-mentioned requirements as well as a new one which we define as the following:

participant-only ticket accessibility (POTA): The only entity who has access to a ticket is its owner. No raffle authority knows the contents of tickets.

In this paper, we propose a new simple raffle scheme based on proxy signatures and symmetric cryptography [6, 7, 8, 9]. We also show that the proposed scheme satisfies all security requirements including the one defined here. The proposed scheme is shown to be more efficient than previous schemes in both communication load and computation complexity.

The rest of the paper is organized as follows. In Section 2, we review proxy signatures. Our motivation for the proposed proxy raffle scheme and its details are presented in Section 3. We analyze the security of the proposed scheme and also a comparison with the previous schemes provided in Section 4. Finally, conclusions of the paper are presented in Section 5.

2 Proxy signature

The concept of proxy signature was introduced by Mambo et al. in [10]. Proxy signatures can be used in cases where due to some reasons (such as absence, workload,etc), an original signer wants to delegate his/her signing rights to other users, called proxy signers who can sign on the behalf of the original signer. The requirement of proxy signer's privacy protection is needed in some practical applications. The security requirements for an anonymous proxy signature are formalized as follows [11]:

- **Distinguishability** Proxy signatures are distinguishable from normal signatures by everyone.
- **Verifiability** From the proxy signature, a verifier can be convinced of the original signer's agreement on the signed message.
- **Strong unforgeability** A proxy signer can create a valid proxy signature for the original signer. But the original signer and any third party cannot create a valid proxy signature on behalf of the proxy signer.
- **Anonymity** Only the original signer can determine the identity of the corresponding proxy signer from the proxy signature.
- **Non-deniability** Once a proxy signer creates a valid proxy signature on behalf of an original signer, he/she cannot repudiate the signature creation.

In order to prevent misuse, there exist different types of delegation in the literature. The one we consider in this paper is delegation by warrant. A warrant is a certificate signed by the original signer indicating the validity period of delegation, the public key of the proxy signer, and the type of messages authorized to be signed. Note that to achieve the anonymity requirement, the warrant should not contain the identity of the proxy signer. To ensure strong unforgeability, a proxy signing key is produced from the delegation key issued by the original signer and the private key of the proxy signer. An anonymous proxy signature scheme consists of the following algorithms.

Key generation This is a probabilistic polynomial-time (PPT) algorithm. Given a security parameter k, output a personal public-private key pair $(pk,\ sk)$.

Delegation signing On input a warrant $Wrnt$, proxy signer's public key pk_P and the original signer's private key sk_O, output a signature σ (a delegation key $dKey_{O\to P}$) on $Wrnt$.

Delegation verification On input the original signer's public key pk_O and his/her signature σ on $Wrnt$ (the delegation key $dKey_{O\to P}$), output "accept" if the signature is valid, and "reject" otherwise.

Proxy key generation On input the proxy signer's private key sk_P and the delegation key $dKey_{O\to P}$, output a proxy signing key $psKey_{O\to P}$.

Proxy signing On input a message m, the public key of the original signer and the proxy signer, the warrant $Wrnt$ and the proxy signing key $psKey_{O\to P}$, output an anonymous proxy signature σ for the message m.

Proxy signature verification On input a message m, an anonymous proxy signature σ, the public key of the original signer and the proxy signer, the warrant $Wrnt$, output "accept" if the signature is valid, and "reject" otherwise.

Examples of anonymous proxy signatures can be found in [11, 12, 13, 14, 15, 16].

3 The proposed scheme

In a raffle scheme there are three participants, 1) a raffle originator (RO), 2) the raffle participants $(P_i, 1 \leq i \leq n)$ and 3) a raffle proxy center (RPC). The raffle originator is responsible for issuing the raffle tickets for qualified participants who later cast their tickets to (RPC). (RPC) is the entity who performs the final drawing phase of the raffle and simulates the witness or lawyer in traditional raffles. Each valid participant P can apply for one or more raffle tickets from the raffle originator (RO).

Our scheme consists of four phases: the initial phase, the raffle ticket issuing phase, the raffle ticket casting and drawing phase, and the raffle prize claiming phase. The diagram of different phases of the proposed scheme are depicted in Figure 1.

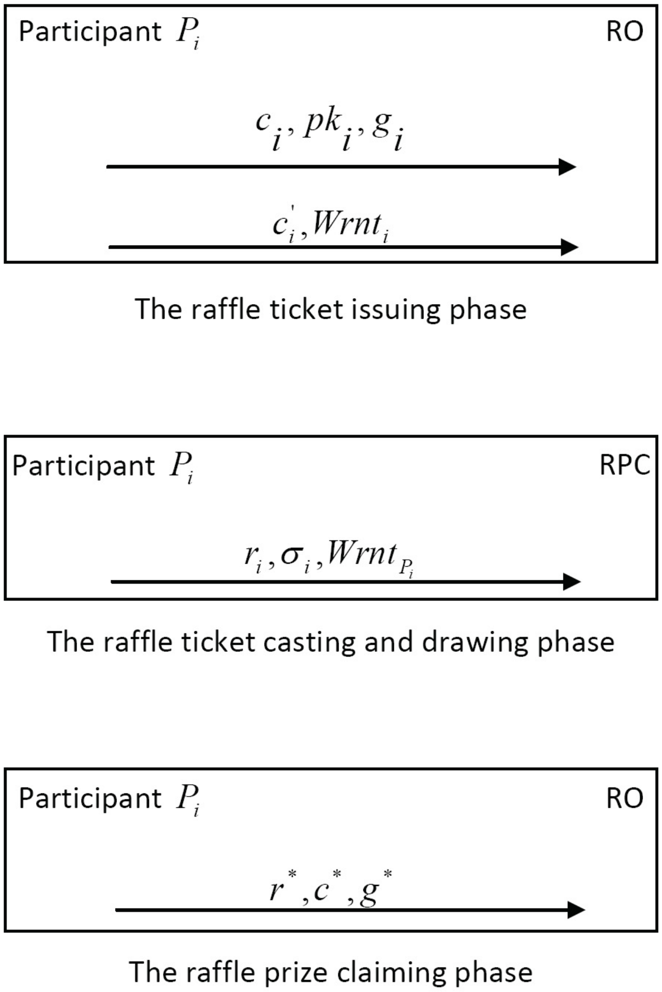
In this section, we first provide the notations used throughout the paper in Table 1, then present our motivation to use proxy signatures in designing the proposed proxy raffle scheme and finally provide the details of the protocol.

3.1 Our motivation to use proxy signatures

Consider an anonymous proxy signature scheme with (RO) as the original signer and the participants as proxy signers. Let the raffle tickets be the proxy signing key of participants. Therefore, by the properties of such signatures, only participants have access to the tickets and this seems to be a good mechanism to achieve participant-only-ticket-accessibility property. We now show how to adopt this idea to propose an e-raffle which satisfies other security requirements as well.

3.2 The scheme

The details of different phases of the scheme are as follows.



The raffle ticket issuing phase

The raffle ticket casting and drawing phase

The raffle prize claiming phase

Figure 1: The diagram of our proposed scheme.

Table 1: Notations

RO	the raffle originator,
RPC	the trusted raffle proxy center,
P_i	a participant,
SN_i	the serial number of P_i's shopping list,
$K_{X\leftrightarrow Y}$	the common session key between X and Y,
sk_i/pk_i	the secret/public key of P_i,
$Enc_K()/Dec_K()$	the symmetric encryption/decryption functions with common key K,
$Wrnt_{O\rightarrow P}$	the warrant given by O to P,
$dKey_{O\rightarrow P}$	the delegation key that O assigns to the proxy signer P,
$psKey_{O\rightarrow P}$	the proxy signing key which the proxy signer P computes from the delegation key of O assigned to him and his/her own private key.
$psign(psKey_{O\rightarrow P}, m)$	the (proxy) signature of P on behalf of O on the message m, where $psKey_{O\rightarrow P}$ denotes the proxy signing key.

3.2.1 The initial phase

In this phase (RO) performs the following steps:

1. Chooses a large prime number p and selects a primitive element g from the Galais field with p elements, i.e. $GF(p)$.
2. Chooses a random number $x \in z_p^*$.
3. Computes $g_0 = g^x \ (mod\ p)$.
4. Publishes $p,\ g,\ g_0$.
5. Chooses a secure symmetric encryption/decryption algorithm (such as PKCS11 [6], PKCS12 [7], PKCS15 [8], and AES [9]).
6. Chooses a provably secure anonymous proxy signature scheme (such as schemes proposed in [11, 12, 13, 14, 15, 16]).

3.2.2 The raffle ticket issuing phase

We assume that each qualified participant P_i (who has ordered certain amount of online purchase) gets SN_i as the receipt of his/her shopping. Then he/she can ask for a raffle ticket from (RO) through the following procedure.

1. P_i:
   (a) Selects a random number $x_i \in z_p^*$.
   (b) Runs the key generation algorithm of anonymous proxy signature scheme to obtain $(sk_i.pk_i)$ as his (private,public) key.
   (c) Computes $g_i = g^{x_i} \ (mod\ p)$.
   (d) Computes $K_{P_i \leftrightarrow RO} = g_0^{x_i} \ (mod\ p)$ as the current common session key.
   (e) Computes $c_i = Enc_{K_{P_i \leftrightarrow RO}} \ (SN_i, pk_i)$.
   (f) Sends $c_i,\ g_i$ to (RO).
2. (RO) after receiving $(c_i,\ g_i,\ pk_i)$:
   (a) Computes $K_{P_i \leftrightarrow RO} = g_i^x = (g^{x_i})^x = g_0^{x_i} \ (mod\ p)$ as the current common session key.
   (b) Computes $SN_i, pk_i = Dec_{K_{P_i \leftrightarrow RO}}(c_i)$.
   (c) Checks the validation of SN_i, If valid, then marks SN_i as an exchanged one; otherwise, the request is rejected.
   (d) Computes the warrant $(Wrnt_{P_i})$. The warrant contains information such as pk_i. The warrant should not contain any information about participant's identity.
   (e) Computes delegation key $dKey_{RO \to P_i}$ from $(sk_{RO}$ and $Wrnt_{P_i})$.
   (f) Computes
   $$c_i' = Enc_{K_{P_i \leftrightarrow RO}} \ (dKey_{RO \to P_i})$$
   .
   (g) Sends $c_i',\ Wrnt_{P_i}$ to P_i.

3. P_i after receiving c'_i, $Wrnt_{P_i}$:

   (a) Computes
   $$dKey_{RO \to P_i} = Dec_{K_{P_i \leftrightarrow RO}}\,(c'_i)$$
   .

   (b) Verifies the validity of proxy key $dKey_{RO \to P_i}$, if o.k., computes his/her proxy signing key, $d_{RO \to P_i}$ from $dKey_{RO \to P_i}$, $Wrnt_{P_i}$ and his/her secret value (sk_i).

   (c) Stores the proxy signing key as his/her ticket for the raffle.

3.2.3 The raffle ticket casting and drawing phase

When P_i receives a raffle ticket, he/she can join the drawing through the following steps.

1. P_i:

   (a) Chooses a random r_i as a seed.

   (b) Computes the proxy signature $\sigma_i = psign_{d_{RO \to P_i}}(r_i)$.

   (c) Sends $(r_i,\ \sigma_i,\ Wrnt_{P_i})$ to (RPC).

2. (RPC) after receiving $(r_i,\ \sigma_i,\ Wrnt_{P_i})$:

   (a) Checks the validity of $Wrnt_{P_i}$.

   (b) Checks the validity of the signature (σ_i) on r_i with warrant $Wrnt_{P_i}$.

   (c) If the above conditions satisfied, publishes r_i, $Wrnt_{P_i}$ in a public list. So the participant will be sure that he/she is in the raffle.

3. When reaching the drawing deadline, (RPC) computes $w = R\ (r_1,\ r_2,\ ...,\ r_n)$, where n is the number of all raffle tickets, and $R\ ()$ is a function which returns one of its inputs at random. Then (RPC) publishes $Wrnt_{winner}$ as the winner ticket.

3.2.4 The raffle prize claiming phase

After the winning ticket is announced, the winner can claim his/her prize from (RO) before the deadline as follows.

1. The winner:

   (a) Chooses a random number $x^* \in z_p^*$.

   (b) Computes $g^* = g^{x^*}\ (mod\ p)$,
   $$K_{winner \leftrightarrow RO} = g_0^{x^*}\ (mod\ p)$$
   .

   (c) Computes the proxy signature $\sigma^* = psign_{d_{RO \to winner}}(AI_{winner})$, where AI_{winner} is the account information of the winner.

   (d) Computes
   $$c^* = Enc_{K_{winner \leftrightarrow RO}}\,(\sigma^*,\ AI_{winner})$$
   .

(e) Sends $(r^*,\ c^*,\ Wrnt_{winner},\ g^*)$ to (RPC).

2. (Ro) after receiving $(r^*,\ c^*,\ Wrnt_{winner},\ g^*)$:

(a) Computes $K_{RO\leftrightarrow winner}\ =\ g^{*^{x}}\ (mod\ p)$.

(b) Computes σ^*,

$$AI_{winner}\ =\ Dec_{K_{RO\leftrightarrow winner}}\ (c^*)$$

.

(c) Checks the validity of σ^* with warrant $Wrnt_{winner}$ on the message AI_{winner}. If it is true, so he's winner.

4 Analysis of the proposed scheme

In this section, we first consider the performance of the scheme against malicious behavior such as replay, impersonation and denial of service attacks. We then analyze the security of our proposed electronic raffle system and show that our scheme can achieve the essentials mentioned in Section 1. Furthermore, we compare the functionality and performance of our proposal with previous schemes.

4.1 Attacks

Assume that there exists a malicious attacker Eve in the communication models who can intercept and observe any publicly exchanged information between two entities. In this section, we show that our scheme is secure against the replay attack, impersonation, and denial of service attacks.

4.1.1 Replay attacks:

Since all messages are encrypted with a common session key between P_i and (RO), so Eve is unable to obtain any information about SN_is and $dKey_{RO\rightarrow P_i}$ from g^x, g^{x_i}, c_i and c_i' and is therefore unable to perform replay attacks.

4.1.2 Impersonation attacks:

Assume that Eve intercepts the message c_i in Step 1f of the Raffle ticket issuing phase and intends to impersonate a legal user in order to apply for a ticket from (RO). To do so, she must randomly select x_i^*, then forge and send a request message $c_i^* = Enc_{K^*_{P_i\leftrightarrow RO}}(SN_i,),\ g_i^*$ to (RO), where $g_i^* = g^{x_i^*}\ (mod\ p)$.However, , since the original c_i is encrypted with a common session key between P_i and (RO), the fake message will not pass the check in Step 2c of this phase. This means that Eve is unable to obtain the qualified SN_i and consequently cannot generate a valid c_i^*.

In the raffle prize claiming phase, the winner signs his/her account information (with his/her private proxy signing key) and then encrypts both these information and the signature with a common session key between himself/herself and (RO). Therefore, in order for Eve, to impersonate the winner, she must first decrypt the ciphertext and then forge the winner's signature on another message. This is impossible by the property of symmetric cryptography and proxy signatures.

4.1.3 Denial of service (DoS) attacks:

This is simply impossible since the winner's warrant is published by (RPC).

Table 2: Functionality comparison between our schemes and previous works

	Our scheme	Chang and Cheng scheme	Chen et al. scheme
Anonymity	Yes	Yes	No
Accuracy	Yes	Yes	Yes
Verifiability	Yes	Yes	Yes
Fairness	Yes	Yes	Yes
Security	Yes	Yes	No
(POTA)	Yes	No	No

Table 3: Resistance against various attacks

attack	Our scheme	Chang and Cheng scheme	Chen et al. scheme
False claim of raffle ticket	Yes	Yes	No
False claim of raffle prize	Yes	No	No
Denial of service attack	Yes	Yes	No
Malicious (RO)	Yes	No	No

4.2 Requirements analysis

In this section, we show that our scheme can achieve the essentials of a general electronic raffle scheme as mentioned in Section 1. Throughout this section, we assume that the underlying anonymous proxy signature has provable security (such as [11]). We also assume (as in other existing e-raffle protocols [1, 2]) that (*RPC*) is honest. The results are summarized in Table 2 and Table 3.

4.2.1 Anonymity

There is no polynomially bounded adversary $\mathscr{A}$ who is able to link a transmitted message to a valid participant.

Proof In raffle ticket issuing phase, by using a pure serial number SN_i from P_i's shopping list, P_i's identity is not transmitted publicly. This information is not embedded in the transmitted messages either. Moreover, the transmitted messages are encrypted with a symmetric key. Hence, $\mathscr{A}$ isn't able to determine the identity of ticket's owner from a ticket. In the raffle ticket casting and drawing phase, the transmitted messages consist entirely of some random values plus anonymous proxy signatures on them. Therefore, if $\mathscr{A}$ could determine the identity of senders from transmitted messages, then she could link these signatures to actual signers. By the properties of strongly unforgeable anonymous proxy signatures, this is impossible. The messages transmitted during the prize claiming phase are also encrypted and do not reveal any identity information.

4.2.2 Accuracy

(RPC) publishes the warrants of all participants who have cast their tickets in a public list. If we assume the trustworthiness of the (RPC), it is possible for participants to ensure they are included in the raffle.

4.2.3 Verifiability

The tickets generated by our scheme are nothing but proxy signing keys computed from participants secret key plus the delegation key issued by (RO). Therefore, to verify valid tickets, it is enough to sign an arbitrary message (with ticket as the key) and see if the signature is verified successfully. This means that verifiability can be achieved from the security of the underlying proxy signature. Note that by the properties of proxy signature, no one except (RO) can generate delegation keys.

4.2.4 Fairness

The entity (RPC) is applied to the new scheme to coordinate the raffle. The reliability of (RPC) is based on the publicly verifiable raffle drawing procedure. We showed that participant's raffle tickets cannot be miscounted or removed. Furthermore, the raffle casting and drawing processes are publicly verifiable. Hence, no one can predict or intervene the outcome and each participant has an equal chance of winning the prize.

4.2.5 Security

In order for a raffle participant to request a ticket, she/he must present a valid SN_i which is issued by (RO). The encryption of all transmitted messages in ticket issuing phase makes it impossible for attackers to obtain a valid SN_i and masquerade as a qualified consumer. On the other hand, the warrant of the winner is published so that only the owner of the corresponding proxy signing key (the true winner) can claim the prize.

4.2.6 participant-only ticket accessibility

We used a proxy signing key generated by a strong unforgeable proxy signature scheme as raffle tickets. The signing key of a participant is computed from participant's secret key plus the delegation key issued by (RO). Therefore, no one except the participant has access to the ticket.

4.3 Performance analysis

In this section, we compare the proposed scheme with previous schemes in terms of computation complexity and communication load. The results show that the proposed scheme outperforms existing approaches in both aspects.

We conduct a comparison among our scheme, the method of [1] and [2]. The properties we consider are computation load for each raffle participant C_1, computation cost of (RO) denoted by C_2, the computation cost of (RPC) (C_3), and (C_4) the communication load of the raffle ticket casting and drawing phase for (RPC). We summarize the results in Table 4 where the following notations are used: T_{asym} denotes the computation time for an asymmetric encryption, decryption, signing or verification, T_{sym} stands for computation time of a symmetric encryption or decryption, T_e represents the computation time of one modular exponentiation and T_h is the computation time for a one-way hash function. Following [17, 18], we assume that

$1\ T_{asym} = 100\ T_{sym},\ 1\ T_{sym} = \frac{5}{3}\ T_e$ and $1\ T_e = 600\ T_h$

for software consideration and

$1\ T_{asym} = 1000\ T_{sym},\ 1\ T_{sym} = \frac{5}{3}\ T_e$ and $1\ T_e = 6000\ T_h$

for hardware consideration. As indicated by Table 4, the proposed scheme is superior to both other methods. Note that since in our scheme warrants are published by (RPC), the participants do not need to obtain a receipt and hence the results about (C_4) are obtained.

5 Conclusion

In this paper, we propose an efficient proxy raffle protocol based on the concept of proxy signatures. The scheme achieves participant-only ticket accessibility property which prevents malicious behavior of

Table 4: Performance comparison

	Chenetal	*ChangandCheng*	*Our Scheme*
C_1	$2\ T_{asym} + 2\ T_H + 2\ T_e$	$3\ T_{asym} + 2\ T_{sym} + 4\ T_e$	$2\ T_{asym} + 3\ T_{sym} + 4\ T_e$
C_2	$6\ T_{asym}$	$2\ T_{asym} + 3\ T_{sym} + 2\ T_e$	$2\ T_{asym} + 3\ T_{sym} + 2\ T_e$
C_3	$4\ T_{asym} + 2\ T_H$	$1\ T_{asym} + 3\ T_{sym} + 2\ T_e$	$1\ T_{asym}$
C_4	1	1	0

C_1 : computation cost for each raffle participant
C_2 : computation cost of (RO)
C_3 : computation cost of (RPC)
C_4 : communication cost of (RPC) in ticket casting and drawing phase

raffle entities. Our proposed scheme satisfies all other security requirements mentioned in the literature. In terms of computational complexity and communication load, the scheme is shown to outperform existing approaches.

References

[1] Y.-Y. Chen, J.-K. Jan, and C.-L. Chen, “Design of a fair proxy raffle protocol on the internet,” *Computer Standards & Interfaces*, vol. 27, no. 4, pp. 415–422, April 2005.

[2] C.-C. Chang and T.-F. Chenga, “An efficient proxy raffle protocol with anonymity-preserving,” *Computer Standards & Interfaces*, vol. 31, no. 4, pp. 772–778, June 2009.

[3] J. Zhou and C. Tan, “Playing lottery on the internet,” in *Proc. of the 3rd International Conference on Information and Communications Security (ICICS’01), Xian, China, LNCS*, vol. 2229. Springer-Verlag, November 2001, pp. 189–201.

[4] A. O. Freier, P. Karlton, and P. C. Kocher, “The ssl protocol version 3.0,” IETF Internet-draft, November 1996, http://tools.ietf.org/html/draft-ietf-tls-ssl-version3.

[5] D. Wagner and B. Schneier, “Analysis of the ssl 3.0 protocol,” in *Proc. of the 2nd USENIXWorkshop on Electronic Commerce (WOEC’96), Berkeley, California, USA*, vol. 2. USENIX Press, November 1996, pp. 29—-40.

[6] RSA Laboratories, “PKCS #11 v2.20: Cryptographic Token Interface Standard,” RSA Lab. PKCS #11, June 2004, ftp://ftp.rsa.com/pub/pkcs/pkcs-11/v2-20/pkcs-11v2-20.pdf.

[7] ——, “PKCS #12 v1.0: Personal Information Exchange Syntax,” RSA Lab. PKCS #12, June 1999, ftp://ftp.rsasecurity.com/pub/pkcs/pkcs-12/pkcs- 12v1.pdf.

[8] ——, “PKCS #15 v1.1: Cryptographic Token Information Syntax Standard,” RSA Lab. PKCS #15, June 2000, ftp://ftp.rsasecurity.com/pub/pkcs/pkcs-15/pkcs- 15v1-1.pdf.

[9] J. Daemen and V. Rijmen, *The Design of Rijndael: AES - The Advanced Encryption Standard*. Springer-Verlag, 2002.

[10] M. Mambo, K. Usuda, and E. Okamoto, “Proxy signature: delegation of the power to sign messages,” *IEICE Transactions on Fundamentals*, vol. E79-A, no. 9, pp. 1338–1354, September 1996.

[11] Y. Yu, C. Xu, X. Huang, and Y. Mu, “An efficient anonymous proxy signature scheme with provable security,” *Computer Standards & Interfaces*, vol. 31, no. 2, pp. 348–353, February 2009.

[12] S. Kim, S. Park, and D. Won, “Proxy signature, revisited,” in *Proc. of the 3rd International Conference on Information and Communications Security (ICICS’01), Beijing, China, LNCS*, vol. 1334. Springer-Verlag, November 1997, pp. 223–232.

[13] B. Lee, H. Kim, and K. Kim, “Secure mobile agent using strong non-designated proxy signature,” in *Proc. of the 6th Australasian Conference on Information Security and Privacy (ACISP’01), Sydney, Australia, LNCS*, vol. 2119. Springer-Verlag, July 2001, pp. 474–486.

[14] M.-S. Hwang, S.-F. Tzeng, and C.-S. Tsai, "Generalization of proxy signature based on elliptic curves," *Computer Standards & Interfaces*, vol. 26, no. 2, pp. 73–84, March 2004.

[15] X. Huang, Y. Mu, W. Susilo, F. Zhang, and X. Chen, "A short proxy signature scheme: efficient authentication in the ubiquitous world," in *Proc. of the 2005 International Conference on Embedded and Ubiquitous Computing (EUC'05), Nagasaki, Japan, LNCS*, vol. 3823. Springer-Verlag, December 2005, pp. 480–489.

[16] X. Hu and S. Huang, "A novel proxy key generation protocol and its application," *Computer Standards & Interfaces*, vol. 29, no. 2, pp. 191–195, February 2007.

[17] B. Schneier, *Applied Cryptography*, 2nd ed. John Wiley and Sons press, 1996.

[18] Bruce Schneier, *Applied Cryptography, Protocols, Algorithms, and Source Code in C*, 2nd ed. John Wiley and Sons press, 1996.

Nasrollah Pakniat received his B.S. degree in Computer Science in 2008 from Shahid Bahonar University, Kerman, Iran, and M. S. degree in Computer Science from Shahid Beheshti University, Tehran, Iran.

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 an associate 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.